# Building Sustainable Supply Chains in the Era of E-Commerce

by
Christian Gatmaitan
B.S. Industrial Engineering, Purdue University, 2013
and
Lisha Yangali
MBA, Pontificia Universidad Católica Del Perú, 2018

SUBMITTED TO THE PROGRAM IN SUPPLY CHAIN MANAGEMENT
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF
MASTER OF APPLIED SCIENCE IN SUPPLY CHAIN MANAGEMENT
AT THE MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2020



Signature of Author: _____
Christian Gatmaitan
Department of Supply Chain Management
May 8, 2020

Signature of Author: _____
Lisha Yangali
Department of Supply Chain Management
May 8, 2020

Certified by: _____
Dr. Josué C. Velázquez Martínez
Executive Director, Supply Chain Management Program
Capstone Advisor

Certified by: _____
Dr. Andrés Muñoz-Villamizar
Postdoctoral Associate
Capstone Co-Advisor

Accepted by: _____
Dr. Yossi Sheffi
Director, Center for Transportation and Logistics
Elisha Gray II Professor of Engineering Systems
Professor, Civil and Environmental Engineering

# Building Sustainable Supply Chains in the Era of E-Commerce

By

Christian Gatmaitan
B.S. Industrial Engineering, Purdue University, 2013

and

Lisha Yangali
MBA, Pontificia Universidad Católica Del Perú, 2018

Submitted to the Program in Supply Chain Management
on May 8, 2020, in Partial Fulfillment of the
Requirements for the Degree of Master of Applied Science in Supply Chain Management

ABSTRACT
Consumer preferences are driving changes within the retail space. E-Commerce is growing rapidly and there are increased pressures on companies to be environmentally friendly, yet still cost-competitive. Therefore, it is of the utmost importance for retailers to adjust their distribution network to accommodate these new customer preferences. Four key steps of this process are (1) measuring the current capacity of the system, (2) forecasting future demand, (3) adjusting capacity in order to meet this forecasted demand while minimizing cost, and (4) quantifying the environmental impact of each adjustment.
Our research focuses on applying these four steps to Coppel, a Mexican retail chain. A time study and capacity analysis were completed to understand current-state capacity. An annual forecast was created using the Holt-Winter Method. This forecast was used as an input for our developed Mixed Integer Linear Program (MILP) that minimizes cost while still meeting customer demand. Finally, the Greenhouse Gas (GHG) method was used to quantify the environmental impact of each cost-reduction measure that the MILP suggests. Our approach reduced the processing cost per unit for Coppel by 4.8% for a total savings of $126M when compared to their previous current-state. This result verifies that our four-step approach can be utilized to reduce cost and therefore yield higher profits for companies in the retail space.

Capstone Advisor: Dr. Josué C. Velázquez Martínez
Title: Executive Director, Supply Chain Management Program

Capstone Co-Advisor: Dr. Andrés Muñoz-Villamizar
Title: Postdoctoral Associate

## ACKNOWLEDGMENTS

We would like to express our sincere gratitude to our advisors, Dr. Andrés Muñoz Villamizar and Dr. Josué Velázquez Martínez, for their guidance and encouragement through each stage of the project.

The contribution from Coppel and their team members to our project was invaluable to our education and we hope our project will make a positive impact on their organization.
We would also like to thank Pamela Siska, Toby Gooley, and the rest of the CTL staff for their support throughout the entirety of this project.

Finally, a special thank you to all of our loved ones for supporting us throughout our project and journey at MIT.

# TABLE OF CONTENTS

# List of Tables

# List of Figures

# CHAPTER 1: INTRODUCTION

Businesses must align with the wants and needs of their customers in order to remain competitive in any industry. Consequently, supply chains are designed to accommodate consumer preferences and must be altered as preferences change. In this capstone project, we created a model to help businesses adjust their supply chains to adapt to two recent trends that have been transforming the retail industry: rising demand due to the growth of e-commerce and the increased desire for sustainable supply chains.

## *1.1 The Rise of E-Commerce*

Worldwide, consumers spent approximately US$3.46 trillion online in 2019, which is a 17.9% increase from 2018 (Young, 2019). The level of growth varies by country, but the same trend is being experienced worldwide. E-commerce sales in the United States increased by 16.4% from 2018 to a total of US$602 billion in 2019. Developed countries[1] are generally facing lower growth rates than developing countries due to the longer presence of e-commerce and therefore higher saturation of the market. The United States has seen a constant growth of ~ 7% of e-commerce as a percentage of total retail sales over the past decade (Figure 1), in contrast to the higher growth rates that developing countries have experienced. For example, in 2017, only 7% of the Mexican population shopped online once a week. In 2018, that number grew to 38% (Forbes, 2019). This global growth trend is projected to continue until at least 2024, and therefore businesses need to plan accordingly (Clement, 2019). To effectively manage this expected growth, businesses need to understand their current-state supply chain capacity, growth in their specific markets, and optimize their supply chain network to accommodate their forecast.

[1] Developed and undeveloped economies are defined by the United Nations based on per capita gross national income (GNI) and other factors (United Nations, 2019)


*Figure 1.* U.S. Retail E-Commerce Sales as a Percent of Retail Sales

Data Retrieved from U.S. Department of Commerce (2019).

## ***1.2 The Need for Sustainable Supply Chains***

Beyond just shopping more online, consumers also prefer goods and services that are more environmentally friendly. Various environmental groups have conducted surveys and found that over half of global consumers would pay more for sustainable products (Sheffi & Blanco, 2018; Fu & Saito, 2018). In addition to customer preferences, many global initiatives are also supporting the desire for greener solutions to supply chain problems. Ten of the 17 goals for sustainable development set by the United Nations are focused on environmental sustainability (United Nations, 2019). In this context, many companies are feeling pressure from stakeholders to align with these targets, and their public commitments reflect this. For example, Amazon has recently set a target for a carbon-neutral footprint for half of its shipments by 2030 due to increased pressures from its investors and consumers (Chasan, 2019).

In order to align their business with this trend, companies that have not yet performed an accurate analysis of their carbon footprint will first need to gain visibility of their current baseline. Once this baseline is established, improvements can be made in order to be competitive

in the area of sustainability. Other trends are making the reduction of carbon footprints more challenging, such as the aforementioned growth of e-commerce. As e-commerce spending continues to grow, businesses are competing by offering faster deliveries. These fast deliveries make certain supply chain practices, such as consolidating orders in a similar location into one truck, much more challenging. Due to the conflicting pressures of having a sustainable business and delivering quickly, it is essential for businesses to understand the trade-off of making fast deliveries to meet customer demands and reducing environmental impact through consolidation.

### ***1.3 Capstone Approach***

As demand increases due to e-commerce, companies in this space need a framework to adjust their supply chains in order to remain competitive. Our capstone project provides a step-by-step approach for making these adjustments at a distribution center level. First, current-state capacity and bottleneck operations are determined at each distribution center. Then, a forecast is created for each distribution center in order to determine if current-state capacity is sufficient. Finally, labor resources within each distribution center are optimized to minimize cost while still ensuring demand can be met. In addition to providing this supply chain insight, we evaluate the resulting environmental savings of this approach.

### ***1.4 Model Application: Coppel***

Our approach was applied to Coppel, a nationwide department store in Mexico that focuses primarily on retail furniture and clothing. In 2018 Coppel was Mexico's largest retail company, possessing approximately 1,700 retail stores and 19 regional distribution centers. Coppel completed 6.3 million annual deliveries in 2018. However, it was unclear if their existing distribution network was sufficient to support the projected growth of e-commerce. Coppel therefore needed a predictive model to project growth of their markets and, if necessary, an

operational plan to increase the capacity of their distribution network. Due to the environmental pressures from stakeholders introduced in Section 1.2, Coppel aimed to design this new distribution network with not only cost in mind, but also with an accurate understanding of the environmental impact of each decision.

Integrating a forecast method, a Mixed Integer Linear Programming (MILP) model, and an environmental impact estimation, we proposed a methodology to support projected growth through the increase in capacity of Coppel's distribution centers and cross-dock platforms. Our approach provides a business with a monthly operational plan that minimizes costs while still meeting expected customer demand.

The remainder of the capstone is divided into sections that describe different aspects of our project in detail. In Chapter 2 we present existing literature that was researched to help create an appropriate methodology. Chapter 3 describes our methodologies and how each step of our approach was performed. The results and recommendations from our approach are shown in Chapter 4, and our final conclusions can be found in Chapter 5.

# CHAPTER 2: LITERATURE REVIEW

This capstone project aimed to develop a comprehensive approach that can be used to forecast demand, provide an operational plan to minimize labor costs, and grant visibility of environmental impact. Current literature explores these tasks individually as distinct, separate topics: (1) demand forecasting, (2) workforce planning, and (3) environmental impact assessment. We explored methods of demand forecasting in order to determine the best method for the case under study. Workforce planning was also reviewed because we wanted to examine existing manufacturing scheduling algorithms and create a model that uses current best practices. Finally, we performed research on environmental assessments to determine the appropriate tool to measure the environmental footprint for our project.

## ***2.1 Demand Forecasting***

In order to create a useful demand forecast, the forecast must possess the appropriate scope and method.

### *2.1.1 Scope of the Forecast*

The key to improving forecast accuracy is to choose an appropriate forecasting method that can capture the systematic behavior and project that behavior into the future (Gilliland, 2010). The first step in making this decision is determining the scope of the forecast. This means being clear about the level of aggregation, forecasting horizon, and type of product. The level of aggregation refers to period of time, item, and location (Zotteri, Kalchschmidt & Caniato, 2005).

Fildes, Mab and Kolassa (2019) stated that the level of forecast aggregation should depend on the level of decision it is being used for. For example, if a store-level decision needs to be made, then the product-level forecast is the best choice. If a larger scope, strategic decision needs to be made, then a location-level forecast is the best option. One of the challenges in the retail industry

is that there are too many variables influencing the outcome and too little information to make decisions. As a result, in order to achieve an adequate level of accuracy, a tradeoff between time and information that best fits the scope should be made.

According to Makridakis, Hyndman, and Wheelwright (1998) it is important to be aware that demand depends on two different types of events: uncontrollable external events (national economy, customers, competitors, and distributors) and controllable internal events (marketing, production, finance and company policies). A forecast is applied to try to estimate variability due to external factors. There are several methods that are divided into two major groups: the subjective and the objective. The objective methods are composed of causal models and time series. The causal model is based on the idea that there is an underlying cause that can be measured that drives the result. Some tools for causal models are econometric models, leading indicators, and input-output models. The time series is based on the observation of patterns such us level, trend, and seasonality.

The scope of our forecast for Coppel was selected to be monthly for each distribution center for both clothing and furniture. A monthly scope was selected because it aggregates demand at a level that is actionable for resource planning and it was the requested scope by Coppel.

### *2.1.2 Applicable Forecasting Methods*

When making forecasts it is important to know what exactly you are trying to forecast. According to Makridakis et al. (1998), a good way to explore data is the use of graphs such as time plots, seasonal plots, and scatterplots to identify patterns. It is important to be aware that to create meaningful graphs, a good amount of historical information is needed. The identification of patterns is key, since the identified pattern provides insight into which method of forecasting is ideal.

In time series, data could express trend, seasonality, and horizontal or cyclical patterns. Trend (b) is the rate of growth or decline of data. Seasonality (F) is a pattern that repeats at specific periods in time. The horizontal pattern (a) is the constant mean on which data fluctuates. A cyclical (C) pattern is related to the rises and falls of a business cycle. It is important to note that besides these four patterns, there is also a random fluctuation, or error (e), that accounts for the remaining variability. Cumulative, naïve, moving-average, simple exponential smoothing, Holt's, and Holt-Winter method were all researched and elaborated on in Section 2.1.3.

In addition to time series forecasting methods, machine learning algorithms have also been created for forecasting application in recent years. One example of this is the eXtreme Gradient Boosting algorithm, also known as XG Boost. XG Boost is a scalable and accurate implementation of gradient boosting machines that utilizes higher levels of computer power for boosted trees algorithms (Brownlee, 2016). XG Boost applies gradient boosting, which is an ensemble method that sequentially adds predictions and corrects previous models and fits a new model to the residuals of the previous prediction, and then minimizes the loss when adding the latest prediction (Morde, 2019). XG Boost is computationally very powerful but requires a very large amount of data to be functional. XG Boost was explored by our team, but our dataset proved insufficient in detail, volume, and the lack of external parameters when attempting to apply XG Boost. When solving similar problems that require forecasting, XG Boost should be considered and applied if sufficient data is present.

### *2.1.3 Time Series Forecast Parameters and Structure*

Makridakis et al. (1998) stated that, when using time series, demand can be predicted by combining patterns and making inferences. If it is determined that the only relevant parameter is horizontal, then the model to use will have the form $X_t = a + e_t$. If it is defined that the

horizontal pattern and trend are the relevant parameters, then the model to use will have the form $X_t = a + bt + e_t$. If the horizontal trend and seasonality are relevant parameters, then the model to use will have the form $X_t = aF_t + e_t$. Finally, if it is defined that a horizontal pattern, trend, and seasonality are all relevant parameters, then the model to use will have the form $X_t = (a + bt)F_t + e_t$.

In the cases where a horizontal pattern is the only relevant parameter, there are four models that can be used. According to Caplice (2014), the choice between these models will depend on the relevance that the most recent values have over the oldest historical values and the storage capacity of the system used. The cumulative forecast $\hat{x}_{t,t+1} = \frac{\sum_{i=1}^{t} X_i}{t}$ uses all the historical information available and gives the same weight to all values, while the naïve forecast $\hat{x}_{t,t+1} = x_t$ only uses the most recent value. In the middle of both models is the moving average forecast $\hat{x}_{t,t+1} = \frac{\sum_{i=t+1-M}^{t} X_i}{M}$ that uses only the last M number of periods. Depending on the value of M, the model will be more or less responsive to more recent information.

In the case that a horizontal pattern is the only relevant parameter, but the value of observations degrade over time, the simple exponential smoothing model $\hat{x}_{t,t+1} = \alpha x_t + (1 - \alpha)\hat{x}_{t-1,t}$ should be used, with α as the smoothing constant ranging between 0 and 1 (Caplice, 2014). As $\alpha$ grows, more weight is given to recent events. The advantage of the simple exponential smoothing model over the moving average is the use of the $\hat{x}_{t-1,t}$ parameter. This parameter allows the history of the variable to be contained in a single data point and can be very beneficial when forecasting thousands of products.

If the horizontal pattern and trend are both relevant and the value of observations degrade over time, the Holt´s method $\hat{x}_{t,t+1} = \hat{a}_t + \tau\hat{b}_t$ should be used (Caplice, 2014). Where $\tau$ represents

the number of periods into the future the forecast is predicting, $\hat{a}_t = \alpha x_t + (1-\alpha)(\hat{a}_{t-1} + \hat{b}_{t-1})$ is the estimate for the horizontal pattern, and $\hat{b}_t = \beta(\hat{a}_t - \hat{a}_{t-1}) + (1-\beta)\hat{b}_{t-1}$ is the estimate for the trend. It is important to update the value of the estimates after each iteration.

In the case that not only the horizontal, but also the seasonality parameter is relevant and the value of observations degrade over time, the double exponential smoothing method $\hat{x}_{t,t+\tau} = \hat{a}_t \hat{F}_{t+\tau-P}$ should be used (Caplice, 2014). $\tau$ represents the number of periods into the future the forecast is predicting, $P = \sum_{i=1}^{P} \hat{F}_i$ is the number of periods that have distinct seasonal differences, and $\gamma$ is the seasonality smoothing factor. It is important to update the value of the estimates after each iteration where $\hat{a}_t = \propto (\frac{x_t}{\hat{F}_{t-P}}) + (1-\alpha)(\hat{a}_{t-1})$ is the updating procedure for the horizontal pattern and $\hat{F}_t = \gamma(\frac{x_t}{\hat{a}_t}) + (1-\gamma)\hat{F}_{t-P}$ is the updating procedure for the seasonality factor.

If the horizontal pattern, the trend, and the seasonality parameters are all relevant and the value of observations degrade over time, the Holt-Winter method $\hat{x}_{t,t+\tau} = (\hat{a}_t + \tau\, \hat{b}_t)\hat{F}_{t+\tau-P}$ should be used (Caplice, 2014). Where $\tau$ once again represents the number of periods into the future the forecast is hoping to predict, $\hat{a}_t = \propto (\frac{x_t}{\hat{F}_{t-P}}) + (1-\alpha)(\hat{a}_{t-1} + \hat{b}_{t-1})$ is the updating procedure for the horizontal, $\hat{b}_t = \beta(\hat{a}_t - \hat{a}_{t-1}) + (1-\beta)\hat{b}_{t-1}$ is the estimate for the trend and $\hat{F}_t = \gamma(\frac{x_t}{\hat{a}_t}) + (1-\gamma)\hat{F}_{t-P}$ is the updating procedure for the seasonality factor.

*2.1.4 Final Forecast Selection*

When choosing a model to utilize for a forecast it is important to measure its quality, which can be done by measuring accuracy. Accuracy was measured by mean average percent error (MAPE). MAPE is the average of absolute percent error for each data point (Caplice, 2014). The

equation for MAPE is as follows, where n is number of observations, t is the time period, and e is the error:

$$MAPE = \frac{(\sum_{t=1}^{n} |e_t|)}{n} \quad (1)$$

In order to accurately predict future demand, overfitting of the forecast should be avoided, so imperfect accuracy is acceptable. A good way to avoid overfitting is by using one set of information to train an initial model and a second set to test it.

The forecasting method that is optimal for a problem will change based on how much information is available. If there is sufficient historical data and there is no causality between the factors, time series is typically the best choice. In order to select the appropriate time series method, graphing the data is very helpful to be able to detect the main parameters, such as trend and seasonality. Once relevant parameters are defined, models that may be appropriate for the forecast can be chosen and then tested. Before a final selection is made, the cost of maintaining the necessary information required to update the forecast properly should be analyzed.

### ***2.2 Workforce Planning***

Optimizing headcount per location is incredibly important in the retail industry because profit margins are lower than most other sectors, ranging from 0.5% to 3.5% (Ross, 2019). This means that being inefficient in internal processes, such as operations within distribution centers, is critical for the business because these inefficiencies could decrease an already small profit margin. Due to the presence of holidays throughout the year, the retail industry has a strong trend and seasonal pattern, and therefore there is high variability in terms of resource requirements. As indicated by Cox (1989), a high level of flexibility in processing capacity is a major factor for success in this type of scenario. This flexibility equates to the quickness and ease in which facilities can respond to changes in demand.

Oke's (2000) research indicates that increasing human resource flexibility is pivotal in remaining competitive in this industry. However, not many firms improve this flexibility due to the challenges of balancing full-time workers, overtime, part-time workers, and contractors. Within the manufacturing industry; Holt, Modigliani, Muth, and Simon (1960) pioneered solutions by proposing a production model to adjust workforce availability within the business constraints of demand and cost.

There has been ample research completed on workforce scheduling using MILP models where researchers consider overtime, hiring, and firing costs. However, the majority of these models are focused in a production setting. According to Pinedo (2005), there are three main differences between manufacturing and services settings when planning and scheduling. The first difference is that in manufacturing it is usually possible to keep a stock of goods, whereas in services that is not possible. The second difference in manufacturing is the number of resources (which are typically machines) is usually fixed in the short term, whereas in services the number of resources is more flexible. The third difference is that in services it is more common to deny service to a customer is a more common practice than not delivering a product to a customer in a manufacturing setting. The aforementioned differences between manufacturing and services affect the processing restrictions, constraints, and objective of a MILP. Hence, service models tend to be very different from manufacturing models.

In the case of MILP models focused on services, Hung (1999) has shown how to schedule workers throughout the year under a practical annualized hours (AH) scenario. Similar studies on annualized hours scenarios were conducted by Clutterbuck (1982), Lynch (1985), Curran (1992), MacMeeking (1995), Mazur (1995) and Corominas and Pastor (2000). The AH concept gives the employer a certain number of labor hours available in a year that can be allocated according to

manpower as needed, which is beneficial to successfully tackle seasonal demand. However, AH cannot be applied in every companies due to company's cultural policies specific or specific country labor regulations. Therefore, we have built a MILP model that combines the knowledge from manufacturing and AH models.

To summarize, the second goal of our research was to develop a MILP model that meets the requirements of the retail industry as a service. To apply this MILP model to Coppel's network, we tailored it to Mexican regulations by adding shift constraints and the appropriate overtime rates.

### ***2.3 Environmental Impact Assessment***

According to the Intergovernmental Panel on Climate Change (2013), the climate is changing because greenhouse gases (GHG) have increased their concentration in the atmosphere. This has been caused by human activity like industrialization and changes in agriculture and land use. The main greenhouse gases are carbon dioxide, methane, nitrous oxide, hydrofluorocarbons, perfluorocarbons, and sulfur hexafluoride. The IPCC (2007) also provided a recommended metric to compare the impact of different types of emissions is global warming potential (GWP). GWP converts an impact to the equivalent amount of $CO_2$ trapped in the atmosphere that would cause the same impact. Therefore, by using the GWP conversion, we can aggregate GHG emissions into a single metric, carbon dioxide equivalent (CO2e).

Carbon equivalents has become the most widely used method to measure environmental footprint (Minx et al, 2009). Supporting this, government regulation reports and taxes are calculated based on carbon emissions (Bouchery, Corbett, Fransoo & Tan, 2017). Currently many companies report their emissions in order to be prepared for future climate policies

(Carbon Trust, 2014). Similarly, many companies report their emissions in the form of carbon emissions because of the pressure exerted by the public, previously stated in Section 1.3. Regarding measuring the carbon footprint, Bouchery, Corbett, Fransoo and Tan (2017) stated that the Greenhouse Gas (GHG) Protocol is the most used framework to account for carbon emissions. The GHG Protocol released in 2001 had as a goal to provide guidance for companies in preparing a carbon emission inventory (WRI and WBCSD, 2004). The GHG Protocol also provides recommendations on how to set organizational and operational boundaries on emissions. Within the organizational boundaries there are two distinct approaches: (A) the equity share approach and (B) the control approach. Within the operational boundaries the GHG Protocol distinguishes three scopes. Scope 1 emissions are related to the direct emissions of the company by sources that they own. Scope 2 accounts for the emissions generated by the source of purchased electricity consumed by the company. Finally, Scope 3 encompasses all the indirect emissions that are not accounted in Scope 1 and 2 (Bouchery, Corbett, Fransoo & Tan, 2017). iN our project, we will only focus on the measurement of Scope 1 and 2, specifically on the electricity consumption of distribution centers. The best practices that were gathered from this literature review were used to create our methodologies, which are presented in the next chapter.

# CHAPTER 3: METHODOLOGY

Our approach consists of four distinct steps, shown in Figure 2: They are to identify current-state capacity of an existing supply chain, predict future demand in an industry that experiences trend and seasonality, optimize headcount by minimizing cost while still meeting customer demand, and provide visibility of the environmental impact of any changes brought by the previous three steps. Before any of these steps can be completed, numerous current-state, company-specific inputs are required. An understanding of current headcount, the time required for each task, historical demand, and various operational costs are necessary for effective application. The outputs of our approach are current-state capacity and operational bottlenecks, a demand forecast for the next year, an optimized plan for labor resource allocation, and environmental footprint visibility. Due to the differences in operation cycle time and independent workforce for each product type, our approach is applied to clothing and furniture separately.

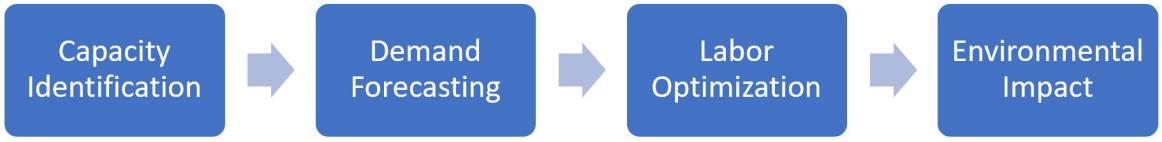



*Figure 2.* Step-by-Step Methodology

## ***3.1 Current-state Capacity***

It is necessary to understand the current state of a supply chain before any optimization or possible expansion is explored. Without understanding if a business' current assets are sufficient to meet its demand, decisions for the future would be uninformed and most likely incorrect. Our model focuses on the labor at Coppel distribution centers, and therefore our first step was to determine their baseline.

In applying our approach to Coppel, our team looked at warehouse operations including picking, transferring, loading, and then driving, shown in Figure 3. These operations are identical for both clothing and furniture. The worker who performs the picking operation, referred to as Worker 1, removes the required items from their storage location and prepares them for transportation. The transfer operation is completed by another individual, Worker 2, who moves the material to the loading station. Worker 3 completes the loading step, which consists of loading the desired goods into a truck. Worker 4, the driver, also assists in the loading process and then completes the delivery. Our capstone does not focus on the unload and storage steps because these tasks typically do not occur in the same month and were deemed out of scope. Workforces are independent for clothing and furniture besides for Worker 3 and 4.

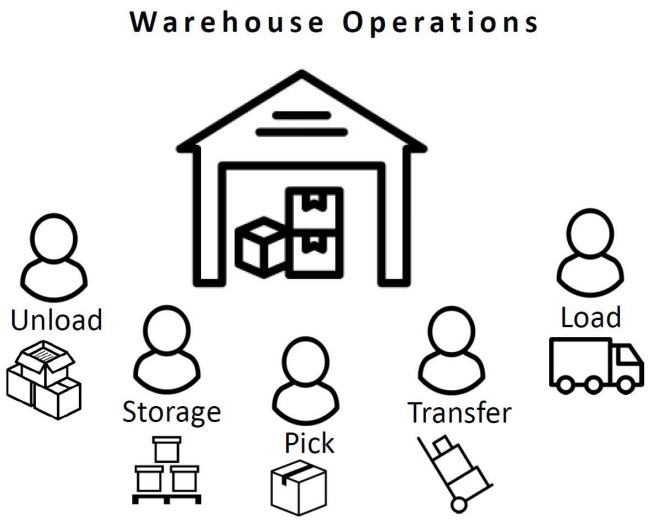



*Figure 3.* Coppel Warehouse Operations

Headcount data was provided by the Coppel team for each operation and product at each distribution center in the Coppel network. The average cycle time[2] of each operation was

[2] Cycle Time refers to the total time from the beginning to the end of a process, including any process and delay time (iSixSigma, 2020)

required to be able to calculate the output capacity of each distribution center and therefore the entire Coppel network. To gather this cycle time data, a time study was completed at eight different distribution centers over the course of three weeks for both clothing and furniture operations. The data were aggregated and averaged to calculate the cycle time per operation. Variance between each distribution center and within each process was calculated as well to better understand how standardized each process was. Next, the product of the average cycle time per transaction and the available labor hours of each distribution center was calculated to understand the theoretical number of units that each distribution center could process. This information allowed for the identification of potential bottlenecks within each distribution center and the calculation of the entire network's capacity. Current-state capacity was calculated using the following equation:

$$Capacity = MIN[\frac{Available\ Time_{operation}}{Cycle\ Time_{operation}}] (2)$$

### ***3.2 Forecasting Methodology***

A forecast of demand for the next year of both clothing and furniture was required to determine if current-state capacity at each distribution center was sufficient. There are many ways that forecasting can be performed, and to determine that the Holt-Winter Method fits our data set most appropriately of the methods discussed in Section 2.1, we used mean average percent error (MAPE) to evaluate forecast accuracy.

Coppel provided three years of data for furniture and clothing sales. 2017 and 2018 were used as training data for our forecasting model, while 2019 was to test the forecast. Using the Python coding language, a general-purpose programming language (Python Software Foundation, 2020), Coppel sales data was aggregated by units sold per month at each DC.

After plotting the historical data, it was clear that there was both trend and seasonality present, as shown in Figure 4.

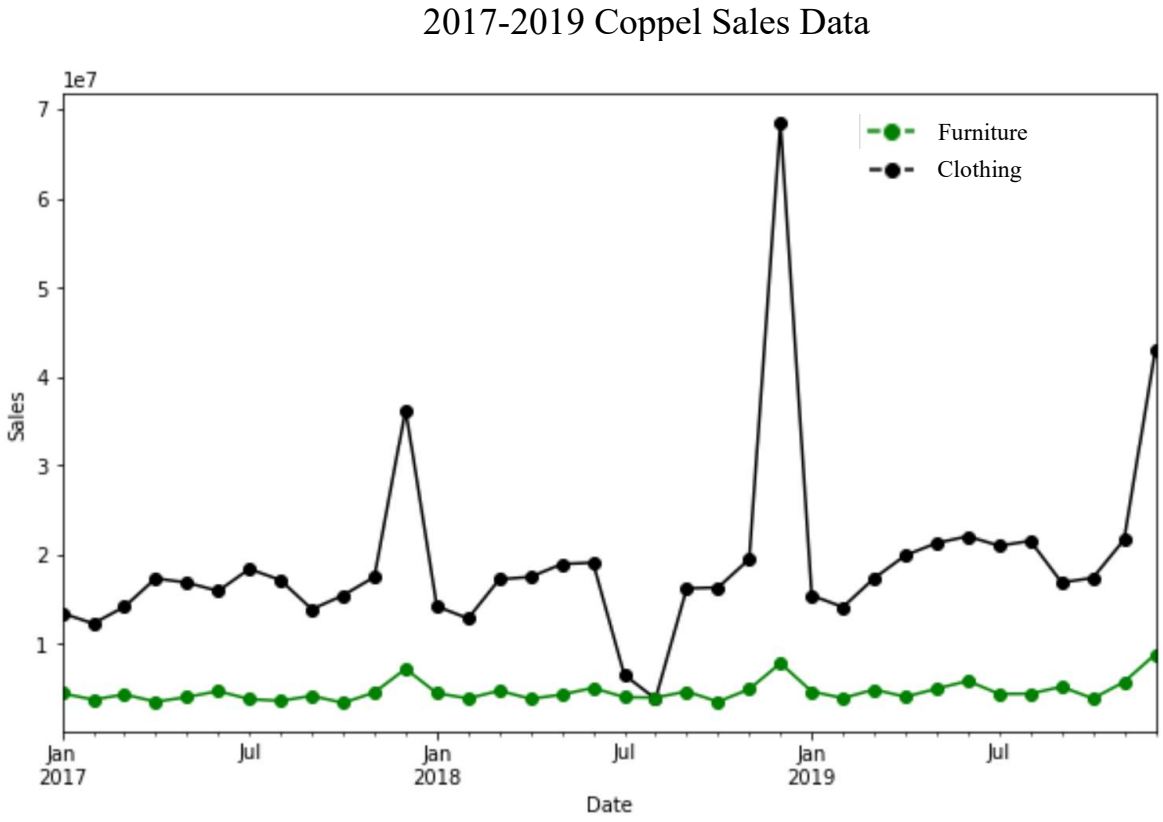



*Figure 4.* Historical Coppel Sales from 2017 to 2019

### *3.2.1 Holt-Winter Model*

The Holt-Winter Model, described in Section 2.1, assumes a linear trend with a multiplicative seasonality effect over both level and trend. The level estimate is made with "de-seasoned" values of the latest observation, and the trend is estimated in the same way as the Holt method, which uses two variables (Alpha and Beta) to value historical data and smooth past data. The seasonality estimate functions the same way as the Double Exponential Smoothing model, which functions by applying a seasonality factor that is used to influence the forecast and is continually updated based on newly observed data. The

equation for the Holt-Winter Model is $\hat{x}_{t,t+\tau} = (\hat{a}_t + \tau\,\hat{b}_t)\hat{F}_{t+\tau-P}$ , which is elaborated on in detail in Section 2.1.

Based on our research and the available data provided to us, Holt-Winter was assumed to be the best forecasting methodology for our project. To confirm this assumption, Naïve, Moving-Average, Simple Exponential Smoothing, Holt's Method, and Holt-Winter Methods were all tested, and Holt-Winter had the lowest mean average percent error (MAPE). Combinations of additive and multiplicative trend and seasonality were all calculated to determine the best fit. This fit was also judged by MAPE, which can be found in Chapter 4, Results and Discussion. This Holt-Winter Model was applied using the Statsmodel package in Python. Statsmodel is a Python module that provides functions for the estimation of different statistical models and data exploration (Seabold, Skipper, & Perktold, 2010). The script used to create this forecast can be found in Appendix D.

### ***3.3 Labor Optimization***

The results of the time study and forecast gave our team information regarding Coppel's current-state capacity and estimated demand per month. We created an optimization model that takes capacity, demand, and other operational costs as inputs. As mentioned before, the model's objective is to minimize total cost while following local regulations that act as constraints, all while still meeting customer demand. The model provides an output of operational decisions, such as hiring workers, firing workers, running overtime, transferring workers, and using external contractors. The output provides these recommendations for the next 12 months. Due to independent labor forces and different operation cycle times, the model is applied separately to furniture and clothing. The mathematical model was created and run using the Gurobi Solver in the Python coding language, using the code

presented in Appendix C. Gurobi Solver is a commercial optimization solver for mixed-integer linear programming and other optimization models (Gurobi, 2020).

Our optimization function minimizes total cost for Coppel while ensuring that total demand is satisfied and regional overtime regulations are adhered to. Coppel's team provided the costs for hiring, firing, and transferring workers from one role to another. The outputs of this model is a table that describes what operational decisions should be made at each time period to minimize total cost for clothing and furniture. The logic of the MILP model is described in Section 3.3.1.

### ***3.3.1 MILP Model***

In order to minimize operational costs for each distribution center we created a Mixed-Integer Linear Program (MILP).

Problem formulation

**Sets:**

i, j = operations {1=Picking, 2=Transfer, 3=Load and 4=Drivers) }

t =months {1- 12}

**Parameters:**

$\mathbf{D_t}$ = Demand of units to process in month *t*.

$\mathbf{L_i}$ = Time required to process one unit at station I (hours/unit)

$\mathbf{K}$ = Number of monthly hours an employee works according to shift (hours/person)

$\mathbf{M^O}$ = Maximum hours of overtime type 1 allowed per employee per month

$\mathbf{M^Q}$ represent the maximum hours of overtime type 2 allowed per employee per month.

$\boldsymbol{W_{i0}}$ = Size of the workforce at the beginning of the planning period (month 0).

$\boldsymbol{c_i^W}$ = Cost per employee ($/person/month)

$\boldsymbol{c_i^F}$ = Cost of firing one employee ($/person)

$\boldsymbol{c_i^H}$ = Cost of hiring one employee ($/person)

$\boldsymbol{c_{ij}^X}$ = Cost of transferring one employee from station i to station j ($/person)

$\boldsymbol{c_i^O}$ = Cost of overtime type 1 ($/hour)

$\boldsymbol{c_i^Q}$ = Cost of overtime type 2 ($/hour)

$\boldsymbol{c_i^C}$= Cost of outsourcing the processing of units to a third party ($/unit)

$\boldsymbol{H_{it}}$ = Number of employees type *i* to hire at start of month *t*.

$\boldsymbol{F_{it}}$ = Number of employees type *i* to fire at start of month *t*.

$\boldsymbol{W_{it}}$= Number of employees type *i* in month *t*.

$\boldsymbol{X_{ijt}}$ = Number of employees type *i* turning into type *i* in month *t*.

$\boldsymbol{O_{it}}$ = Number of overtime type1 hours to work in month *t*.

$\boldsymbol{Q_{it}}$ = Number of overtime type2 hours to work in month *t*.

$\boldsymbol{P_t}$ = Number of internal units to process in month *t*.

$\boldsymbol{C_t}$ = Number of outsourced units to process in month *t*.

$H_{it}, F_{it}, W_{it}, X_{ijt}, O_{it}, Q_{it}, P_t, C_t \geq 0, integer$

The mathematical model is presented next:

$$Min\ z = \sum_{t=1}^{T} \sum_{i=1}^{I} c_i^W \boldsymbol{W_{it}} + c_i^H \boldsymbol{H_{it}} + c_{ij}^X \boldsymbol{X_{ijt}} + c_i^F \boldsymbol{F_{it}} + c_i^O \boldsymbol{O_{it}} + c_i^Q \boldsymbol{Q_{it}} + c_i^C \boldsymbol{C_{it}} \quad (3)$$

**Subject to**:

$$D_{it} - P_{it} - C_{it} = 0, \forall\ i\ \in \mathrm{I}, \forall\ t\ \in \mathrm{T} \quad (4)$$

$$P_{it} - \left(\frac{K\, W_{it} + O_{it} + Q_{it}}{L_i}\right) \leq 0, \forall\ i\ \in \mathrm{I}, \forall\ t\ \in \mathrm{T} \quad (5)$$

$$W_{it} - W_{it-1} - H_{it} + F_{it} = 0, \forall \in \mathrm{I}, t = 1, 2 \quad (6)$$

$$W_{3t} - W_{3t-1} - H_{3t} + F_{3t} + X_{3t} = 0, \forall\, i \in \mathrm{I}, t = 3 \quad (7)$$

$$W_{4t} - W_{4t-1} - H_{4t} + F_{4t} - X_{3t} = 0, \forall\, i \in \mathrm{I}, t = 4 \quad (8)$$

$$O_{it} - \mathrm{M}^{\mathrm{O}} W_{it} \leq 0, \forall\, i \in \mathrm{I}, \forall\, t \in \mathrm{T} \quad (9)$$

$$Q_{it} - \mathrm{M}^{\mathrm{Q}} W_{it} \leq 0, \forall\, i \in \mathrm{I}, \forall\, t \in \mathrm{T} \quad (10)$$

Constraint (4) defines that demand has to be met by the sum of internal production $\boldsymbol{P_{it}}$ and a third party's production $\boldsymbol{C_{it}}$.

Constraint (5) defines that internal production should be met by the sum of current month worker's regular hours and extra hours (type 1 and 2).

Constraints (6), (7) and (8) define the balance in number of workers for the four types of employees. Workers at the end of period $\boldsymbol{W_{it}}$ is equal to the workers at the beginning of period $\boldsymbol{W_{it-1}}$ plus the numbers of workers hired in the period $\boldsymbol{H_{it}}$ minus the number of workers fired in the period $\boldsymbol{F_{it}}$.

Constraints (9) and (10) define a limit of the total amount of extra hours by multiplying the maximum amount of individual extra hours ($\mathbf{M^O}$ *and* $\mathbf{M^Q}$) by the number of available workers $\boldsymbol{W_{it}}$ .

### ***3.4 Carbon Footprint Calculation***

Due to the pressures being placed on companies to be environmentally conscious, as described in Section 1.2, the environmental impact of each supply decision should be quantified. In our application of our approach to Coppel, we will measure the environmental footprint for the evaluated facilities and how our optimization model will impact this footprint. Coppel has used the Greenhouse Gas Protocol (GHG) to quantify this impact, which is described in Section 2.3.

Our environmental assessment focuses on emissions that would be affected by our forecast and optimization model, which are the physical Coppel facilities. These facilities contain primarily Scope 2 emissions in the form of electricity consumption. Our team estimated future emissions based on changes in area of Coppel's facilities, as well as any changes to the hours of operation of these facilities. The Scope 2 emissions of Coppel's facilities in 2019 were of 744.74 tons of CO2e. The average Coppel store is approximately 1,000 square meters and works for 67 hours a week (Figure 5 shows an example of the store dimensions). The average Coppel distribution center is approximately 15,000 square meters and works for 50 hours a week (Figure 6 shows an example of the DC dimensions).

In 2019 there were 24 distribution centers and 3,146 stores. Table 1 shows the calculations of yearly emissions per type of facility. Using total 2019 Scope 2 emissions and the combined area of Coppel facilities, an average emission rate of 0.000939 tons of CO2e per Distribution Center per hour was found, described in Table 2. This value was used to calculate the effects of any expansion by Coppel in future years. We also found that $Emissions_{DC2019}$ = 58.59 tCO2e.

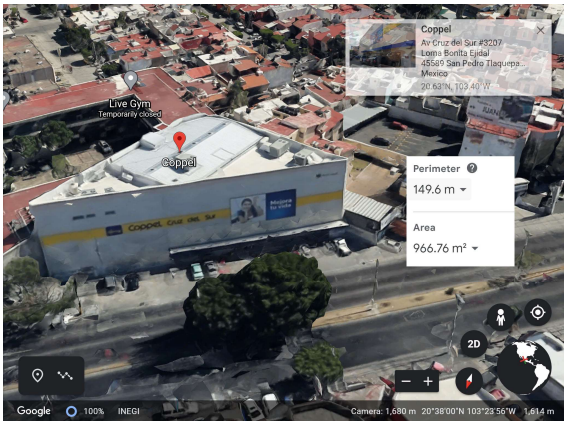



*Figure 5.* Average Coppel store's dimensions
Retrieved from Google Earth (Google Earth, 2020a)

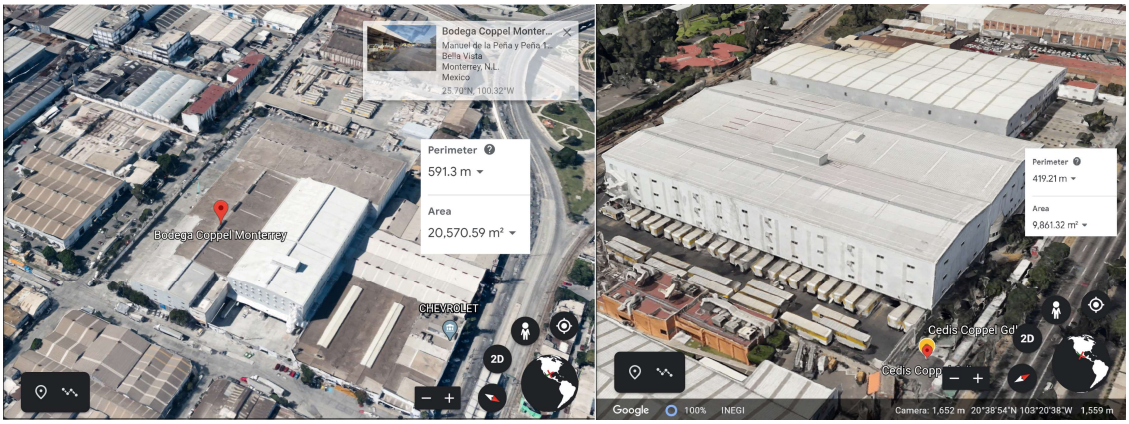



*Figure 6.* Average Coppel DC's dimension
Retrieved from Google Earth (Google Earth, 2020b)

**Table 1.**
*Scope 2 emissions per type of facility*

| Building Type | Distribution Center | Store |
|---|---|---|
| Count | 24 | 3,146 |
| Area (m2) | 15,000 | 1,000 |
| Total Area | 360,000 | 3,146,000 |
| Average working Hours per year | 2,600 | 3,484 |
| Total Hours worked x Area | 936,000,000 | 10,960,664,000 |
| % Consumption | 8% | 92% |
| Total Emissions (tCO2e) | 58.59 | 686.15 |

**Table 2.**
*Scope 2 emission per hour*

| Building Type | Distribution Center | Store |
|---|---|---|
| Average emissions per year (tCO2e) | 2.44 | 0.22 |
| Average emissions per hour (tCO2e) | 0.000939 | 0.000063 |

Additionally, the output of our optimization model could suggest running overtime production for cases where hiring and firing costs are higher than offering existing employees overtime rates. Each hour of overtime outside of typical business hours would increase the environmental footprint of Coppel due to the need to keep facilities running and utilizing power. Each DC under current conditions operates one shift for 52 weeks, which is equivalent to 2,496 hours per year. The impact on Scope 2 emissions for DCs was calculated using the equation:

$$Emissions_{DC2020} = 0.000939\ (Count_{DC} * 2496) + Total\ OT\ Hours \ldots (11)$$

Using this equation 11, we are able to measure the impact of running the DCs for longer hours. For example, if Operation 1 was running 8 hours of overtime and Operation 2 was running 12 hours, 12 hours would contribute to the sum of total overtime since both Operations 1 and 2 could be run in the facility at the same time.

In the next chapter, we discuss and analyze the results of each methodology, as well as provide recommendations for Coppel.

# CHAPTER 4: RESULTS AND DISCUSSION

Utilizing the capacity equation, demand forecasting method, optimization model, and environmental impact calculation discussed in Chapter 3, we applied the four steps of our approach to Coppel's data. When analyzing our results, we found that each step's result yielded additional recommendations for the Coppel team. Current state capacity calculations identified process bottlenecks that can be targeted for future improvement projects. The created forecast can be used to make more informed business decisions and serve as an input for our optimization model. The optimization model provides recommended operational decisions for each month in order to minimize labor costs at each distribution center. Finally, the environmental impact calculation provides Coppel with visibility of the emissions impact of each decision our optimization model recommends.

## ***4.1 Current State Capacity***

The time study performed at various Coppel distribution centers was performed over three weeks at eight different distribution centers. Using this data, the average cycle time per unit for each process was calculated, which is shown in Table 3. The operations in order from shortest to longest cycle time were Load, Deliver, Pick, and then Transfer. The variation of these time observations was also calculated in order to better grasp how cycle time could vary across Coppel, with a snapshot shown in Figure 7.

***Table 3.***
*Average operation cycle time (secs) across all DCs*

| Operation | Picking | Transfer | Load | Deliver |
|---|---|---|---|---|
| Clothing (secs/unit) | 44 | 179 | 20 | 20 |
| Furniture (secs/unit) | 161 | 243 | 25 | 25 |

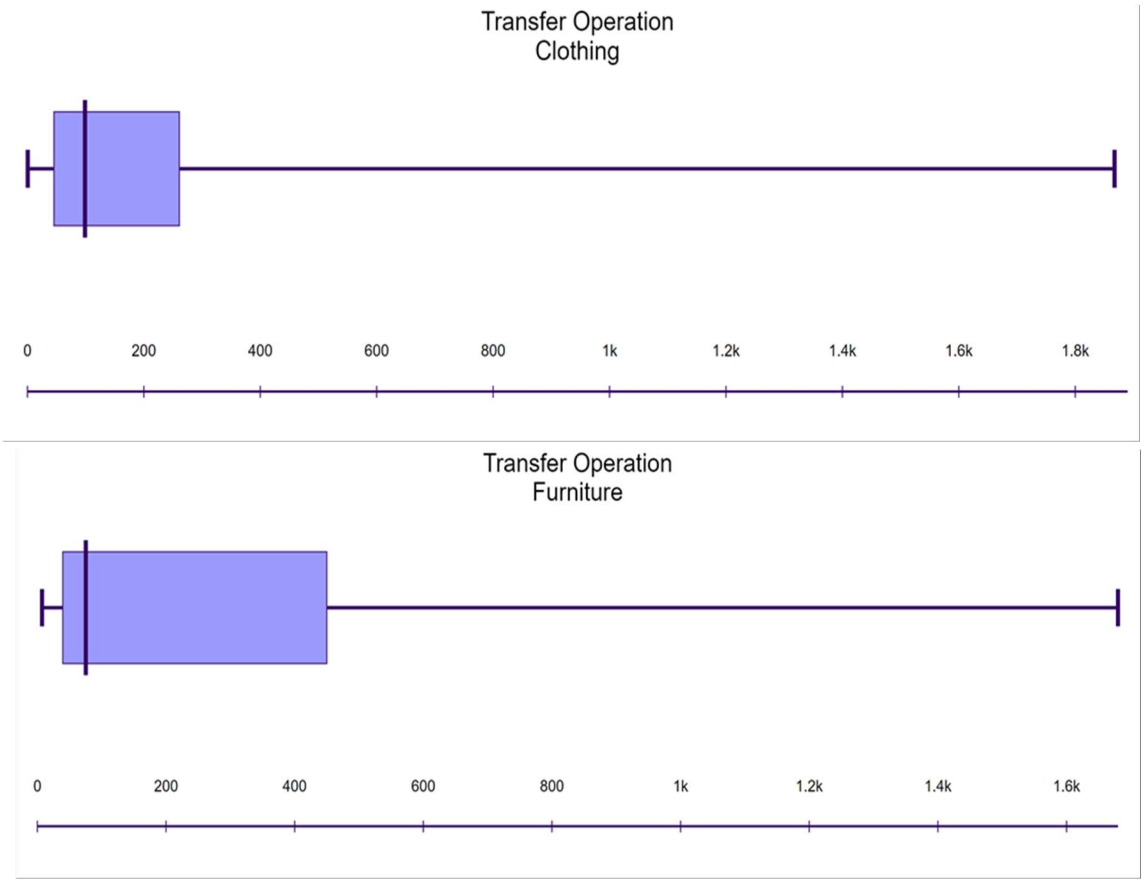



*Figure 7.* Box Plot for all Transfer Time Study observations for clothing and furniture (seconds/unit/employee)

In addition to observing variance across all of Coppel, we also compared the variation between distribution centers for each operation. Figure 8 shows a snapshot of significant variation between DCs for the picking operation of clothing.

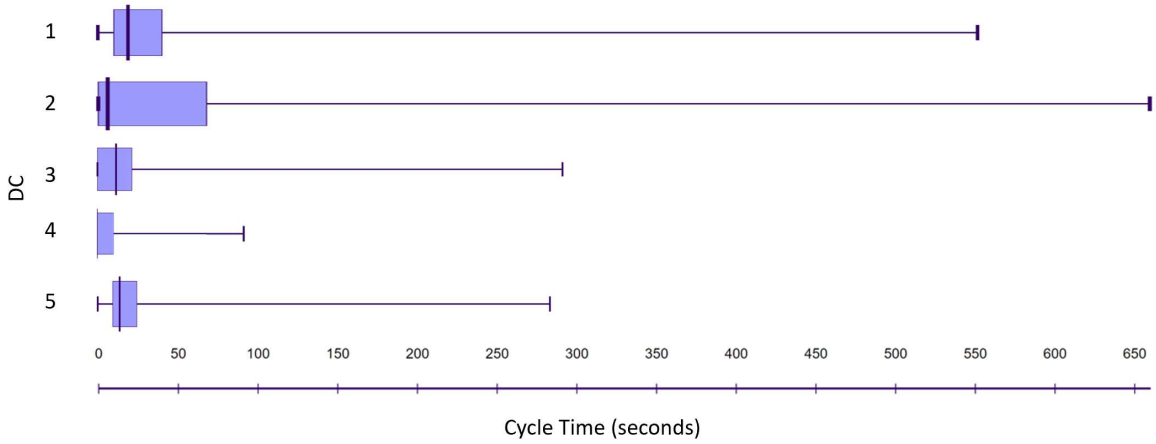



*Figure 8.* Box plot of pick operation cycle for clothing time at five distribution centers (seconds)

One distribution center was observed to have an average picking operation cycle time for clothing of more than double the average among the other distribution centers. There is currently a significant gap between how long each task takes at each DC. It would require a deeper dive into each individual process in order to identify the root cause, but on the surface the large variation within operations and between distribution centers appears to be an opportunity for best practice sharing. The rest of the time study results can be found in Appendix E. Due to the scope of this project being focused on resource management and not shop operations improvement, the average cycle times for each operation were applied to the remaining distribution centers. These values, along with the provided headcount data for each distribution center, were used to calculate the current state capacity for each DC, shown in Figure 9 for both clothing and furniture.

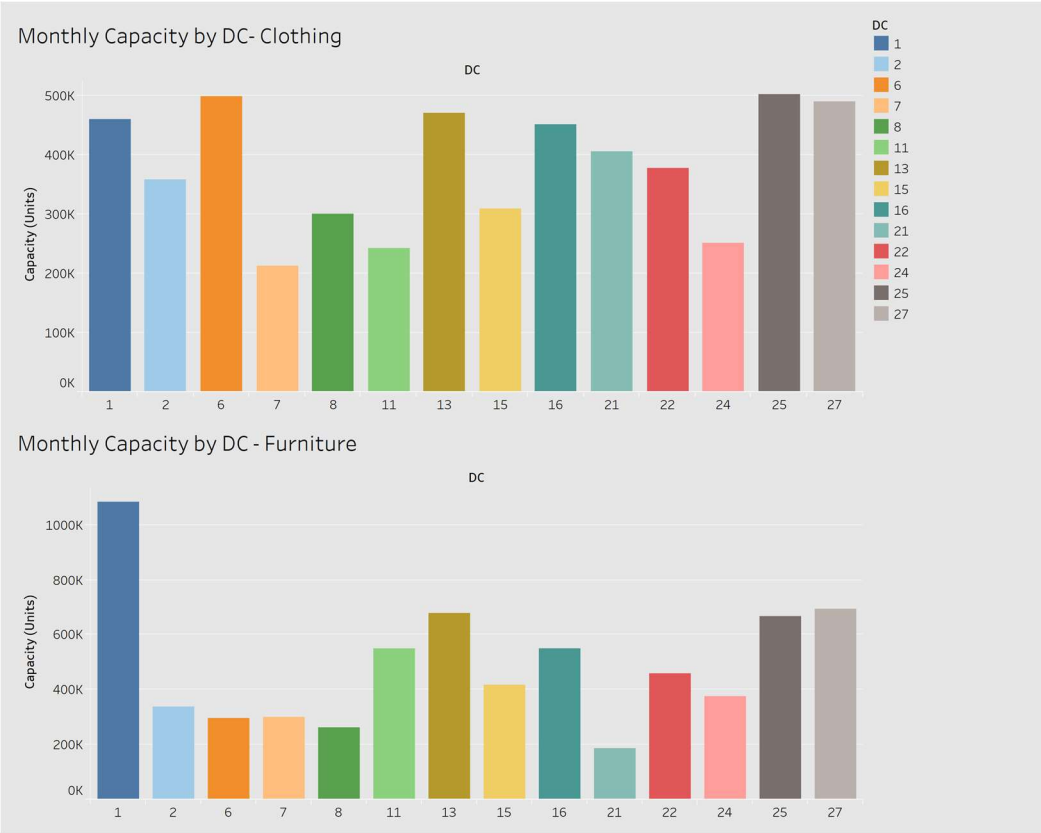



*Figure 9.* Distribution Center Capacity (units/year)

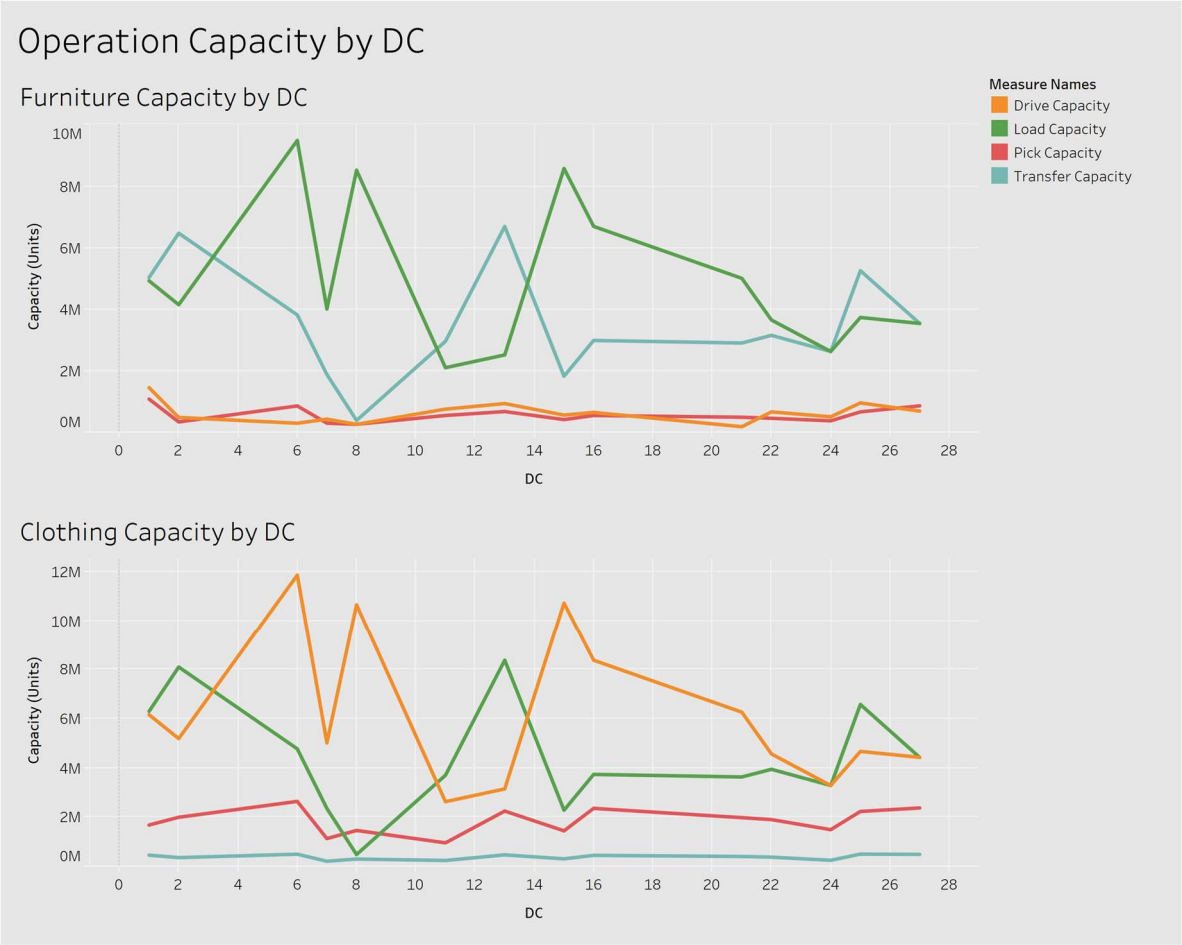



*Figure 10.* Operation Capacity by DC

The capacity for each operation was calculated and the minimum capacity operation at each distribution center was identified as the bottleneck. There was a large gap between capacities of different operations within a DC, shown in Figure 10, which appears to be a waste since capacity is maxed out by the bottleneck operation. The breakdown of which operations were the bottleneck at each distribution center can be found in Figures 11 and 12 for clothing and furniture, respectively. The transfer operation was the bottleneck process for the majority of DCs for clothing operations. This information provides the recommendation that if capacity improvements are desired for clothing, the cycle time of the transfer process should be reduced through improvement projects or headcount should be increased. Additionally, improvements to other processes or headcount additions would

not improve overall capacity of the distribution center and potential savings could occur by reducing headcount of non-bottleneck operations.

For furniture operations, the bottlenecks follow a similar trend of the transfer operation being the bottleneck for the majority of DCs, but some experience the picking operation as their bottleneck, shown in Figure 12. This data recommends that each distribution center should focus on its own bottleneck operation, as there is not an overwhelmingly common bottleneck that can be targeted. For both clothing and furniture, there could be potential savings or additional productivity by reallocating headcount to match capacity of the operations.

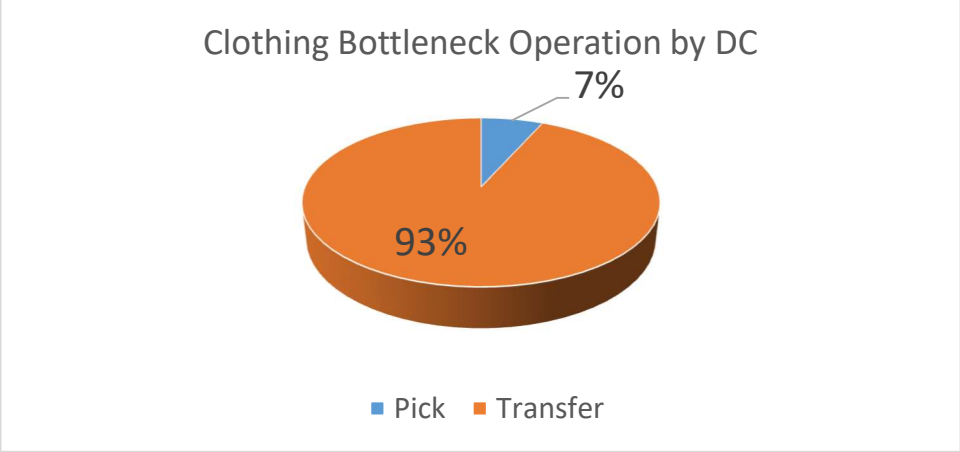



*Figure 11.* Bottleneck Operation by DC Pie Chart - Clothing

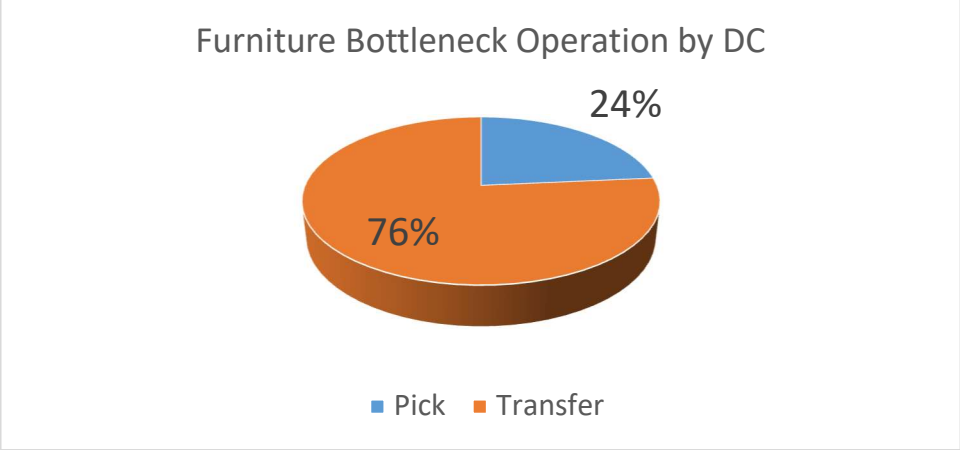



*Figure 12.* Bottleneck Operation by DC Pie Chart - Furniture

## *4.2 Forecast Results*

The Coppel team provided sales data for furniture and clothing from 2017 to 2019. 2017 and 2018 were used as training data, while 2019 was used to test the forecast. After aggregating sales data by each distribution center, the Holt-Winter method was applied in Python on the training data. This resulted in a monthly forecast for each distribution center for both furniture and clothing. An example of this is shown in Figure 13.

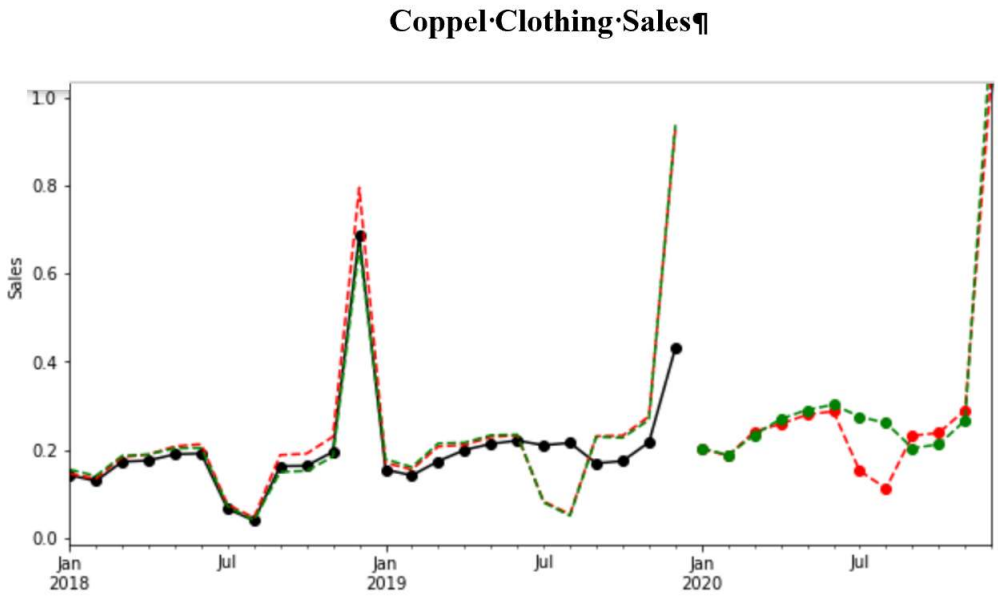



*Figure 13.* Coppel Clothing Sales from 2018 to 2019 and Forecast 2020 (Fit 1 and Fit 2)

A snapshot of the monthly forecast by distribution center is shown in Figure 14 and the entire forecast can be found in Appendix A. This forecast was used as an input for the optimization model we created in order to minimize operational costs at each distribution center.

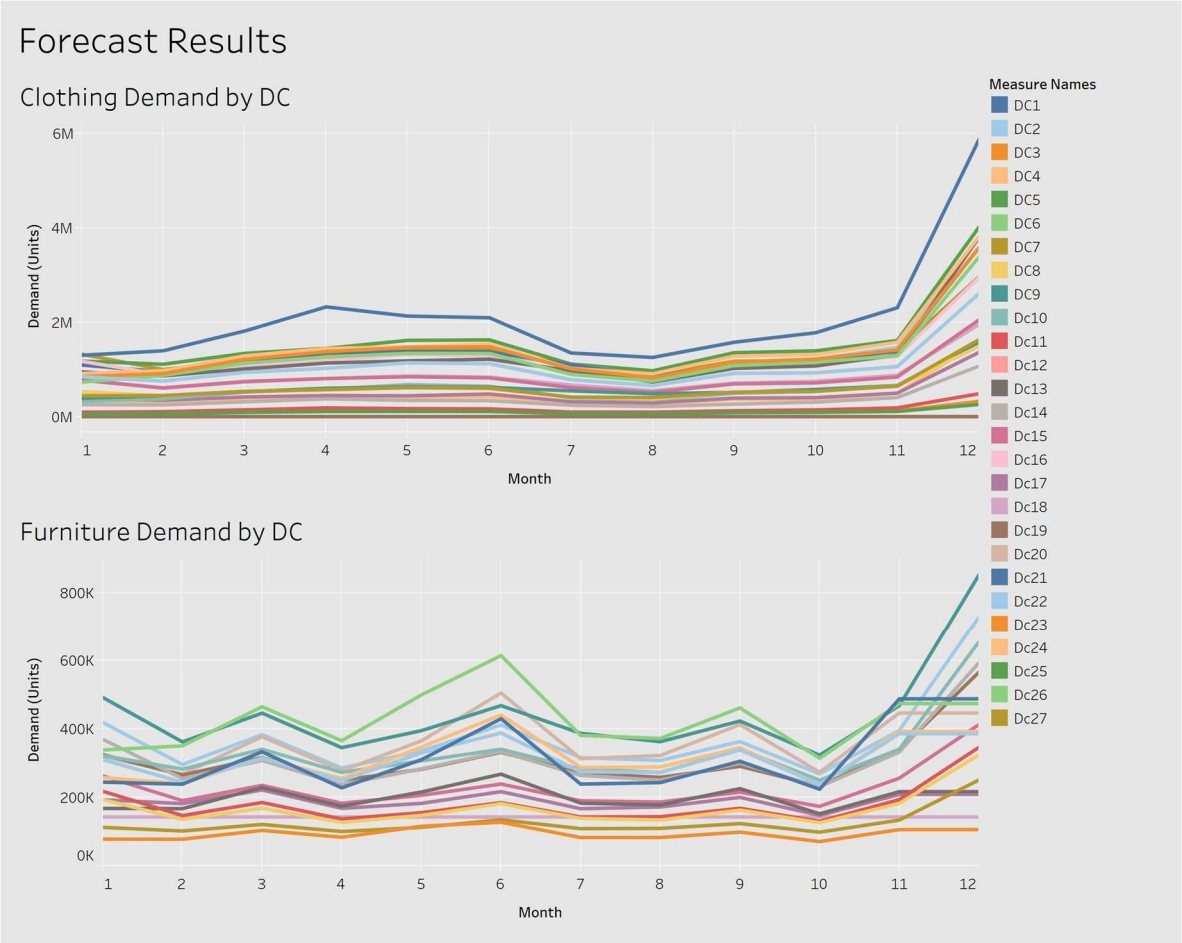



*Figure 14.* Distribution Center Monthly Forecast (Units/Month)

Additionally, the forecast provides data that can be used to form recommendations for other aspects of the company, such as how inventory should be managed. Four different forecasts were created, and their accuracies were judged based on the fit to the test data of 2019. Fit 1 consisted of additive trend and additive seasonality and Fit 2 was additive trend and multiplicative seasonality. Fit 3 and Fit 4 were the same as Fit 1 and 2, but a dampener was applied to the trend pattern. A dampener is used when trend is not expected to persist at the same rate year over year, so the previous year's trend is reduced. Fit 2 provided the highest accuracy when judged by mean average percent error, with a value of 17.74, shown in Table 4. This forecast was created for the next twelve months, but can be easily adjusted to

project farther into the future. The forecast can be rolled out further in the future as new months of sales data are collected that can be input into the Python file.

**Table 4**.
*Mean Average Percent Error per Forecast*

| Fit 1 MAPE | Fit 2 MAPE | Fit 3 MAPE | Fit 4 MAPE |
|---|---|---|---|
| 19.17 | 17.74 | 25.15 | 25.19 |


*Figure 15.* Annual Demand versus Capacity at each DC (Units)

Using the calculated bottlenecks and capacities for each DC from Section 4.1, it was determined that 100% of DCs cannot accommodate their forecasted annual demand for clothing, and 26% of DCs cannot handle the annual demand forecasted for furniture. Figure 15 displays annual capacity versus demand for each DC. With current-state conditions, 55% of clothing demand and 36% of furniture demand would be unfulfilled. Of the 74% of

DCs that can handle furniture demand, they have excess capacity to produce ~80% more than is necessary. This proves that current-state conditions are insufficient to accommodate forecasted demand. Changes in resource allocation through the implementation of the next step our approach, headcount optimization, can ensure that these demand misses will not occur at such a high magnitude moving forward.

### ***4.3 Labor Optimization***

The distribution center forecast, current state capacity, and operational costs (monthly salary, hiring cost, firing cost, transferring cost and outsourcing cost) were used as inputs in our optimization function that is solved using the Gurobi Solver in Python. The outputs of the model is a table that describes operational decisions at each time period in order to minimize costs, with an example shown in Table 5. Operation 1 refers to the picking operation, operation 2 is for the transfer operation, operation 3 is for loading, and operation 4 is the driving operation. The other outputs follow the same pattern and represent different decisions. “Hire” represents the number of employees hired, “Fire” depicts the number of employees fired, “OT_1” represents the number shifts of overtime allocated, “OT_2” represents the number of shifts of overtime allocated after the first restriction is met (in this case, OT_1 is double the normal salary, where OT_2 is triple the normal salary). “Transfer” refers to any employees that would be changed from one operation to another, and “ContractUnits” refers to the number of units contracted out to an external company. The rows labeled as “Workers” display what the existing headcount is for that operation at each time period. Shown in Table 5, if any action is recommended, the variable “Hire,” “Fire,” “OT_1,” “OT_2,” or “ContractUnits” will be displayed with the appropriate value. In this case, existing capacity and headcount were too large and headcount was reduced after month 1. Due to the cost of overtime versus the hiring costs, the model elected to run

overtime instead of hiring additional workers because it helped reduce overall labor cost. Because workers three and four are shared between furniture and clothing, outputs of headcount for each operation are shown in the "Value_Total" column.

**Table 5.**
*Distribution Center 2, Operation 2, Operational Decision Matrix to Minimize Cost*

| Month | DC | Operation | Variable | Value_Clothing | Value_Furniture | Value_Total |
|---|---|---|---|---|---|---|
| 1 | 2 | 1 | Workers | 127 | 113 | 240 |
| 1 | 2 | 1 | Fire | 45 | 34 | 79 |
| 2 | 2 | 1 | Workers | 82 | 79 | 161 |
| 3 | 2 | 1 | Workers | 82 | 79 | 161 |
| 4 | 2 | 1 | Workers | 82 | 79 | 161 |
| 5 | 2 | 1 | Workers | 82 | 79 | 161 |
| 6 | 2 | 1 | OT_1 | 0 | 11 | 11 |
| 6 | 2 | 1 | Workers | 82 | 79 | 161 |
| 7 | 2 | 1 | Workers | 82 | 79 | 161 |
| 8 | 2 | 1 | Workers | 82 | 79 | 161 |
| 9 | 2 | 1 | Workers | 82 | 79 | 161 |
| 10 | 2 | 1 | Workers | 82 | 79 | 161 |
| 11 | 2 | 1 | OT_1 | 0 | 10 | 10 |
| 11 | 2 | 1 | Workers | 82 | 79 | 161 |
| 12 | 2 | 1 | OT_1 | 15 | 10 | 25 |
| 12 | 2 | 1 | OT_2 | 68 | 0 | 68 |
| 12 | 2 | 1 | Workers | 82 | 79 | 161 |

It was observed that increases and decreases in monthly demand did not typically result in drastic headcount adjustment, as the hiring and firing costs required to fluctuate capacity were higher than keeping the extra workforce on payroll, with an example shown in Figure 16. In most cases, a capacity that was higher than the expected monthly demand for the majority of the year was selected and headcount was kept relatively constant. Overtime was utilized in these cases to help accommodate seasonal jumps because the cost of paying overtime rates was lower than having to pay the additional hiring costs that come with increasing headcount. An example of this is shown in months 11 and 12 of Figure 16. The results for every DC and operation can be found in Appendix B.

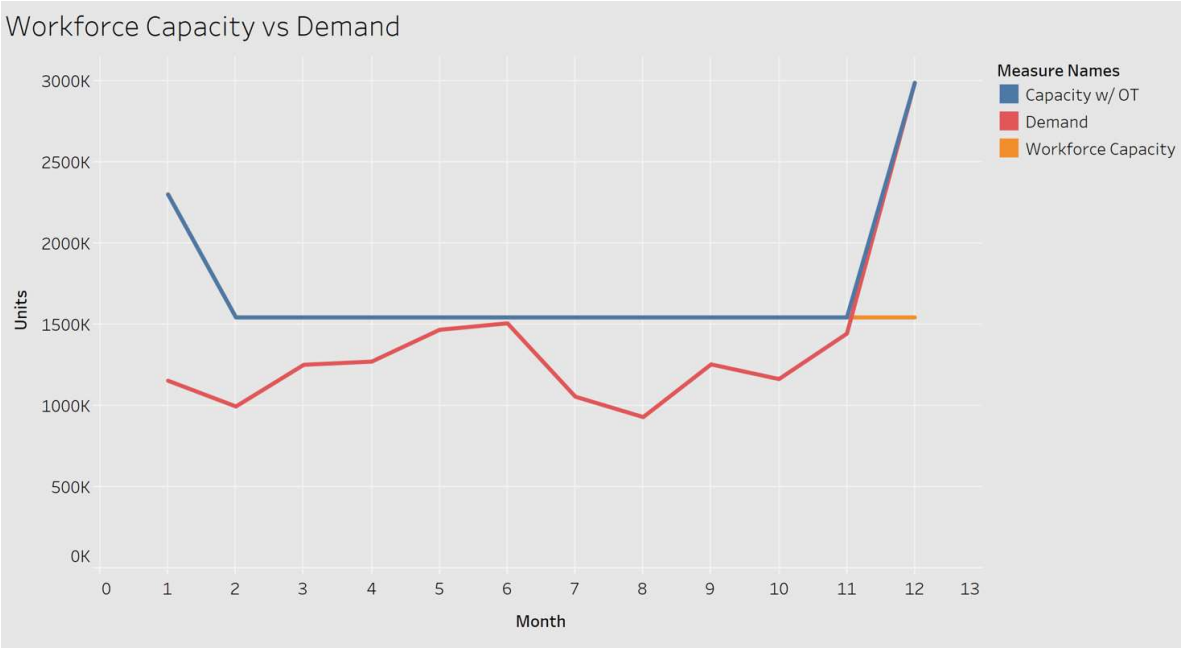



*Figure 16.* Workforce Capacity vs Demand for DC 2, Operation 1

Another output from the optimization model is the annual labor cost for each DC. The average cost per DC in 2019 was paired with 2020 demand to project expected cost if no changes were made, and this was compared to the cost per DC after implementation of our approach. These results can be found in Table 6. Overall, this optimization yields a total cost reduction of $125,359,858 (~4.8%) when compared to current-state operations

**Table 6**.
*Optimization Savings*

| Condition | Cost/DC | Cost/Unit | Total Cost (Adjusted for 2020 Demand) |
|---|---|---|---|
| Current State | $109,082,673.00 | $8.36 | $2,621,816,820.53 |
| Post-Optimization | $99,858,278.49 | $7.96 | $2,496,456,962.35 |

A sensitivity analysis was performed to show the effect of altered demand on unit cost. The optimization model was run on all distribution centers with 80% of forecasted demand and 120% of forecasted demand. The average unit price change can be found in Table 7. Unit price was very sensitive to demand and as demand increased, unit cost decreased. Even though ~40% more overtime was run in the 120% demand scenario, unit cost was still

reduced. This provides the recommendation that if the distribution center has the physical constraints for accommodation, higher volumes at each distribution center lead to lower costs.

**Table 7**.
*Unit Cost Sensitivity Analysis*

| Conditions | Total Cost | Demand (units) | Unit Cost ($/ |
|---|---|---|---|
| Baseline | $2,496,456,962.35 | 313,659,405 | $7.96 |
| 80% Demand | $2,148,393,857.24 | 250,927,524 | $8.56 |
| 120% Demand | $2,848,449,292.55 | 376,391,286 | $7.57 |

In order to reduce the effects of changes in demand on unit price, this optimization should be run monthly as the previous months of sales data is obtained.

## ***4.4 Environmental Impact***

The optimization of labor resources not only impacts cost, but also environmental footprint. The running of overtime shifts requires distribution centers to be using power for more operating hours and the expansion of sales requires additional stores. Based on Coppel's expansion plan and estimated demand, Coppel is expected to open ~100 stores during 2020. Applying our approach of estimating emissions per square meter discussed in Section 3.4, this growth will equate to an increase in ~21.81 tCO2e (3.18%) of Scope 2 emissions, shown in Table 8.

**Table 8.**
*Scope 2 emissions due to Store Expansion*

| Emission Rate per year (tCO2) | Expansion (Stores) | Yearly missions Impact (tCO2) |
|---|---|---|
| 0.22 | 100 | 21.81 |

| Store emissions 2019 | Store emissions 2020 | Delta |
|---|---|---|
| 686.15 | 707.96 | 3.18% |

On a distribution center level, the new optimized operational plan keeps facilities open for a combined 2,534 additional hours throughout the year due to overtime production. Applying the formula for estimating additional Scope 2 emissions impact from overtime hours discussed in Section 3.4, the Scope 2 emissions in DCs for 2020 will be 60.97 tCO2e, which is a 4.06% compared to 2019. However, when comparing the emission per unit sold, the optimization model has allowed a decrease of 1.24%

**Table 9.**
*Scope 2 emissions in Distribution Centers*

| | 2019 | 2020 | Delta |
|---|---|---|---|
| DC Emissions 2019 | 58.59 | 60.97 | 4.06% |
| Sales (Units) | 313,200,886 | 330,013,728 | 5.37% |
| Emission per unit | 1.87082E-07 | 1.84761E-07 | -1.24% |

# CHAPTER 5: CONCLUSION

In this capstone we have created an approach that determines current-state capacity, forecasts demand for the next year, calculates the optimal headcount per facility to minimize cost and meet demand, and evaluates the resulting environmental savings of implementation.

The inputs of our project were 3 years of monthly sales by store (January 2017 to December 2019), the average cycle time per process within Coppel facilities, and current headcount for each operation. The outputs of our project consist of an annual forecast (Appendix D) and an operational plan to optimize headcount (Appendix B). The application of these outputs allows the company to make decisions on a distribution center level that reduce labor costs while still maintaining the desired level of service to stores.

It is important to note that the recommendations provided are heavily dependent on the accuracy of current, available data. Even with taking steps to minimize the forecast error, the actual demand could still differ from the model due to unforeseen circumstances. Therefore, in order to ensure accuracy is as high as possible it is important to feed new data into the model as it becomes available. Our recommendations is to run the model on monthly basis. The operational decision for the number of workers to hire, fire and transfer per month and facility are all based on this forecast, so maintaining accuracy is key. Additionally, regular observations of distribution center processes are required to ensure the optimization model is representative of the conditions at the DCs.

Overall, this approach yielded a theoretical savings of MXN $126M (4.8%) over the next year for Coppel and decreased the per unit sold environmental footprint in 1.24% by providing a tool to build future forecasts, optimize headcount, and grants visibility of the

effects of supply chain decisions on CO2 emissions. With adequate data, this approach can be applied to any company in the retail space to help maintain a flexible and effective supply chain.

One of the limitations of our model was that the existing environmental footprint was aggregated over all of the company's facilities and our team had to estimate the impact of individual facilities. As a next research step, we suggest to include the environmental footprint as part of the optimization model and run a multi-objective optimization with weightage of cost and environmental footprint.

## REFERENCES


Alon, I., Qi, M., & Sadowski, R. J. (2001, May). Forecasting aggregate retail sales: a comparison of artificial neural networks and traditional methods. Journal of Retailing and Consumer Services, 8, 147-156.

Bouchery, Y., Corbett , C. J., Fransoo, J. C. & Tan, T. (Eds.). (2017). *Sustainable Supply Chains*. Switzerland. Springer.

Brownlee, J. (2019, August 21). *Machine Learning Mastery*. Retrieved from Machine Learning Mastery: https://machinelearningmastery.com/gentle-introduction-xgboost-applied-machine-learning/

Caplice, C. [SCx]. (2014, August 29). MITSCXX1T314-V009700_100 [Video file]. Recoverd from https://youtu.be/IYiXqrifdms

Caplice, C. [SCx]. (2014, September 8). MITSCXX1T314-V010500_100 [Video file]. Recoverd from https://youtu.be/ETL8XHYDHqk

Carbon Trust (2014) http://www.carbontrust.com

Chasan, E. (2019). Amazon's New Environmental Report Will Show How Bad Two-Day Shipping Is. Bloomberg, 1.

Clement, J. (2020, February 6). *Statista*. Retrieved from Statista: https://www.statista.com/statistics/272391/us-retail-e-commerce-sales-forecast/

Clutterbuck D (1982). After flexible hours, now it's flexiyears. *Int Mngt* **37**: 3, 31–36.

Corominas A and Pastor R (2000). Manpower planning and scheduling in services with seasonal demand. In: Dominguez JA (ed). *Proceedings of the First World Conference on Production and Operations Management (POM)*, Sevilla, CD-Rom.

Cox Jr T (1989). Towards the measurement of manufacturing flexibility. Prod Invent Mngt J 30(1): 68–72.

Curran P (1992) Annual hours brings productivity boost to Spicers. *Mngt Services* (July): **32**–33.

Fildes, R., Mab, S., & Kolassa, S. (2019, December 5). Retail forecasting: Research and practice. International Journal of Forecasting. doi:https://doi.org/10.1016/j.ijforecast.2019.06.004

Forbes. (2019, March 5). Asociación Mexicana de Venta Online. Retrieved from Forbes Mexico: https://www.forbes.com.mx/el-panorama-de-e-commerce-en-mexico-en-2019/ (In Spanish)

Gilliland, M. (2010). *The business forecasting deal: exposing myths, eliminating bad practices, providing practical solutions.* Hoboken, NJ: Wiley, 2010.

Golalizadeh, F., & Sharifi, M. (2016). Exploring the effect of customers' perceptions of electronic retailer ethics on revisit and purchase intention of retailer website. 2016 10th International Conference on E-Commerce in Developing Countries: With Focus on e-Tourism (ECDC), 1–6. https://doi.org/10.1109/ECDC.2016.7492975

Google Earth (2020a). [Google Earth: Coppel Tlapaque]. Retrieved April 28, 2020, from https://bit.ly/2yEDHLH

Google Earth (2020b). [Google Earth: Coppel CEDIS]. Retrieved April 28, 2020, from https://bit.ly/2SOSPNh

Gurobi. (2020, January 1). Gurobi. Retrieved from Gurobi: https://www.gurobi.com/company/about-gurobi/

Ho, S.-C., Kauffman, R. J., & Liang, T.-P. (2007). A growth theory perspective on B2C e-commerce growth in Europe: An exploratory study. Electronic Commerce Research and Applications, 6(3), 237–259. https://doi.org/10.1016/j.elerap.2006.06.003

Holt CC, Modigliani F, Muth JM and Simon HA (1960). *Planning, Production, Inventories and Work Force.* Prentice Hall: Englewood Cliffs, NJ.

Hung R (1999a). A multiple-shift workforce scheduling model under annualized hours. *Naval Res Logist* **46**: 726–736.

Hung R (1999b). Scheduling under annualized hours. *Int J Prod Res* **37**: 2419–2427.

IPCC (2007) Contribution of working groups I, II and III to the fourth assessment report of the intergovernmental panel on climate change. Intergovernmental Panel on Climate Change report.

IPCC (2013). IPCC fifth assessment report: summary for policymakers. Intergovernmental Panel on Climate Change report

Lynch P (1985). Annual hours: an idea whose time has come. *Personnel Mngt* (November) **17**: 46–50.

MacMeeking J (1995). Why Tesco's new composite distribution needed annual hours. *Int J Retail Distrib Mngt* **23**: 9, 36–38.

Makridakis, S., Hyndman, R. J., & Wheelwright, S. C. (n.d). *Forecasting: methods and applications.* Hoboken, NJ: J. Wiley, 1998.

Mazur L (1995). Coming: the annual workweeks. *Across the Board* **32**: 42–45.

Minx JC, Wiedmann T, Wood R, Peters GP, Lenzen M, Owen A, Scott K, Barrett J, Hubacek K, Baiocchi G, Paul A, Dawkins E, Briggs J, Guan D, Suh S, Ackerman F (2009). *Input–output analysis and carbon footprinting: an overview of applications*. Econ Syst Res **21**(3):187–216

Morde, V. (2019, April 7). *Towards Data Science*. Retrieved from Towards Data Science: https://towardsdatascience.com/https-medium-com-vishalmorde-xgboost-algorithm-long-she-may-rein-edd9f99be63d

Oke A (2000). Linking human resource flexibility with manufacturing flexibility: enablers of labour capacity flexibility in manufacturing plants. In: Dominguez JA (ed). Proceedings of the First World Conference on Production and Operations Management (POM), Sevilla, CD-Rom.

Pinedo, M. (2005). *Planning and Scheduling in Manufacturing and Services.* Springer.

Ross, S. (2019, January 9). Investopedia. Retrieved from Investopedia: https://www.investopedia.com/ask/answers/071615/what-profit-margin-usual-company-retail-sector.asp

Seabold, Skipper, and Perktold (2010). Statsmodels: Econometric and statistical modeling with python. *Proceedings of the 9th Python in Science Conference.*

Sheffi, Y., & Blanco, E. (2018). Balancing Green: When to Embrace Sustainability in a Business (And When Not To). Cambridge: The MIT Press.

U.S. Department of Commerce. (2019). Quarterly Retail E-commerce Sales. Washington, D.C.: U.S. Department of Commerce: https://www.census.gov/retail/mrts/www/data/pdf/ec_current.pdf

United Nations. (2019, October 19). Sustainable Development Goals. Retrieved from United Nations: un.org/sustainabledevelopment/sustainable-development-goals/

WEF (2014) Global risks 2014, 9th edn. World Economic Forum

What is Python? Executive Summary. (2020) Retrieved from https://www.python.org/doc/essays/blurb/

WRI and WBCSD (2004). The greenhouse protocol: a corporate accounting and reporting standard, revised edition. World Resources Institute and World Business Council for Sustainable Development

WRI and WBCSD (2011) Corporate value chain (Scope 3) accounting and reporting standard. World Resources Institute and World Business Council for Sustainable Development

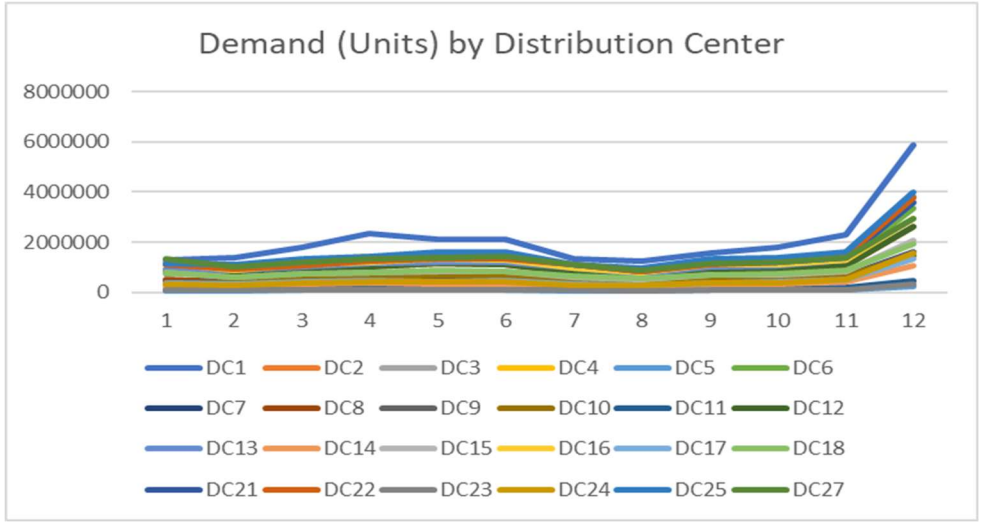
Young, J. (2019, November 13). *Digital Commerce 360*. Retrieved from Digital Commerce 360: https://www.digitalcommerce360.com/article/global-ecommerce-sales/

Zotteri, G., Kalchschmidt, M., & Caniato, F. (2005, January 8). The impact of aggregation level on forecasting performance. International Journal of Production Economics, 93-94, 479-491

# APPENDICES

## *A. Forecast Results*

| Month | DC1 | DC2 | DC3 | DC4 | DC5 | DC6 | DC7 | DC8 | DC9 | DC10 | DC11 | DC12 | DC13 | DC14 | DC15 | DC16 | DC17 | DC18 | DC21 | DC22 | DC23 | DC24 | DC25 | DC27 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 1299695 | 845312 | 846555 | 914984 | 53492 | 729193 | 443574 | 532443 | 387342 | 324376 | 94747 | 845312 | 947453 | 251978 | 771496 | 1171732 | 382024 | 781244 | 1093409 | 1105824 | 84883 | 316643 | 1175943 | 1322268 |
| 2 | 1394581 | 752664 | 911839 | 973680 | 61646 | 872454 | 446951 | 439346 | 441299 | 377030 | 104410 | 752664 | 861201 | 248768 | 608904 | 888051 | 346983 | 600754 | 949849 | 909666 | 88605 | 299358 | 1109961 | 997140 |
| 3 | 1814344 | 939219 | 1212441 | 1288338 | 88981 | 1132982 | 535200 | 539770 | 528062 | 519498 | 143879 | 939219 | 1009977 | 314872 | 744444 | 1101295 | 416094 | 736184 | 1154210 | 1071620 | 104060 | 367228 | 1339540 | 1206506 |
| 4 | 2324775 | 1023969 | 1376860 | 1442486 | 108122 | 1262843 | 574623 | 561298 | 600768 | 580191 | 185520 | 1023969 | 1135264 | 376358 | 807301 | 1183196 | 447669 | 806018 | 1352231 | 1247473 | 113285 | 427738 | 1446834 | 1325916 |
| 5 | 2130170 | 1139798 | 1465306 | 1482026 | 108685 | 1340888 | 615079 | 610677 | 632187 | 676578 | 170247 | 1139798 | 1182215 | 344048 | 844778 | 1321784 | 441235 | 862100 | 1379361 | 1324330 | 113917 | 410086 | 1615674 | 1389731 |
| 6 | 2095187 | 1121740 | 1480433 | 1538550 | 108730 | 1338020 | 604516 | 591819 | 620298 | 642512 | 165212 | 1121740 | 1216927 | 338306 | 821247 | 1287322 | 479387 | 835674 | 1373705 | 1331287 | 109060 | 425692 | 1625249 | 1437491 |
| 7 | 1348528 | 782275 | 1012700 | 1021274 | 63765 | 893242 | 412415 | 412247 | 545749 | 401334 | 97748 | 782275 | 971820 | 241109 | 614544 | 952267 | 316718 | 668787 | 1100827 | 1099253 | 89143 | 267382 | 1113989 | 1091228 |
| 8 | 1254069 | 658629 | 821490 | 906348 | 61884 | 755235 | 405244 | 376809 | 486658 | 354715 | 94878 | 658629 | 735595 | 213451 | 517105 | 779534 | 289128 | 550434 | 845005 | 855563 | 68768 | 265817 | 971800 | 871866 |
| 9 | 1575754 | 917616 | 1158302 | 1282077 | 83841 | 1080407 | 516070 | 525414 | 513839 | 495709 | 124755 | 917616 | 1021809 | 265694 | 694050 | 1091142 | 392825 | 710598 | 1136512 | 1103321 | 94988 | 364790 | 1354720 | 1173642 |
| 10 | 1777765 | 928026 | 1217088 | 1314730 | 89457 | 1154397 | 538754 | 536302 | 575913 | 530717 | 143089 | 928026 | 1073302 | 307747 | 715733 | 1091521 | 402409 | 748478 | 1186609 | 1222675 | 99167 | 367208 | 1393642 | 1185251 |
| 11 | 2306754 | 1060085 | 1415100 | 1575678 | 110499 | 1291192 | 645049 | 649754 | 657952 | 662827 | 186315 | 1060085 | 1307144 | 404058 | 838649 | 1277847 | 497510 | 878052 | 1386215 | 1468422 | 121140 | 496137 | 1612885 | 1391815 |
| 12 | 5868181 | 2605934 | 3575509 | 3813704 | 257453 | 3378026 | 1629774 | 1489065 | 1583525 | 1496118 | 484775 | 2605934 | 3800071 | 1070205 | 2049399 | 2948553 | 1360243 | 1951303 | 3578000 | 3782377 | 324286 | 1586154 | 4014210 | 2961466 |



Distribution Center Monthly Forecast – Clothing (Units)

| Month | DC1 | DC2 | DC3 | DC4 | DC5 | DC6 | DC7 | DC8 |
|---|---|---|---|---|---|---|---|---|
| 1 | 1,299,695 | 845,312 | 846,555 | 914,984 | 53,492 | 729,193 | 443,574 | 532,444 |
| 2 | 1,394,581 | 752,664 | 911,840 | 973,680 | 61,646 | 872,454 | 446,951 | 439,347 |
| 3 | 1,814,344 | 939,219 | 1,212,441 | 1,288,338 | 88,981 | 1,132,982 | 535,201 | 539,770 |
| 4 | 2,324,775 | 1,023,969 | 1,376,860 | 1,442,486 | 108,122 | 1,262,843 | 574,623 | 561,298 |
| 5 | 2,130,170 | 1,139,798 | 1,465,306 | 1,482,026 | 108,685 | 1,340,888 | 615,079 | 610,677 |
| 6 | 2,095,187 | 1,121,740 | 1,480,433 | 1,538,550 | 108,730 | 1,338,020 | 604,516 | 591,819 |
| 7 | 1,348,528 | 782,275 | 1,012,700 | 1,021,274 | 63,765 | 893,242 | 412,415 | 412,247 |
| 8 | 1,254,069 | 658,629 | 821,490 | 906,348 | 61,884 | 755,235 | 405,244 | 376,809 |
| 9 | 1,575,754 | 917,616 | 1,158,302 | 1,282,077 | 83,841 | 1,080,407 | 516,070 | 525,414 |
| 10 | 1,777,765 | 928,026 | 1,217,088 | 1,314,730 | 89,457 | 1,154,397 | 538,754 | 536,302 |
| 11 | 2,306,754 | 1,060,085 | 1,415,100 | 1,575,678 | 110,499 | 1,291,192 | 645,049 | 649,754 |
| 12 | 5,868,181 | 2,605,934 | 3,575,509 | 3,813,704 | 257,453 | 3,378,026 | 1,629,774 | 1,489,065 |

Distribution Center Monthly Forecast – Furniture (Units)

| Month | DC1 | DC2 | DC3 | DC4 | DC6 | DC7 | DC8 | DC9 |
|---|---|---|---|---|---|---|---|---|
| 1 | 244,355 | 309,918 | 78,159 | 257,645 | 338,023 | 112,481 | 193,568 | 491,187 |
| 2 | 238,388 | 243,929 | 77,965 | 240,208 | 350,055 | 101,887 | 135,940 | 361,639 |
| 3 | 332,786 | 314,422 | 103,692 | 313,656 | 464,021 | 121,091 | 168,309 | 445,611 |
| 4 | 227,663 | 249,101 | 83,747 | 255,491 | 365,038 | 100,408 | 127,003 | 344,837 |
| 5 | 309,909 | 329,514 | 115,775 | 346,683 | 498,609 | 112,481 | 147,496 | 394,422 |
| 6 | 429,196 | 387,618 | 127,466 | 440,165 | 613,183 | 134,560 | 181,473 | 467,379 |
| 7 | 238,588 | 274,591 | 82,882 | 287,292 | 380,595 | 108,754 | 138,818 | 386,598 |
| 8 | 243,044 | 272,567 | 83,033 | 288,607 | 371,794 | 109,542 | 133,827 | 362,264 |
| 9 | 305,244 | 338,354 | 98,655 | 344,583 | 460,647 | 123,511 | 162,759 | 422,451 |
| 10 | 223,635 | 237,470 | 71,179 | 236,751 | 313,668 | 98,467 | 126,049 | 323,033 |
| 11 | 487,134 | 385,528 | 105,974 | 391,861 | 473,435 | 133,703 | 182,221 | 467,304 |
| 12 | 487,134 | 385,528 | 105,974 | 391,861 | 473,435 | 250,630 | 323,963 | 851,195 |

## ***B. Optimization Results***

| t | DC | Operation | Variable | Value_Clothing | Value_Furniture | Value_Total |
|---|---|---|---|---|---|---|
| 1 | 1 | 1 | fire | 0 | 243 | 243 |
| 1 | 1 | 1 | hire | 39 | 0 | 39 |
| 1 | 1 | 1 | workers | 146 | 338 | 484 |
| 1 | 1 | 2 | fire | 0 | 236 | 236 |
| 1 | 1 | 2 | hire | 316 | 0 | 316 |
| 1 | 1 | 2 | workers | 435 | 380 | 815 |
| 1 | 1 | 3 | fire | 98 | 167 | 265 |
| 1 | 1 | 3 | workers | 182 | 182 | 364 |
| 1 | 1 | 4 | fire | 94 | 163 | 257 |
| 1 | 1 | 4 | workers | 178 | 178 | 356 |
| 1 | 2 | 1 | fire | 45 | 34 | 79 |
| 1 | 2 | 1 | workers | 127 | 113 | 240 |
| 1 | 2 | 2 | hire | 120 | 0 | 120 |
| 1 | 2 | 2 | ot_1 | 5 | 0 | 5 |
| 1 | 2 | 2 | workers | 213 | 119 | 332 |
| 1 | 2 | 3 | fire | 197 | 222 | 419 |
| 1 | 2 | 3 | workers | 234 | 234 | 468 |
| 1 | 2 | 4 | fire | 113 | 138 | 251 |
| 1 | 2 | 4 | workers | 150 | 150 | 300 |
| 1 | 3 | 1 | hire | 56 | 10 | 66 |
| 1 | 3 | 1 | ot_1 | 0 | 1 | 1 |
| 1 | 3 | 1 | ot_2 | 0 | 6 | 6 |
| 1 | 3 | 1 | workers | 56 | 10 | 66 |
| 1 | 3 | 2 | hire | 229 | 14 | 243 |
| 1 | 3 | 2 | ot_1 | 0 | 2 | 2 |
| 1 | 3 | 2 | ot_2 | 0 | 10 | 10 |
| 1 | 3 | 2 | workers | 229 | 14 | 243 |
| 1 | 3 | 3 | fire | 0 | 1 | 1 |
| 1 | 3 | 3 | hire | 25 | 2 | 27 |
| 1 | 3 | 3 | transfer | 1 | 0 | 1 |
| 1 | 3 | 3 | workers | 25 | 2 | 27 |
| 1 | 3 | 4 | fire | 0 | 1 | 1 |
| 1 | 3 | 4 | hire | 26 | 2 | 28 |
| 1 | 3 | 4 | workers | 26 | 2 | 28 |
| 1 | 4 | 1 | hire | 18 | 0 | 18 |
| 1 | 4 | 1 | workers | 102 | 62 | 164 |
| 1 | 4 | 2 | hire | 215 | 45 | 260 |
| 1 | 4 | 2 | ot_1 | 0 | 8 | 8 |
| 1 | 4 | 2 | ot_2 | 0 | 37 | 37 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 4 | 2 | workers | 273 | 45 | 318 |
| 1 | 4 | 3 | fire | 54 | 96 | 150 |
| 1 | 4 | 3 | workers | 109 | 109 | 218 |
| 1 | 4 | 4 | fire | 15 | 57 | 72 |
| 1 | 4 | 4 | workers | 70 | 70 | 140 |
| 1 | 5 | 1 | hire | 4 | 0 | 4 |
| 1 | 5 | 1 | workers | 4 | 0 | 4 |
| 1 | 5 | 2 | hire | 16 | 0 | 16 |
| 1 | 5 | 2 | workers | 16 | 0 | 16 |
| 1 | 5 | 3 | hire | 2 | 0 | 2 |
| 1 | 5 | 3 | workers | 2 | 0 | 2 |
| 1 | 5 | 4 | hire | 2 | 0 | 2 |
| 1 | 5 | 4 | workers | 2 | 0 | 2 |
| 1 | 6 | 1 | fire | 61 | 0 | 61 |
| 1 | 6 | 1 | hire | 0 | 1 | 1 |
| 1 | 6 | 1 | ot_1 | 0 | 8 | 8 |
| 1 | 6 | 1 | workers | 168 | 70 | 238 |
| 1 | 6 | 2 | fire | 0 | 127 | 127 |
| 1 | 6 | 2 | hire | 152 | 0 | 152 |
| 1 | 6 | 2 | workers | 281 | 302 | 583 |
| 1 | 6 | 3 | fire | 90 | 120 | 210 |
| 1 | 6 | 3 | workers | 138 | 138 | 276 |
| 1 | 6 | 4 | fire | 295 | 325 | 620 |
| 1 | 6 | 4 | workers | 343 | 343 | 686 |
| 1 | 7 | 1 | fire | 20 | 71 | 91 |
| 1 | 7 | 1 | workers | 72 | 100 | 172 |
| 1 | 7 | 2 | fire | 0 | 61 | 61 |
| 1 | 7 | 2 | hire | 77 | 0 | 77 |
| 1 | 7 | 2 | workers | 132 | 105 | 237 |
| 1 | 7 | 3 | fire | 45 | 63 | 108 |
| 1 | 7 | 3 | workers | 68 | 68 | 136 |
| 1 | 7 | 4 | fire | 122 | 140 | 262 |
| 1 | 7 | 4 | workers | 145 | 145 | 290 |
| 1 | 8 | 1 | fire | 46 | 21 | 67 |
| 1 | 8 | 1 | workers | 93 | 61 | 154 |
| 1 | 8 | 2 | fire | 0 | 32 | 32 |
| 1 | 8 | 2 | hire | 56 | 0 | 56 |
| 1 | 8 | 2 | ot_1 | 3 | 0 | 3 |
| 1 | 8 | 2 | workers | 134 | 92 | 226 |
| 1 | 8 | 3 | fire | 0 | 8 | 8 |
| 1 | 8 | 3 | ot_1 | 1 | 0 | 1 |
| 1 | 8 | 3 | transfer | 8 | 0 | 8 |
| 1 | 8 | 3 | workers | 14 | 14 | 28 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 8 | 4 | fire | 278 | 302 | 580 |
| 1 | 8 | 4 | workers | 308 | 308 | 616 |
| 1 | 9 | 1 | fire | 0 | 10 | 10 |
| 1 | 9 | 1 | hire | 25 | 57 | 82 |
| 1 | 9 | 1 | ot_1 | 0 | 10 | 10 |
| 1 | 9 | 1 | ot_2 | 0 | 46 | 46 |
| 1 | 9 | 1 | workers | 25 | 57 | 82 |
| 1 | 9 | 2 | fire | 0 | 15 | 15 |
| 1 | 9 | 2 | hire | 101 | 86 | 187 |
| 1 | 9 | 2 | ot_1 | 0 | 16 | 16 |
| 1 | 9 | 2 | ot_2 | 0 | 70 | 70 |
| 1 | 9 | 2 | workers | 101 | 86 | 187 |
| 1 | 9 | 3 | fire | 0 | 2 | 2 |
| 1 | 9 | 3 | hire | 11 | 9 | 20 |
| 1 | 9 | 3 | ot_1 | 0 | 1 | 1 |
| 1 | 9 | 3 | ot_2 | 0 | 7 | 7 |
| 1 | 9 | 3 | transfer | 1 | 0 | 1 |
| 1 | 9 | 3 | workers | 11 | 9 | 20 |
| 1 | 9 | 4 | fire | 0 | 2 | 2 |
| 1 | 9 | 4 | hire | 12 | 9 | 21 |
| 1 | 9 | 4 | ot_1 | 0 | 1 | 1 |
| 1 | 9 | 4 | ot_2 | 0 | 7 | 7 |
| 1 | 9 | 4 | workers | 12 | 9 | 21 |
| 1 | 10 | 1 | hire | 24 | 38 | 62 |
| 1 | 10 | 1 | ot_1 | 0 | 7 | 7 |
| 1 | 10 | 1 | ot_2 | 0 | 29 | 29 |
| 1 | 10 | 1 | workers | 24 | 38 | 62 |
| 1 | 10 | 2 | hire | 96 | 57 | 153 |
| 1 | 10 | 2 | ot_1 | 0 | 10 | 10 |
| 1 | 10 | 2 | ot_2 | 0 | 45 | 45 |
| 1 | 10 | 2 | workers | 96 | 57 | 153 |
| 1 | 10 | 3 | hire | 10 | 6 | 16 |
| 1 | 10 | 3 | ot_2 | 0 | 4 | 4 |
| 1 | 10 | 3 | transfer | 1 | 0 | 1 |
| 1 | 10 | 3 | workers | 10 | 6 | 16 |
| 1 | 10 | 4 | hire | 11 | 6 | 17 |
| 1 | 10 | 4 | ot_2 | 0 | 4 | 4 |
| 1 | 10 | 4 | workers | 11 | 6 | 17 |
| 1 | 11 | 1 | fire | 45 | 133 | 178 |
| 1 | 11 | 1 | workers | 61 | 176 | 237 |
| 1 | 11 | 2 | fire | 0 | 128 | 128 |
| 1 | 11 | 2 | workers | 63 | 193 | 256 |
| 1 | 11 | 3 | fire | 100 | 100 | 200 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 11 | 3 | workers | 107 | 107 | 214 |
| 1 | 11 | 4 | fire | 69 | 69 | 138 |
| 1 | 11 | 4 | workers | 76 | 76 | 152 |
| 1 | 12 | 1 | hire | 41 | 0 | 41 |
| 1 | 12 | 1 | ot_1 | 7 | 0 | 7 |
| 1 | 12 | 1 | ot_2 | 5 | 0 | 5 |
| 1 | 12 | 1 | workers | 41 | 0 | 41 |
| 1 | 12 | 2 | hire | 167 | 0 | 167 |
| 1 | 12 | 2 | ot_1 | 31 | 0 | 31 |
| 1 | 12 | 2 | ot_2 | 20 | 0 | 20 |
| 1 | 12 | 2 | workers | 167 | 0 | 167 |
| 1 | 12 | 3 | fire | 1 | 0 | 1 |
| 1 | 12 | 3 | hire | 19 | 0 | 19 |
| 1 | 12 | 3 | ot_1 | 3 | 0 | 3 |
| 1 | 12 | 3 | ot_2 | 1 | 0 | 1 |
| 1 | 12 | 3 | workers | 19 | 0 | 19 |
| 1 | 12 | 4 | fire | 1 | 0 | 1 |
| 1 | 12 | 4 | hire | 19 | 0 | 19 |
| 1 | 12 | 4 | ot_1 | 3 | 0 | 3 |
| 1 | 12 | 4 | ot_2 | 1 | 0 | 1 |
| 1 | 12 | 4 | workers | 19 | 0 | 19 |
| 1 | 13 | 1 | fire | 23 | 165 | 188 |
| 1 | 13 | 1 | workers | 143 | 218 | 361 |
| 1 | 13 | 2 | fire | 0 | 158 | 158 |
| 1 | 13 | 2 | hire | 182 | 0 | 182 |
| 1 | 13 | 2 | workers | 304 | 238 | 542 |
| 1 | 13 | 3 | fire | 188 | 234 | 422 |
| 1 | 13 | 3 | workers | 242 | 242 | 484 |
| 1 | 13 | 4 | fire | 37 | 83 | 120 |
| 1 | 13 | 4 | workers | 91 | 91 | 182 |
| 1 | 14 | 1 | fire | 0 | 14 | 14 |
| 1 | 14 | 1 | hire | 17 | 43 | 60 |
| 1 | 14 | 1 | ot_1 | 0 | 8 | 8 |
| 1 | 14 | 1 | ot_2 | 0 | 34 | 34 |
| 1 | 14 | 1 | workers | 17 | 43 | 60 |
| 1 | 14 | 2 | fire | 0 | 22 | 22 |
| 1 | 14 | 2 | hire | 69 | 65 | 134 |
| 1 | 14 | 2 | ot_1 | 0 | 12 | 12 |
| 1 | 14 | 2 | ot_2 | 0 | 52 | 52 |
| 1 | 14 | 2 | workers | 69 | 65 | 134 |
| 1 | 14 | 3 | fire | 0 | 3 | 3 |
| 1 | 14 | 3 | hire | 7 | 7 | 14 |
| 1 | 14 | 3 | ot_1 | 0 | 1 | 1 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 14 | 3 | ot_2 | 0 | 5 | 5 |
| 1 | 14 | 3 | transfer | 1 | 0 | 1 |
| 1 | 14 | 3 | workers | 7 | 7 | 14 |
| 1 | 14 | 4 | fire | 0 | 3 | 3 |
| 1 | 14 | 4 | hire | 8 | 7 | 15 |
| 1 | 14 | 4 | ot_1 | 0 | 1 | 1 |
| 1 | 14 | 4 | ot_2 | 0 | 5 | 5 |
| 1 | 14 | 4 | workers | 8 | 7 | 15 |
| 1 | 15 | 1 | fire | 27 | 76 | 103 |
| 1 | 15 | 1 | workers | 92 | 131 | 223 |
| 1 | 15 | 2 | fire | 0 | 63 | 63 |
| 1 | 15 | 2 | hire | 91 | 0 | 91 |
| 1 | 15 | 2 | ot_1 | 28 | 0 | 28 |
| 1 | 15 | 2 | workers | 171 | 146 | 317 |
| 1 | 15 | 3 | fire | 37 | 58 | 95 |
| 1 | 15 | 3 | workers | 66 | 66 | 132 |
| 1 | 15 | 4 | fire | 281 | 302 | 583 |
| 1 | 15 | 4 | workers | 310 | 310 | 620 |
| 1 | 16 | 1 | fire | 57 | 72 | 129 |
| 1 | 16 | 1 | workers | 150 | 151 | 301 |
| 1 | 16 | 2 | fire | 0 | 74 | 74 |
| 1 | 16 | 2 | hire | 130 | 0 | 130 |
| 1 | 16 | 2 | ot_1 | 46 | 0 | 46 |
| 1 | 16 | 2 | ot_2 | 9 | 0 | 9 |
| 1 | 16 | 2 | workers | 247 | 193 | 440 |
| 1 | 16 | 3 | fire | 66 | 96 | 162 |
| 1 | 16 | 3 | workers | 108 | 108 | 216 |
| 1 | 16 | 4 | fire | 200 | 230 | 430 |
| 1 | 16 | 4 | workers | 242 | 242 | 484 |
| 1 | 17 | 1 | hire | 21 | 23 | 44 |
| 1 | 17 | 1 | ot_1 | 3 | 4 | 7 |
| 1 | 17 | 1 | ot_2 | 0 | 17 | 17 |
| 1 | 17 | 1 | workers | 21 | 23 | 44 |
| 1 | 17 | 2 | hire | 87 | 34 | 121 |
| 1 | 17 | 2 | ot_1 | 11 | 6 | 17 |
| 1 | 17 | 2 | ot_2 | 0 | 27 | 27 |
| 1 | 17 | 2 | workers | 87 | 34 | 121 |
| 1 | 17 | 3 | fire | 0 | 1 | 1 |
| 1 | 17 | 3 | hire | 9 | 4 | 13 |
| 1 | 17 | 3 | ot_1 | 1 | 0 | 1 |
| 1 | 17 | 3 | ot_2 | 1 | 2 | 3 |
| 1 | 17 | 3 | transfer | 1 | 0 | 1 |
| 1 | 17 | 3 | workers | 9 | 4 | 13 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 17 | 4 | fire | 0 | 1 | 1 |
| 1 | 17 | 4 | hire | 10 | 4 | 14 |
| 1 | 17 | 4 | ot_1 | 1 | 0 | 1 |
| 1 | 17 | 4 | ot_2 | 0 | 2 | 2 |
| 1 | 17 | 4 | workers | 10 | 4 | 14 |
| 1 | 18 | 1 | fire | 51 | 81 | 132 |
| 1 | 18 | 1 | workers | 113 | 114 | 227 |
| 1 | 18 | 2 | fire | 0 | 82 | 82 |
| 1 | 18 | 2 | hire | 78 | 0 | 78 |
| 1 | 18 | 2 | ot_1 | 30 | 0 | 30 |
| 1 | 18 | 2 | workers | 172 | 132 | 304 |
| 1 | 18 | 3 | fire | 152 | 175 | 327 |
| 1 | 18 | 3 | workers | 180 | 180 | 360 |
| 1 | 18 | 4 | fire | 336 | 359 | 695 |
| 1 | 18 | 4 | workers | 364 | 364 | 728 |
| 1 | 19 | 1 | fire | 0 | 2 | 2 |
| 1 | 19 | 1 | hire | 0 | 38 | 38 |
| 1 | 19 | 1 | ot_1 | 0 | 7 | 7 |
| 1 | 19 | 1 | ot_2 | 0 | 30 | 30 |
| 1 | 19 | 1 | workers | 0 | 38 | 38 |
| 1 | 19 | 2 | fire | 0 | 3 | 3 |
| 1 | 19 | 2 | hire | 0 | 57 | 57 |
| 1 | 19 | 2 | ot_1 | 0 | 10 | 10 |
| 1 | 19 | 2 | ot_2 | 0 | 46 | 46 |
| 1 | 19 | 2 | workers | 0 | 57 | 57 |
| 1 | 19 | 3 | fire | 0 | 1 | 1 |
| 1 | 19 | 3 | hire | 0 | 6 | 6 |
| 1 | 19 | 3 | ot_1 | 0 | 1 | 1 |
| 1 | 19 | 3 | ot_2 | 0 | 4 | 4 |
| 1 | 19 | 3 | workers | 0 | 6 | 6 |
| 1 | 19 | 4 | fire | 0 | 1 | 1 |
| 1 | 19 | 4 | hire | 0 | 6 | 6 |
| 1 | 19 | 4 | ot_1 | 0 | 1 | 1 |
| 1 | 19 | 4 | ot_2 | 0 | 4 | 4 |
| 1 | 19 | 4 | workers | 0 | 6 | 6 |
| 1 | 20 | 1 | hire | 0 | 37 | 37 |
| 1 | 20 | 1 | ot_1 | 0 | 6 | 6 |
| 1 | 20 | 1 | ot_2 | 0 | 30 | 30 |
| 1 | 20 | 1 | workers | 0 | 37 | 37 |
| 1 | 20 | 2 | hire | 0 | 56 | 56 |
| 1 | 20 | 2 | ot_1 | 0 | 10 | 10 |
| 1 | 20 | 2 | ot_2 | 0 | 45 | 45 |
| 1 | 20 | 2 | workers | 0 | 56 | 56 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 20 | 3 | fire | 0 | 1 | 1 |
| 1 | 20 | 3 | hire | 0 | 6 | 6 |
| 1 | 20 | 3 | ot_1 | 0 | 1 | 1 |
| 1 | 20 | 3 | ot_2 | 0 | 4 | 4 |
| 1 | 20 | 3 | workers | 0 | 6 | 6 |
| 1 | 20 | 4 | fire | 0 | 1 | 1 |
| 1 | 20 | 4 | hire | 0 | 6 | 6 |
| 1 | 20 | 4 | ot_1 | 0 | 1 | 1 |
| 1 | 20 | 4 | ot_2 | 0 | 4 | 4 |
| 1 | 20 | 4 | workers | 0 | 6 | 6 |
| 1 | 21 | 1 | fire | 13 | 43 | 56 |
| 1 | 21 | 1 | workers | 126 | 43 | 169 |
| 1 | 21 | 2 | fire | 0 | 173 | 173 |
| 1 | 21 | 2 | hire | 177 | 0 | 177 |
| 1 | 21 | 2 | ot_1 | 1 | 0 | 1 |
| 1 | 21 | 2 | workers | 282 | 173 | 455 |
| 1 | 21 | 3 | fire | 54 | 105 | 159 |
| 1 | 21 | 3 | workers | 105 | 105 | 210 |
| 1 | 21 | 4 | fire | 130 | 181 | 311 |
| 1 | 21 | 4 | workers | 181 | 181 | 362 |
| 1 | 22 | 1 | fire | 2 | 65 | 67 |
| 1 | 22 | 1 | workers | 121 | 155 | 276 |
| 1 | 22 | 2 | fire | 0 | 26 | 26 |
| 1 | 22 | 2 | hire | 193 | 0 | 193 |
| 1 | 22 | 2 | workers | 291 | 161 | 452 |
| 1 | 22 | 3 | fire | 60 | 100 | 160 |
| 1 | 22 | 3 | workers | 114 | 114 | 228 |
| 1 | 22 | 4 | fire | 78 | 118 | 196 |
| 1 | 22 | 4 | workers | 132 | 132 | 264 |
| 1 | 23 | 1 | hire | 5 | 0 | 5 |
| 1 | 23 | 1 | workers | 5 | 0 | 5 |
| 1 | 23 | 2 | hire | 21 | 0 | 21 |
| 1 | 23 | 2 | workers | 21 | 0 | 21 |
| 1 | 23 | 3 | hire | 2 | 0 | 2 |
| 1 | 23 | 3 | transfer | 1 | 0 | 1 |
| 1 | 23 | 3 | workers | 2 | 0 | 2 |
| 1 | 23 | 4 | hire | 3 | 0 | 3 |
| 1 | 23 | 4 | workers | 3 | 0 | 3 |
| 1 | 24 | 1 | fire | 45 | 118 | 163 |
| 1 | 24 | 1 | workers | 95 | 118 | 213 |
| 1 | 24 | 2 | fire | 0 | 132 | 132 |
| 1 | 24 | 2 | hire | 69 | 0 | 69 |
| 1 | 24 | 2 | workers | 134 | 132 | 266 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 1 | 24 | 3 | fire | 72 | 95 | 167 |
| 1 | 24 | 3 | workers | 95 | 95 | 190 |
| 1 | 24 | 4 | fire | 72 | 95 | 167 |
| 1 | 24 | 4 | workers | 95 | 95 | 190 |
| 1 | 25 | 1 | fire | 15 | 223 | 238 |
| 1 | 25 | 1 | workers | 142 | 223 | 365 |
| 1 | 25 | 2 | fire | 0 | 234 | 234 |
| 1 | 25 | 2 | hire | 192 | 0 | 192 |
| 1 | 25 | 2 | workers | 322 | 234 | 556 |
| 1 | 25 | 3 | fire | 133 | 190 | 323 |
| 1 | 25 | 3 | workers | 190 | 190 | 380 |
| 1 | 25 | 4 | fire | 78 | 135 | 213 |
| 1 | 25 | 4 | workers | 135 | 135 | 270 |
| 2 | 1 | 1 | workers | 185 | 95 | 280 |
| 2 | 1 | 2 | workers | 751 | 144 | 895 |
| 2 | 1 | 3 | workers | 84 | 15 | 99 |
| 2 | 1 | 4 | workers | 84 | 15 | 99 |
| 2 | 2 | 1 | workers | 82 | 79 | 161 |
| 2 | 2 | 2 | workers | 333 | 119 | 452 |
| 2 | 2 | 3 | workers | 37 | 12 | 49 |
| 2 | 2 | 4 | workers | 37 | 12 | 49 |
| 2 | 3 | 1 | workers | 112 | 20 | 132 |
| 2 | 3 | 2 | workers | 458 | 28 | 486 |
| 2 | 3 | 3 | workers | 51 | 3 | 54 |
| 2 | 3 | 4 | workers | 51 | 3 | 54 |
| 2 | 4 | 1 | workers | 120 | 62 | 182 |
| 2 | 4 | 2 | workers | 488 | 90 | 578 |
| 2 | 4 | 3 | workers | 55 | 13 | 68 |
| 2 | 4 | 4 | workers | 55 | 13 | 68 |
| 2 | 5 | 1 | workers | 8 | 0 | 8 |
| 2 | 5 | 2 | workers | 32 | 0 | 32 |
| 2 | 5 | 3 | workers | 4 | 0 | 4 |
| 2 | 5 | 4 | workers | 4 | 0 | 4 |
| 2 | 6 | 1 | ot_1 | 0 | 10 | 10 |
| 2 | 6 | 1 | workers | 107 | 71 | 178 |
| 2 | 6 | 2 | workers | 433 | 175 | 608 |
| 2 | 6 | 3 | workers | 48 | 18 | 66 |
| 2 | 6 | 4 | workers | 48 | 18 | 66 |
| 2 | 7 | 1 | workers | 52 | 29 | 81 |
| 2 | 7 | 2 | workers | 209 | 44 | 253 |
| 2 | 7 | 3 | workers | 23 | 5 | 28 |
| 2 | 7 | 4 | workers | 23 | 5 | 28 |
| 2 | 8 | 1 | workers | 47 | 40 | 87 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 2 | 8 | 2 | workers | 190 | 60 | 250 |
| 2 | 8 | 3 | workers | 22 | 6 | 28 |
| 2 | 8 | 4 | workers | 22 | 6 | 28 |
| 2 | 9 | 1 | workers | 50 | 104 | 154 |
| 2 | 9 | 2 | workers | 202 | 157 | 359 |
| 2 | 9 | 3 | workers | 23 | 16 | 39 |
| 2 | 9 | 4 | workers | 23 | 16 | 39 |
| 2 | 10 | 1 | workers | 48 | 76 | 124 |
| 2 | 10 | 2 | workers | 192 | 114 | 306 |
| 2 | 10 | 3 | workers | 21 | 12 | 33 |
| 2 | 10 | 4 | workers | 21 | 12 | 33 |
| 2 | 11 | 1 | workers | 16 | 43 | 59 |
| 2 | 11 | 2 | workers | 63 | 65 | 128 |
| 2 | 11 | 3 | workers | 7 | 7 | 14 |
| 2 | 11 | 4 | workers | 7 | 7 | 14 |
| 2 | 12 | 1 | workers | 82 | 0 | 82 |
| 2 | 12 | 2 | workers | 334 | 0 | 334 |
| 2 | 12 | 3 | workers | 37 | 0 | 37 |
| 2 | 12 | 4 | workers | 37 | 0 | 37 |
| 2 | 13 | 1 | workers | 120 | 53 | 173 |
| 2 | 13 | 2 | workers | 486 | 80 | 566 |
| 2 | 13 | 3 | workers | 54 | 8 | 62 |
| 2 | 13 | 4 | workers | 54 | 8 | 62 |
| 2 | 14 | 1 | workers | 34 | 72 | 106 |
| 2 | 14 | 2 | workers | 138 | 108 | 246 |
| 2 | 14 | 3 | workers | 15 | 11 | 26 |
| 2 | 14 | 4 | workers | 15 | 11 | 26 |
| 2 | 15 | 1 | workers | 65 | 55 | 120 |
| 2 | 15 | 2 | workers | 262 | 83 | 345 |
| 2 | 15 | 3 | workers | 29 | 8 | 37 |
| 2 | 15 | 4 | workers | 29 | 8 | 37 |
| 2 | 16 | 1 | workers | 93 | 79 | 172 |
| 2 | 16 | 2 | workers | 377 | 119 | 496 |
| 2 | 16 | 3 | workers | 42 | 12 | 54 |
| 2 | 16 | 4 | workers | 42 | 12 | 54 |
| 2 | 17 | 1 | workers | 42 | 46 | 88 |
| 2 | 17 | 2 | workers | 174 | 68 | 242 |
| 2 | 17 | 3 | workers | 19 | 7 | 26 |
| 2 | 17 | 4 | workers | 19 | 7 | 26 |
| 2 | 18 | 1 | workers | 62 | 33 | 95 |
| 2 | 18 | 2 | workers | 250 | 50 | 300 |
| 2 | 18 | 3 | workers | 28 | 5 | 33 |
| 2 | 18 | 4 | workers | 28 | 5 | 33 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 2 | 19 | 1 | workers | 0 | 74 | 74 |
| 2 | 19 | 2 | workers | 0 | 111 | 111 |
| 2 | 19 | 3 | workers | 0 | 11 | 11 |
| 2 | 19 | 4 | workers | 0 | 11 | 11 |
| 2 | 20 | 1 | workers | 0 | 74 | 74 |
| 2 | 20 | 2 | workers | 0 | 112 | 112 |
| 2 | 20 | 3 | workers | 0 | 11 | 11 |
| 2 | 20 | 4 | workers | 0 | 11 | 11 |
| 2 | 21 | 1 | workers | 113 | 0 | 113 |
| 2 | 21 | 2 | workers | 459 | 0 | 459 |
| 2 | 21 | 3 | workers | 51 | 0 | 51 |
| 2 | 21 | 4 | workers | 51 | 0 | 51 |
| 2 | 22 | 1 | workers | 119 | 90 | 209 |
| 2 | 22 | 2 | workers | 484 | 135 | 619 |
| 2 | 22 | 3 | workers | 54 | 14 | 68 |
| 2 | 22 | 4 | workers | 54 | 14 | 68 |
| 2 | 23 | 1 | workers | 10 | 0 | 10 |
| 2 | 23 | 2 | workers | 42 | 0 | 42 |
| 2 | 23 | 3 | workers | 5 | 0 | 5 |
| 2 | 23 | 4 | workers | 5 | 0 | 5 |
| 2 | 24 | 1 | workers | 50 | 0 | 50 |
| 2 | 24 | 2 | workers | 203 | 0 | 203 |
| 2 | 24 | 3 | workers | 23 | 0 | 23 |
| 2 | 24 | 4 | workers | 23 | 0 | 23 |
| 2 | 25 | 1 | workers | 127 | 0 | 127 |
| 2 | 25 | 2 | workers | 514 | 0 | 514 |
| 2 | 25 | 3 | workers | 57 | 0 | 57 |
| 2 | 25 | 4 | workers | 57 | 0 | 57 |
| 3 | 1 | 1 | workers | 185 | 95 | 280 |
| 3 | 1 | 2 | workers | 751 | 144 | 895 |
| 3 | 1 | 3 | workers | 84 | 15 | 99 |
| 3 | 1 | 4 | workers | 84 | 15 | 99 |
| 3 | 2 | 1 | workers | 82 | 79 | 161 |
| 3 | 2 | 2 | workers | 333 | 119 | 452 |
| 3 | 2 | 3 | workers | 37 | 12 | 49 |
| 3 | 2 | 4 | workers | 37 | 12 | 49 |
| 3 | 3 | 1 | ot_1 | 0 | 3 | 3 |
| 3 | 3 | 1 | workers | 112 | 20 | 132 |
| 3 | 3 | 2 | ot_1 | 0 | 5 | 5 |
| 3 | 3 | 2 | ot_2 | 0 | 3 | 3 |
| 3 | 3 | 2 | workers | 458 | 28 | 486 |
| 3 | 3 | 3 | workers | 51 | 3 | 54 |
| 3 | 3 | 4 | workers | 51 | 3 | 54 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 3 | 4 | 1 | ot_1 | 0 | 11 | 11 |
| 3 | 4 | 1 | workers | 120 | 62 | 182 |
| 3 | 4 | 2 | ot_1 | 0 | 16 | 16 |
| 3 | 4 | 2 | ot_2 | 0 | 3 | 3 |
| 3 | 4 | 2 | workers | 488 | 90 | 578 |
| 3 | 4 | 3 | workers | 55 | 13 | 68 |
| 3 | 4 | 4 | workers | 55 | 13 | 68 |
| 3 | 5 | 1 | workers | 8 | 0 | 8 |
| 3 | 5 | 2 | workers | 32 | 0 | 32 |
| 3 | 5 | 3 | workers | 4 | 0 | 4 |
| 3 | 5 | 4 | workers | 4 | 0 | 4 |
| 3 | 6 | 1 | ot_1 | 0 | 13 | 13 |
| 3 | 6 | 1 | ot_2 | 0 | 23 | 23 |
| 3 | 6 | 1 | workers | 107 | 71 | 178 |
| 3 | 6 | 2 | workers | 433 | 175 | 608 |
| 3 | 6 | 3 | workers | 48 | 18 | 66 |
| 3 | 6 | 4 | workers | 48 | 18 | 66 |
| 3 | 7 | 1 | workers | 52 | 29 | 81 |
| 3 | 7 | 2 | workers | 209 | 44 | 253 |
| 3 | 7 | 3 | workers | 23 | 5 | 28 |
| 3 | 7 | 4 | workers | 23 | 5 | 28 |
| 3 | 8 | 1 | workers | 47 | 40 | 87 |
| 3 | 8 | 2 | workers | 190 | 60 | 250 |
| 3 | 8 | 3 | workers | 22 | 6 | 28 |
| 3 | 8 | 4 | workers | 22 | 6 | 28 |
| 3 | 9 | 1 | workers | 50 | 104 | 154 |
| 3 | 9 | 2 | workers | 202 | 157 | 359 |
| 3 | 9 | 3 | workers | 23 | 16 | 39 |
| 3 | 9 | 4 | workers | 23 | 16 | 39 |
| 3 | 10 | 1 | ot_1 | 0 | 3 | 3 |
| 3 | 10 | 1 | workers | 48 | 76 | 124 |
| 3 | 10 | 2 | ot_1 | 0 | 5 | 5 |
| 3 | 10 | 2 | workers | 192 | 114 | 306 |
| 3 | 10 | 3 | workers | 21 | 12 | 33 |
| 3 | 10 | 4 | workers | 21 | 12 | 33 |
| 3 | 11 | 1 | workers | 16 | 43 | 59 |
| 3 | 11 | 2 | workers | 63 | 65 | 128 |
| 3 | 11 | 3 | workers | 7 | 7 | 14 |
| 3 | 11 | 4 | workers | 7 | 7 | 14 |
| 3 | 12 | 1 | workers | 82 | 0 | 82 |
| 3 | 12 | 2 | workers | 334 | 0 | 334 |
| 3 | 12 | 3 | workers | 37 | 0 | 37 |
| 3 | 12 | 4 | workers | 37 | 0 | 37 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 3 | 13 | 1 | fire | 0 | 1 | 1 |
| 3 | 13 | 1 | workers | 120 | 53 | 173 |
| 3 | 13 | 2 | fire | 0 | 1 | 1 |
| 3 | 13 | 2 | workers | 486 | 80 | 566 |
| 3 | 13 | 3 | workers | 54 | 8 | 62 |
| 3 | 13 | 4 | workers | 54 | 8 | 62 |
| 3 | 14 | 1 | workers | 34 | 72 | 106 |
| 3 | 14 | 2 | workers | 138 | 108 | 246 |
| 3 | 14 | 3 | workers | 15 | 11 | 26 |
| 3 | 14 | 4 | workers | 15 | 11 | 26 |
| 3 | 15 | 1 | workers | 65 | 55 | 120 |
| 3 | 15 | 2 | workers | 262 | 83 | 345 |
| 3 | 15 | 3 | workers | 29 | 8 | 37 |
| 3 | 15 | 4 | workers | 29 | 8 | 37 |
| 3 | 16 | 1 | workers | 93 | 79 | 172 |
| 3 | 16 | 2 | workers | 377 | 119 | 496 |
| 3 | 16 | 3 | workers | 42 | 12 | 54 |
| 3 | 16 | 4 | workers | 42 | 12 | 54 |
| 3 | 17 | 1 | ot_1 | 0 | 5 | 5 |
| 3 | 17 | 1 | workers | 42 | 46 | 88 |
| 3 | 17 | 2 | ot_1 | 0 | 9 | 9 |
| 3 | 17 | 2 | workers | 174 | 68 | 242 |
| 3 | 17 | 3 | ot_1 | 0 | 1 | 1 |
| 3 | 17 | 3 | workers | 19 | 7 | 26 |
| 3 | 17 | 4 | ot_1 | 0 | 1 | 1 |
| 3 | 17 | 4 | workers | 19 | 7 | 26 |
| 3 | 18 | 1 | workers | 62 | 33 | 95 |
| 3 | 18 | 2 | workers | 250 | 50 | 300 |
| 3 | 18 | 3 | workers | 28 | 5 | 33 |
| 3 | 18 | 4 | workers | 28 | 5 | 33 |
| 3 | 19 | 1 | workers | 0 | 74 | 74 |
| 3 | 19 | 2 | workers | 0 | 111 | 111 |
| 3 | 19 | 3 | workers | 0 | 11 | 11 |
| 3 | 19 | 4 | workers | 0 | 11 | 11 |
| 3 | 20 | 1 | ot_1 | 0 | 13 | 13 |
| 3 | 20 | 1 | workers | 0 | 74 | 74 |
| 3 | 20 | 2 | ot_1 | 0 | 20 | 20 |
| 3 | 20 | 2 | workers | 0 | 112 | 112 |
| 3 | 20 | 3 | ot_1 | 0 | 2 | 2 |
| 3 | 20 | 3 | workers | 0 | 11 | 11 |
| 3 | 20 | 4 | ot_1 | 0 | 2 | 2 |
| 3 | 20 | 4 | workers | 0 | 11 | 11 |
| 3 | 21 | 1 | workers | 113 | 0 | 113 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 3 | 21 | 2 | workers | 459 | 0 | 459 |
| 3 | 21 | 3 | workers | 51 | 0 | 51 |
| 3 | 21 | 4 | workers | 51 | 0 | 51 |
| 3 | 22 | 1 | workers | 119 | 90 | 209 |
| 3 | 22 | 2 | workers | 484 | 135 | 619 |
| 3 | 22 | 3 | workers | 54 | 14 | 68 |
| 3 | 22 | 4 | workers | 54 | 14 | 68 |
| 3 | 23 | 1 | workers | 10 | 0 | 10 |
| 3 | 23 | 2 | workers | 42 | 0 | 42 |
| 3 | 23 | 3 | workers | 5 | 0 | 5 |
| 3 | 23 | 4 | workers | 5 | 0 | 5 |
| 3 | 24 | 1 | workers | 50 | 0 | 50 |
| 3 | 24 | 2 | workers | 203 | 0 | 203 |
| 3 | 24 | 3 | workers | 23 | 0 | 23 |
| 3 | 24 | 4 | workers | 23 | 0 | 23 |
| 3 | 25 | 1 | workers | 127 | 0 | 127 |
| 3 | 25 | 2 | workers | 514 | 0 | 514 |
| 3 | 25 | 3 | workers | 57 | 0 | 57 |
| 3 | 25 | 4 | workers | 57 | 0 | 57 |
| 4 | 1 | 1 | workers | 185 | 95 | 280 |
| 4 | 1 | 2 | workers | 751 | 144 | 895 |
| 4 | 1 | 3 | workers | 84 | 15 | 99 |
| 4 | 1 | 4 | workers | 84 | 15 | 99 |
| 4 | 2 | 1 | workers | 82 | 79 | 161 |
| 4 | 2 | 2 | workers | 333 | 119 | 452 |
| 4 | 2 | 3 | workers | 37 | 12 | 49 |
| 4 | 2 | 4 | workers | 37 | 12 | 49 |
| 4 | 3 | 1 | workers | 112 | 20 | 132 |
| 4 | 3 | 2 | ot_1 | 0 | 1 | 1 |
| 4 | 3 | 2 | workers | 458 | 28 | 486 |
| 4 | 3 | 3 | workers | 51 | 3 | 54 |
| 4 | 3 | 4 | workers | 51 | 3 | 54 |
| 4 | 4 | 1 | workers | 120 | 62 | 182 |
| 4 | 4 | 2 | workers | 488 | 90 | 578 |
| 4 | 4 | 3 | workers | 55 | 13 | 68 |
| 4 | 4 | 4 | workers | 55 | 13 | 68 |
| 4 | 5 | 1 | workers | 8 | 0 | 8 |
| 4 | 5 | 2 | workers | 32 | 0 | 32 |
| 4 | 5 | 3 | workers | 4 | 0 | 4 |
| 4 | 5 | 4 | workers | 4 | 0 | 4 |
| 4 | 6 | 1 | ot_1 | 0 | 13 | 13 |
| 4 | 6 | 1 | workers | 107 | 71 | 178 |
| 4 | 6 | 2 | workers | 433 | 175 | 608 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 4 | 6 | 3 | workers | 48 | 18 | 66 |
| 4 | 6 | 4 | workers | 48 | 18 | 66 |
| 4 | 7 | 1 | workers | 52 | 29 | 81 |
| 4 | 7 | 2 | workers | 209 | 44 | 253 |
| 4 | 7 | 3 | workers | 23 | 5 | 28 |
| 4 | 7 | 4 | workers | 23 | 5 | 28 |
| 4 | 8 | 1 | workers | 47 | 40 | 87 |
| 4 | 8 | 2 | workers | 190 | 60 | 250 |
| 4 | 8 | 3 | workers | 22 | 6 | 28 |
| 4 | 8 | 4 | workers | 22 | 6 | 28 |
| 4 | 9 | 1 | workers | 50 | 104 | 154 |
| 4 | 9 | 2 | workers | 202 | 157 | 359 |
| 4 | 9 | 3 | workers | 23 | 16 | 39 |
| 4 | 9 | 4 | workers | 23 | 16 | 39 |
| 4 | 10 | 1 | workers | 48 | 76 | 124 |
| 4 | 10 | 2 | workers | 192 | 114 | 306 |
| 4 | 10 | 3 | workers | 21 | 12 | 33 |
| 4 | 10 | 4 | workers | 21 | 12 | 33 |
| 4 | 11 | 1 | workers | 16 | 43 | 59 |
| 4 | 11 | 2 | workers | 63 | 65 | 128 |
| 4 | 11 | 3 | workers | 7 | 7 | 14 |
| 4 | 11 | 4 | workers | 7 | 7 | 14 |
| 4 | 12 | 1 | workers | 82 | 0 | 82 |
| 4 | 12 | 2 | workers | 334 | 0 | 334 |
| 4 | 12 | 3 | workers | 37 | 0 | 37 |
| 4 | 12 | 4 | workers | 37 | 0 | 37 |
| 4 | 13 | 1 | workers | 120 | 52 | 172 |
| 4 | 13 | 2 | workers | 486 | 79 | 565 |
| 4 | 13 | 3 | workers | 54 | 8 | 62 |
| 4 | 13 | 4 | workers | 54 | 8 | 62 |
| 4 | 14 | 1 | workers | 34 | 72 | 106 |
| 4 | 14 | 2 | workers | 138 | 108 | 246 |
| 4 | 14 | 3 | workers | 15 | 11 | 26 |
| 4 | 14 | 4 | workers | 15 | 11 | 26 |
| 4 | 15 | 1 | workers | 65 | 55 | 120 |
| 4 | 15 | 2 | workers | 262 | 83 | 345 |
| 4 | 15 | 3 | workers | 29 | 8 | 37 |
| 4 | 15 | 4 | workers | 29 | 8 | 37 |
| 4 | 16 | 1 | workers | 93 | 79 | 172 |
| 4 | 16 | 2 | workers | 377 | 119 | 496 |
| 4 | 16 | 3 | workers | 42 | 12 | 54 |
| 4 | 16 | 4 | workers | 42 | 12 | 54 |
| 4 | 17 | 1 | workers | 42 | 46 | 88 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 4 | 17 | 2 | workers | 174 | 68 | 242 |
| 4 | 17 | 3 | workers | 19 | 7 | 26 |
| 4 | 17 | 4 | workers | 19 | 7 | 26 |
| 4 | 18 | 1 | workers | 62 | 33 | 95 |
| 4 | 18 | 2 | workers | 250 | 50 | 300 |
| 4 | 18 | 3 | workers | 28 | 5 | 33 |
| 4 | 18 | 4 | workers | 28 | 5 | 33 |
| 4 | 19 | 1 | workers | 0 | 74 | 74 |
| 4 | 19 | 2 | workers | 0 | 111 | 111 |
| 4 | 19 | 3 | workers | 0 | 11 | 11 |
| 4 | 19 | 4 | workers | 0 | 11 | 11 |
| 4 | 20 | 1 | workers | 0 | 74 | 74 |
| 4 | 20 | 2 | workers | 0 | 112 | 112 |
| 4 | 20 | 3 | workers | 0 | 11 | 11 |
| 4 | 20 | 4 | workers | 0 | 11 | 11 |
| 4 | 21 | 1 | workers | 113 | 0 | 113 |
| 4 | 21 | 2 | workers | 459 | 0 | 459 |
| 4 | 21 | 3 | workers | 51 | 0 | 51 |
| 4 | 21 | 4 | workers | 51 | 0 | 51 |
| 4 | 22 | 1 | workers | 119 | 90 | 209 |
| 4 | 22 | 2 | workers | 484 | 135 | 619 |
| 4 | 22 | 3 | workers | 54 | 14 | 68 |
| 4 | 22 | 4 | workers | 54 | 14 | 68 |
| 4 | 23 | 1 | workers | 10 | 0 | 10 |
| 4 | 23 | 2 | workers | 42 | 0 | 42 |
| 4 | 23 | 3 | workers | 5 | 0 | 5 |
| 4 | 23 | 4 | workers | 5 | 0 | 5 |
| 4 | 24 | 1 | workers | 50 | 0 | 50 |
| 4 | 24 | 2 | workers | 203 | 0 | 203 |
| 4 | 24 | 3 | workers | 23 | 0 | 23 |
| 4 | 24 | 4 | workers | 23 | 0 | 23 |
| 4 | 25 | 1 | workers | 127 | 0 | 127 |
| 4 | 25 | 2 | workers | 514 | 0 | 514 |
| 4 | 25 | 3 | workers | 57 | 0 | 57 |
| 4 | 25 | 4 | workers | 57 | 0 | 57 |
| 5 | 1 | 1 | workers | 185 | 95 | 280 |
| 5 | 1 | 2 | workers | 751 | 144 | 895 |
| 5 | 1 | 3 | workers | 84 | 15 | 99 |
| 5 | 1 | 4 | workers | 84 | 15 | 99 |
| 5 | 2 | 1 | workers | 82 | 79 | 161 |
| 5 | 2 | 2 | workers | 333 | 119 | 452 |
| 5 | 2 | 3 | workers | 37 | 12 | 49 |
| 5 | 2 | 4 | workers | 37 | 12 | 49 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 5 | 3 | 1 | ot_1 | 0 | 3 | 3 |
| 5 | 3 | 1 | ot_2 | 0 | 3 | 3 |
| 5 | 3 | 1 | workers | 112 | 20 | 132 |
| 5 | 3 | 2 | ot_1 | 0 | 5 | 5 |
| 5 | 3 | 2 | ot_2 | 0 | 7 | 7 |
| 5 | 3 | 2 | workers | 458 | 28 | 486 |
| 5 | 3 | 3 | workers | 51 | 3 | 54 |
| 5 | 3 | 4 | workers | 51 | 3 | 54 |
| 5 | 4 | 1 | ot_1 | 0 | 11 | 11 |
| 5 | 4 | 1 | ot_2 | 0 | 7 | 7 |
| 5 | 4 | 1 | workers | 120 | 62 | 182 |
| 5 | 4 | 2 | ot_1 | 0 | 16 | 16 |
| 5 | 4 | 2 | ot_2 | 0 | 15 | 15 |
| 5 | 4 | 2 | workers | 488 | 90 | 578 |
| 5 | 4 | 3 | workers | 55 | 13 | 68 |
| 5 | 4 | 4 | workers | 55 | 13 | 68 |
| 5 | 5 | 1 | workers | 8 | 0 | 8 |
| 5 | 5 | 2 | workers | 32 | 0 | 32 |
| 5 | 5 | 3 | workers | 4 | 0 | 4 |
| 5 | 5 | 4 | workers | 4 | 0 | 4 |
| 5 | 6 | 1 | ot_1 | 0 | 13 | 13 |
| 5 | 6 | 1 | ot_2 | 0 | 31 | 31 |
| 5 | 6 | 1 | workers | 107 | 71 | 178 |
| 5 | 6 | 2 | workers | 433 | 175 | 608 |
| 5 | 6 | 3 | workers | 48 | 18 | 66 |
| 5 | 6 | 4 | workers | 48 | 18 | 66 |
| 5 | 7 | 1 | workers | 52 | 29 | 81 |
| 5 | 7 | 2 | workers | 209 | 44 | 253 |
| 5 | 7 | 3 | workers | 23 | 5 | 28 |
| 5 | 7 | 4 | workers | 23 | 5 | 28 |
| 5 | 8 | 1 | workers | 47 | 40 | 87 |
| 5 | 8 | 2 | workers | 190 | 60 | 250 |
| 5 | 8 | 3 | workers | 22 | 6 | 28 |
| 5 | 8 | 4 | workers | 22 | 6 | 28 |
| 5 | 9 | 1 | workers | 50 | 104 | 154 |
| 5 | 9 | 2 | workers | 202 | 157 | 359 |
| 5 | 9 | 3 | workers | 23 | 16 | 39 |
| 5 | 9 | 4 | workers | 23 | 16 | 39 |
| 5 | 10 | 1 | workers | 48 | 76 | 124 |
| 5 | 10 | 2 | workers | 192 | 114 | 306 |
| 5 | 10 | 3 | workers | 21 | 12 | 33 |
| 5 | 10 | 4 | workers | 21 | 12 | 33 |
| 5 | 11 | 1 | workers | 16 | 43 | 59 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 5 | 11 | 2 | workers | 63 | 65 | 128 |
| 5 | 11 | 3 | workers | 7 | 7 | 14 |
| 5 | 11 | 4 | workers | 7 | 7 | 14 |
| 5 | 12 | 1 | workers | 82 | 0 | 82 |
| 5 | 12 | 2 | workers | 334 | 0 | 334 |
| 5 | 12 | 3 | workers | 37 | 0 | 37 |
| 5 | 12 | 4 | workers | 37 | 0 | 37 |
| 5 | 13 | 1 | workers | 120 | 52 | 172 |
| 5 | 13 | 2 | workers | 486 | 79 | 565 |
| 5 | 13 | 3 | workers | 54 | 8 | 62 |
| 5 | 13 | 4 | workers | 54 | 8 | 62 |
| 5 | 14 | 1 | workers | 34 | 72 | 106 |
| 5 | 14 | 2 | workers | 138 | 108 | 246 |
| 5 | 14 | 3 | workers | 15 | 11 | 26 |
| 5 | 14 | 4 | workers | 15 | 11 | 26 |
| 5 | 15 | 1 | workers | 65 | 55 | 120 |
| 5 | 15 | 2 | workers | 262 | 83 | 345 |
| 5 | 15 | 3 | workers | 29 | 8 | 37 |
| 5 | 15 | 4 | workers | 29 | 8 | 37 |
| 5 | 16 | 1 | workers | 93 | 79 | 172 |
| 5 | 16 | 2 | workers | 377 | 119 | 496 |
| 5 | 16 | 3 | workers | 42 | 12 | 54 |
| 5 | 16 | 4 | workers | 42 | 12 | 54 |
| 5 | 17 | 1 | workers | 42 | 46 | 88 |
| 5 | 17 | 2 | workers | 174 | 68 | 242 |
| 5 | 17 | 3 | workers | 19 | 7 | 26 |
| 5 | 17 | 4 | workers | 19 | 7 | 26 |
| 5 | 18 | 1 | workers | 62 | 33 | 95 |
| 5 | 18 | 2 | workers | 250 | 50 | 300 |
| 5 | 18 | 3 | workers | 28 | 5 | 33 |
| 5 | 18 | 4 | workers | 28 | 5 | 33 |
| 5 | 19 | 1 | workers | 0 | 74 | 74 |
| 5 | 19 | 2 | workers | 0 | 111 | 111 |
| 5 | 19 | 3 | workers | 0 | 11 | 11 |
| 5 | 19 | 4 | workers | 0 | 11 | 11 |
| 5 | 20 | 1 | ot_1 | 0 | 10 | 10 |
| 5 | 20 | 1 | workers | 0 | 74 | 74 |
| 5 | 20 | 2 | ot_1 | 0 | 16 | 16 |
| 5 | 20 | 2 | workers | 0 | 112 | 112 |
| 5 | 20 | 3 | ot_1 | 0 | 2 | 2 |
| 5 | 20 | 3 | workers | 0 | 11 | 11 |
| 5 | 20 | 4 | ot_1 | 0 | 2 | 2 |
| 5 | 20 | 4 | workers | 0 | 11 | 11 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 5 | 21 | 1 | workers | 113 | 0 | 113 |
| 5 | 21 | 2 | workers | 459 | 0 | 459 |
| 5 | 21 | 3 | workers | 51 | 0 | 51 |
| 5 | 21 | 4 | workers | 51 | 0 | 51 |
| 5 | 22 | 1 | workers | 119 | 90 | 209 |
| 5 | 22 | 2 | workers | 484 | 135 | 619 |
| 5 | 22 | 3 | workers | 54 | 14 | 68 |
| 5 | 22 | 4 | workers | 54 | 14 | 68 |
| 5 | 23 | 1 | workers | 10 | 0 | 10 |
| 5 | 23 | 2 | workers | 42 | 0 | 42 |
| 5 | 23 | 3 | workers | 5 | 0 | 5 |
| 5 | 23 | 4 | workers | 5 | 0 | 5 |
| 5 | 24 | 1 | workers | 50 | 0 | 50 |
| 5 | 24 | 2 | workers | 203 | 0 | 203 |
| 5 | 24 | 3 | workers | 23 | 0 | 23 |
| 5 | 24 | 4 | workers | 23 | 0 | 23 |
| 5 | 25 | 1 | workers | 127 | 0 | 127 |
| 5 | 25 | 2 | workers | 514 | 0 | 514 |
| 5 | 25 | 3 | workers | 57 | 0 | 57 |
| 5 | 25 | 4 | workers | 57 | 0 | 57 |
| 6 | 1 | 1 | ot_1 | 0 | 4 | 4 |
| 6 | 1 | 1 | workers | 185 | 95 | 280 |
| 6 | 1 | 2 | ot_1 | 0 | 6 | 6 |
| 6 | 1 | 2 | workers | 751 | 144 | 895 |
| 6 | 1 | 3 | workers | 84 | 15 | 99 |
| 6 | 1 | 4 | workers | 84 | 15 | 99 |
| 6 | 2 | 1 | ot_1 | 0 | 11 | 11 |
| 6 | 2 | 1 | workers | 82 | 79 | 161 |
| 6 | 2 | 2 | ot_1 | 0 | 17 | 17 |
| 6 | 2 | 2 | workers | 333 | 119 | 452 |
| 6 | 2 | 3 | ot_1 | 0 | 2 | 2 |
| 6 | 2 | 3 | workers | 37 | 12 | 49 |
| 6 | 2 | 4 | ot_1 | 0 | 2 | 2 |
| 6 | 2 | 4 | workers | 37 | 12 | 49 |
| 6 | 3 | 1 | ot_1 | 0 | 3 | 3 |
| 6 | 3 | 1 | ot_2 | 0 | 5 | 5 |
| 6 | 3 | 1 | workers | 112 | 20 | 132 |
| 6 | 3 | 2 | ot_1 | 0 | 5 | 5 |
| 6 | 3 | 2 | ot_2 | 0 | 11 | 11 |
| 6 | 3 | 2 | workers | 458 | 28 | 486 |
| 6 | 3 | 3 | ot_2 | 0 | 1 | 1 |
| 6 | 3 | 3 | workers | 51 | 3 | 54 |
| 6 | 3 | 4 | ot_2 | 0 | 1 | 1 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 6 | 3 | 4 | workers | 51 | 3 | 54 |
| 6 | 4 | 1 | ot_1 | 0 | 11 | 11 |
| 6 | 4 | 1 | ot_2 | 0 | 28 | 28 |
| 6 | 4 | 1 | workers | 120 | 62 | 182 |
| 6 | 4 | 2 | ot_1 | 0 | 16 | 16 |
| 6 | 4 | 2 | ot_2 | 0 | 47 | 47 |
| 6 | 4 | 2 | workers | 488 | 90 | 578 |
| 6 | 4 | 3 | ot_1 | 0 | 1 | 1 |
| 6 | 4 | 3 | ot_2 | 0 | 1 | 1 |
| 6 | 4 | 3 | workers | 55 | 13 | 68 |
| 6 | 4 | 4 | ot_1 | 0 | 1 | 1 |
| 6 | 4 | 4 | ot_2 | 0 | 1 | 1 |
| 6 | 4 | 4 | workers | 55 | 13 | 68 |
| 6 | 5 | 1 | workers | 8 | 0 | 8 |
| 6 | 5 | 2 | workers | 32 | 0 | 32 |
| 6 | 5 | 3 | workers | 4 | 0 | 4 |
| 6 | 5 | 4 | workers | 4 | 0 | 4 |
| 6 | 6 | 1 | ot_1 | 0 | 13 | 13 |
| 6 | 6 | 1 | ot_2 | 0 | 58 | 58 |
| 6 | 6 | 1 | workers | 107 | 71 | 178 |
| 6 | 6 | 2 | fire | 0 | 9 | 9 |
| 6 | 6 | 2 | ot_1 | 0 | 32 | 32 |
| 6 | 6 | 2 | ot_2 | 0 | 7 | 7 |
| 6 | 6 | 2 | workers | 433 | 175 | 608 |
| 6 | 6 | 3 | fire | 0 | 1 | 1 |
| 6 | 6 | 3 | ot_1 | 0 | 3 | 3 |
| 6 | 6 | 3 | workers | 48 | 18 | 66 |
| 6 | 6 | 4 | fire | 0 | 1 | 1 |
| 6 | 6 | 4 | ot_1 | 0 | 3 | 3 |
| 6 | 6 | 4 | workers | 48 | 18 | 66 |
| 6 | 7 | 1 | ot_1 | 0 | 2 | 2 |
| 6 | 7 | 1 | workers | 52 | 29 | 81 |
| 6 | 7 | 2 | ot_1 | 0 | 3 | 3 |
| 6 | 7 | 2 | workers | 209 | 44 | 253 |
| 6 | 7 | 3 | workers | 23 | 5 | 28 |
| 6 | 7 | 4 | workers | 23 | 5 | 28 |
| 6 | 8 | 1 | ot_1 | 0 | 2 | 2 |
| 6 | 8 | 1 | workers | 47 | 40 | 87 |
| 6 | 8 | 2 | ot_1 | 0 | 3 | 3 |
| 6 | 8 | 2 | workers | 190 | 60 | 250 |
| 6 | 8 | 3 | workers | 22 | 6 | 28 |
| 6 | 8 | 4 | workers | 22 | 6 | 28 |
| 6 | 9 | 1 | ot_1 | 0 | 4 | 4 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 6 | 9 | 1 | workers | 50 | 104 | 154 |
| 6 | 9 | 2 | ot_1 | 0 | 7 | 7 |
| 6 | 9 | 2 | workers | 202 | 157 | 359 |
| 6 | 9 | 3 | workers | 23 | 16 | 39 |
| 6 | 9 | 4 | workers | 23 | 16 | 39 |
| 6 | 10 | 1 | ot_1 | 0 | 3 | 3 |
| 6 | 10 | 1 | workers | 48 | 76 | 124 |
| 6 | 10 | 2 | ot_1 | 0 | 5 | 5 |
| 6 | 10 | 2 | workers | 192 | 114 | 306 |
| 6 | 10 | 3 | workers | 21 | 12 | 33 |
| 6 | 10 | 4 | workers | 21 | 12 | 33 |
| 6 | 11 | 1 | workers | 16 | 43 | 59 |
| 6 | 11 | 2 | workers | 63 | 65 | 128 |
| 6 | 11 | 3 | workers | 7 | 7 | 14 |
| 6 | 11 | 4 | workers | 7 | 7 | 14 |
| 6 | 12 | 1 | workers | 82 | 0 | 82 |
| 6 | 12 | 2 | workers | 334 | 0 | 334 |
| 6 | 12 | 3 | workers | 37 | 0 | 37 |
| 6 | 12 | 4 | workers | 37 | 0 | 37 |
| 6 | 13 | 1 | fire | 0 | 1 | 1 |
| 6 | 13 | 1 | ot_1 | 0 | 9 | 9 |
| 6 | 13 | 1 | workers | 120 | 52 | 172 |
| 6 | 13 | 2 | fire | 0 | 3 | 3 |
| 6 | 13 | 2 | ot_1 | 0 | 14 | 14 |
| 6 | 13 | 2 | workers | 486 | 79 | 565 |
| 6 | 13 | 3 | ot_1 | 0 | 1 | 1 |
| 6 | 13 | 3 | workers | 54 | 8 | 62 |
| 6 | 13 | 4 | ot_1 | 0 | 1 | 1 |
| 6 | 13 | 4 | workers | 54 | 8 | 62 |
| 6 | 14 | 1 | ot_1 | 0 | 5 | 5 |
| 6 | 14 | 1 | workers | 34 | 72 | 106 |
| 6 | 14 | 2 | ot_1 | 0 | 9 | 9 |
| 6 | 14 | 2 | workers | 138 | 108 | 246 |
| 6 | 14 | 3 | ot_1 | 0 | 1 | 1 |
| 6 | 14 | 3 | workers | 15 | 11 | 26 |
| 6 | 14 | 4 | ot_1 | 0 | 1 | 1 |
| 6 | 14 | 4 | workers | 15 | 11 | 26 |
| 6 | 15 | 1 | workers | 65 | 55 | 120 |
| 6 | 15 | 2 | workers | 262 | 83 | 345 |
| 6 | 15 | 3 | workers | 29 | 8 | 37 |
| 6 | 15 | 4 | workers | 29 | 8 | 37 |
| 6 | 16 | 1 | ot_1 | 0 | 11 | 11 |
| 6 | 16 | 1 | workers | 93 | 79 | 172 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 6 | 16 | 2 | ot_1 | 0 | 17 | 17 |
| 6 | 16 | 2 | workers | 377 | 119 | 496 |
| 6 | 16 | 3 | ot_1 | 0 | 2 | 2 |
| 6 | 16 | 3 | workers | 42 | 12 | 54 |
| 6 | 16 | 4 | ot_1 | 0 | 2 | 2 |
| 6 | 16 | 4 | workers | 42 | 12 | 54 |
| 6 | 17 | 1 | ot_1 | 0 | 4 | 4 |
| 6 | 17 | 1 | workers | 42 | 46 | 88 |
| 6 | 17 | 2 | ot_1 | 0 | 8 | 8 |
| 6 | 17 | 2 | workers | 174 | 68 | 242 |
| 6 | 17 | 3 | workers | 19 | 7 | 26 |
| 6 | 17 | 4 | workers | 19 | 7 | 26 |
| 6 | 18 | 1 | workers | 62 | 33 | 95 |
| 6 | 18 | 2 | workers | 250 | 50 | 300 |
| 6 | 18 | 3 | workers | 28 | 5 | 33 |
| 6 | 18 | 4 | workers | 28 | 5 | 33 |
| 6 | 19 | 1 | ot_1 | 0 | 3 | 3 |
| 6 | 19 | 1 | workers | 0 | 74 | 74 |
| 6 | 19 | 2 | ot_1 | 0 | 5 | 5 |
| 6 | 19 | 2 | workers | 0 | 111 | 111 |
| 6 | 19 | 3 | workers | 0 | 11 | 11 |
| 6 | 19 | 4 | workers | 0 | 11 | 11 |
| 6 | 20 | 1 | ot_1 | 0 | 13 | 13 |
| 6 | 20 | 1 | ot_2 | 0 | 29 | 29 |
| 6 | 20 | 1 | workers | 0 | 74 | 74 |
| 6 | 20 | 2 | ot_1 | 0 | 21 | 21 |
| 6 | 20 | 2 | ot_2 | 0 | 44 | 44 |
| 6 | 20 | 2 | workers | 0 | 112 | 112 |
| 6 | 20 | 3 | ot_1 | 0 | 2 | 2 |
| 6 | 20 | 3 | ot_2 | 0 | 5 | 5 |
| 6 | 20 | 3 | workers | 0 | 11 | 11 |
| 6 | 20 | 4 | ot_1 | 0 | 2 | 2 |
| 6 | 20 | 4 | ot_2 | 0 | 5 | 5 |
| 6 | 20 | 4 | workers | 0 | 11 | 11 |
| 6 | 21 | 1 | workers | 113 | 0 | 113 |
| 6 | 21 | 2 | workers | 459 | 0 | 459 |
| 6 | 21 | 3 | workers | 51 | 0 | 51 |
| 6 | 21 | 4 | workers | 51 | 0 | 51 |
| 6 | 22 | 1 | ot_1 | 0 | 5 | 5 |
| 6 | 22 | 1 | workers | 119 | 90 | 209 |
| 6 | 22 | 2 | ot_1 | 0 | 9 | 9 |
| 6 | 22 | 2 | workers | 484 | 135 | 619 |
| 6 | 22 | 3 | workers | 54 | 14 | 68 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 6 | 22 | 4 | workers | 54 | 14 | 68 |
| 6 | 23 | 1 | workers | 10 | 0 | 10 |
| 6 | 23 | 2 | workers | 42 | 0 | 42 |
| 6 | 23 | 3 | workers | 5 | 0 | 5 |
| 6 | 23 | 4 | workers | 5 | 0 | 5 |
| 6 | 24 | 1 | workers | 50 | 0 | 50 |
| 6 | 24 | 2 | workers | 203 | 0 | 203 |
| 6 | 24 | 3 | workers | 23 | 0 | 23 |
| 6 | 24 | 4 | workers | 23 | 0 | 23 |
| 6 | 25 | 1 | workers | 127 | 0 | 127 |
| 6 | 25 | 2 | workers | 514 | 0 | 514 |
| 6 | 25 | 3 | workers | 57 | 0 | 57 |
| 6 | 25 | 4 | workers | 57 | 0 | 57 |
| 7 | 1 | 1 | workers | 185 | 95 | 280 |
| 7 | 1 | 2 | workers | 751 | 144 | 895 |
| 7 | 1 | 3 | workers | 84 | 15 | 99 |
| 7 | 1 | 4 | workers | 84 | 15 | 99 |
| 7 | 2 | 1 | workers | 82 | 79 | 161 |
| 7 | 2 | 2 | workers | 333 | 119 | 452 |
| 7 | 2 | 3 | workers | 37 | 12 | 49 |
| 7 | 2 | 4 | workers | 37 | 12 | 49 |
| 7 | 3 | 1 | workers | 112 | 20 | 132 |
| 7 | 3 | 2 | ot_1 | 0 | 1 | 1 |
| 7 | 3 | 2 | workers | 458 | 28 | 486 |
| 7 | 3 | 3 | workers | 51 | 3 | 54 |
| 7 | 3 | 4 | workers | 51 | 3 | 54 |
| 7 | 4 | 1 | ot_1 | 0 | 4 | 4 |
| 7 | 4 | 1 | workers | 120 | 62 | 182 |
| 7 | 4 | 2 | ot_1 | 0 | 11 | 11 |
| 7 | 4 | 2 | workers | 488 | 90 | 578 |
| 7 | 4 | 3 | workers | 55 | 13 | 68 |
| 7 | 4 | 4 | workers | 55 | 13 | 68 |
| 7 | 5 | 1 | workers | 8 | 0 | 8 |
| 7 | 5 | 2 | workers | 32 | 0 | 32 |
| 7 | 5 | 3 | workers | 4 | 0 | 4 |
| 7 | 5 | 4 | workers | 4 | 0 | 4 |
| 7 | 6 | 1 | ot_1 | 0 | 13 | 13 |
| 7 | 6 | 1 | ot_2 | 0 | 4 | 4 |
| 7 | 6 | 1 | workers | 107 | 71 | 178 |
| 7 | 6 | 2 | workers | 433 | 166 | 599 |
| 7 | 6 | 3 | workers | 48 | 17 | 65 |
| 7 | 6 | 4 | workers | 48 | 17 | 65 |
| 7 | 7 | 1 | workers | 52 | 29 | 81 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 7 | 7 | 2 | workers | 209 | 44 | 253 |
| 7 | 7 | 3 | workers | 23 | 5 | 28 |
| 7 | 7 | 4 | workers | 23 | 5 | 28 |
| 7 | 8 | 1 | workers | 47 | 40 | 87 |
| 7 | 8 | 2 | workers | 190 | 60 | 250 |
| 7 | 8 | 3 | workers | 22 | 6 | 28 |
| 7 | 8 | 4 | workers | 22 | 6 | 28 |
| 7 | 9 | 1 | workers | 50 | 104 | 154 |
| 7 | 9 | 2 | workers | 202 | 157 | 359 |
| 7 | 9 | 3 | workers | 23 | 16 | 39 |
| 7 | 9 | 4 | workers | 23 | 16 | 39 |
| 7 | 10 | 1 | workers | 48 | 76 | 124 |
| 7 | 10 | 2 | workers | 192 | 114 | 306 |
| 7 | 10 | 3 | workers | 21 | 12 | 33 |
| 7 | 10 | 4 | workers | 21 | 12 | 33 |
| 7 | 11 | 1 | workers | 16 | 43 | 59 |
| 7 | 11 | 2 | workers | 63 | 65 | 128 |
| 7 | 11 | 3 | workers | 7 | 7 | 14 |
| 7 | 11 | 4 | workers | 7 | 7 | 14 |
| 7 | 12 | 1 | workers | 82 | 0 | 82 |
| 7 | 12 | 2 | workers | 334 | 0 | 334 |
| 7 | 12 | 3 | workers | 37 | 0 | 37 |
| 7 | 12 | 4 | workers | 37 | 0 | 37 |
| 7 | 13 | 1 | workers | 120 | 51 | 171 |
| 7 | 13 | 2 | workers | 486 | 76 | 562 |
| 7 | 13 | 3 | workers | 54 | 8 | 62 |
| 7 | 13 | 4 | workers | 54 | 8 | 62 |
| 7 | 14 | 1 | workers | 34 | 72 | 106 |
| 7 | 14 | 2 | workers | 138 | 108 | 246 |
| 7 | 14 | 3 | workers | 15 | 11 | 26 |
| 7 | 14 | 4 | workers | 15 | 11 | 26 |
| 7 | 15 | 1 | workers | 65 | 55 | 120 |
| 7 | 15 | 2 | workers | 262 | 83 | 345 |
| 7 | 15 | 3 | workers | 29 | 8 | 37 |
| 7 | 15 | 4 | workers | 29 | 8 | 37 |
| 7 | 16 | 1 | workers | 93 | 79 | 172 |
| 7 | 16 | 2 | workers | 377 | 119 | 496 |
| 7 | 16 | 3 | workers | 42 | 12 | 54 |
| 7 | 16 | 4 | workers | 42 | 12 | 54 |
| 7 | 17 | 1 | workers | 42 | 46 | 88 |
| 7 | 17 | 2 | workers | 174 | 68 | 242 |
| 7 | 17 | 3 | workers | 19 | 7 | 26 |
| 7 | 17 | 4 | workers | 19 | 7 | 26 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 7 | 18 | 1 | workers | 62 | 33 | 95 |
| 7 | 18 | 2 | workers | 250 | 50 | 300 |
| 7 | 18 | 3 | workers | 28 | 5 | 33 |
| 7 | 18 | 4 | workers | 28 | 5 | 33 |
| 7 | 19 | 1 | workers | 0 | 74 | 74 |
| 7 | 19 | 2 | workers | 0 | 111 | 111 |
| 7 | 19 | 3 | workers | 0 | 11 | 11 |
| 7 | 19 | 4 | workers | 0 | 11 | 11 |
| 7 | 20 | 1 | workers | 0 | 74 | 74 |
| 7 | 20 | 2 | workers | 0 | 112 | 112 |
| 7 | 20 | 3 | workers | 0 | 11 | 11 |
| 7 | 20 | 4 | workers | 0 | 11 | 11 |
| 7 | 21 | 1 | workers | 113 | 0 | 113 |
| 7 | 21 | 2 | workers | 459 | 0 | 459 |
| 7 | 21 | 3 | workers | 51 | 0 | 51 |
| 7 | 21 | 4 | workers | 51 | 0 | 51 |
| 7 | 22 | 1 | workers | 119 | 90 | 209 |
| 7 | 22 | 2 | workers | 484 | 135 | 619 |
| 7 | 22 | 3 | workers | 54 | 14 | 68 |
| 7 | 22 | 4 | workers | 54 | 14 | 68 |
| 7 | 23 | 1 | workers | 10 | 0 | 10 |
| 7 | 23 | 2 | workers | 42 | 0 | 42 |
| 7 | 23 | 3 | workers | 5 | 0 | 5 |
| 7 | 23 | 4 | workers | 5 | 0 | 5 |
| 7 | 24 | 1 | workers | 50 | 0 | 50 |
| 7 | 24 | 2 | workers | 203 | 0 | 203 |
| 7 | 24 | 3 | workers | 23 | 0 | 23 |
| 7 | 24 | 4 | workers | 23 | 0 | 23 |
| 7 | 25 | 1 | workers | 127 | 0 | 127 |
| 7 | 25 | 2 | workers | 514 | 0 | 514 |
| 7 | 25 | 3 | workers | 57 | 0 | 57 |
| 7 | 25 | 4 | workers | 57 | 0 | 57 |
| 8 | 1 | 1 | workers | 185 | 95 | 280 |
| 8 | 1 | 2 | workers | 751 | 144 | 895 |
| 8 | 1 | 3 | workers | 84 | 15 | 99 |
| 8 | 1 | 4 | workers | 84 | 15 | 99 |
| 8 | 2 | 1 | workers | 82 | 79 | 161 |
| 8 | 2 | 2 | workers | 333 | 119 | 452 |
| 8 | 2 | 3 | workers | 37 | 12 | 49 |
| 8 | 2 | 4 | workers | 37 | 12 | 49 |
| 8 | 3 | 1 | workers | 112 | 20 | 132 |
| 8 | 3 | 2 | ot_1 | 0 | 1 | 1 |
| 8 | 3 | 2 | workers | 458 | 28 | 486 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 8 | 3 | 3 | workers | 51 | 3 | 54 |
| 8 | 3 | 4 | workers | 51 | 3 | 54 |
| 8 | 4 | 1 | ot_1 | 0 | 5 | 5 |
| 8 | 4 | 1 | workers | 120 | 62 | 182 |
| 8 | 4 | 2 | ot_1 | 0 | 11 | 11 |
| 8 | 4 | 2 | workers | 488 | 90 | 578 |
| 8 | 4 | 3 | workers | 55 | 13 | 68 |
| 8 | 4 | 4 | workers | 55 | 13 | 68 |
| 8 | 5 | 1 | workers | 8 | 0 | 8 |
| 8 | 5 | 2 | workers | 32 | 0 | 32 |
| 8 | 5 | 3 | workers | 4 | 0 | 4 |
| 8 | 5 | 4 | workers | 4 | 0 | 4 |
| 8 | 6 | 1 | ot_1 | 0 | 13 | 13 |
| 8 | 6 | 1 | ot_2 | 0 | 2 | 2 |
| 8 | 6 | 1 | workers | 107 | 71 | 178 |
| 8 | 6 | 2 | workers | 433 | 166 | 599 |
| 8 | 6 | 3 | workers | 48 | 17 | 65 |
| 8 | 6 | 4 | workers | 48 | 17 | 65 |
| 8 | 7 | 1 | workers | 52 | 29 | 81 |
| 8 | 7 | 2 | workers | 209 | 44 | 253 |
| 8 | 7 | 3 | workers | 23 | 5 | 28 |
| 8 | 7 | 4 | workers | 23 | 5 | 28 |
| 8 | 8 | 1 | workers | 47 | 40 | 87 |
| 8 | 8 | 2 | workers | 190 | 60 | 250 |
| 8 | 8 | 3 | workers | 22 | 6 | 28 |
| 8 | 8 | 4 | workers | 22 | 6 | 28 |
| 8 | 9 | 1 | workers | 50 | 104 | 154 |
| 8 | 9 | 2 | workers | 202 | 157 | 359 |
| 8 | 9 | 3 | workers | 23 | 16 | 39 |
| 8 | 9 | 4 | workers | 23 | 16 | 39 |
| 8 | 10 | 1 | workers | 48 | 76 | 124 |
| 8 | 10 | 2 | workers | 192 | 114 | 306 |
| 8 | 10 | 3 | workers | 21 | 12 | 33 |
| 8 | 10 | 4 | workers | 21 | 12 | 33 |
| 8 | 11 | 1 | workers | 16 | 43 | 59 |
| 8 | 11 | 2 | workers | 63 | 65 | 128 |
| 8 | 11 | 3 | workers | 7 | 7 | 14 |
| 8 | 11 | 4 | workers | 7 | 7 | 14 |
| 8 | 12 | 1 | workers | 82 | 0 | 82 |
| 8 | 12 | 2 | workers | 334 | 0 | 334 |
| 8 | 12 | 3 | workers | 37 | 0 | 37 |
| 8 | 12 | 4 | workers | 37 | 0 | 37 |
| 8 | 13 | 1 | workers | 120 | 51 | 171 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 8 | 13 | 2 | workers | 486 | 76 | 562 |
| 8 | 13 | 3 | workers | 54 | 8 | 62 |
| 8 | 13 | 4 | workers | 54 | 8 | 62 |
| 8 | 14 | 1 | workers | 34 | 72 | 106 |
| 8 | 14 | 2 | workers | 138 | 108 | 246 |
| 8 | 14 | 3 | workers | 15 | 11 | 26 |
| 8 | 14 | 4 | workers | 15 | 11 | 26 |
| 8 | 15 | 1 | workers | 65 | 55 | 120 |
| 8 | 15 | 2 | workers | 262 | 83 | 345 |
| 8 | 15 | 3 | workers | 29 | 8 | 37 |
| 8 | 15 | 4 | workers | 29 | 8 | 37 |
| 8 | 16 | 1 | workers | 93 | 79 | 172 |
| 8 | 16 | 2 | workers | 377 | 119 | 496 |
| 8 | 16 | 3 | workers | 42 | 12 | 54 |
| 8 | 16 | 4 | workers | 42 | 12 | 54 |
| 8 | 17 | 1 | workers | 42 | 46 | 88 |
| 8 | 17 | 2 | workers | 174 | 68 | 242 |
| 8 | 17 | 3 | workers | 19 | 7 | 26 |
| 8 | 17 | 4 | workers | 19 | 7 | 26 |
| 8 | 18 | 1 | workers | 62 | 33 | 95 |
| 8 | 18 | 2 | workers | 250 | 50 | 300 |
| 8 | 18 | 3 | workers | 28 | 5 | 33 |
| 8 | 18 | 4 | workers | 28 | 5 | 33 |
| 8 | 19 | 1 | workers | 0 | 74 | 74 |
| 8 | 19 | 2 | workers | 0 | 111 | 111 |
| 8 | 19 | 3 | workers | 0 | 11 | 11 |
| 8 | 19 | 4 | workers | 0 | 11 | 11 |
| 8 | 20 | 1 | ot_1 | 0 | 1 | 1 |
| 8 | 20 | 1 | workers | 0 | 74 | 74 |
| 8 | 20 | 2 | ot_1 | 0 | 1 | 1 |
| 8 | 20 | 2 | workers | 0 | 112 | 112 |
| 8 | 20 | 3 | workers | 0 | 11 | 11 |
| 8 | 20 | 4 | workers | 0 | 11 | 11 |
| 8 | 21 | 1 | workers | 113 | 0 | 113 |
| 8 | 21 | 2 | workers | 459 | 0 | 459 |
| 8 | 21 | 3 | workers | 51 | 0 | 51 |
| 8 | 21 | 4 | workers | 51 | 0 | 51 |
| 8 | 22 | 1 | workers | 119 | 90 | 209 |
| 8 | 22 | 2 | workers | 484 | 135 | 619 |
| 8 | 22 | 3 | workers | 54 | 14 | 68 |
| 8 | 22 | 4 | workers | 54 | 14 | 68 |
| 8 | 23 | 1 | workers | 10 | 0 | 10 |
| 8 | 23 | 2 | workers | 42 | 0 | 42 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 8 | 23 | 3 | workers | 5 | 0 | 5 |
| 8 | 23 | 4 | workers | 5 | 0 | 5 |
| 8 | 24 | 1 | workers | 50 | 0 | 50 |
| 8 | 24 | 2 | workers | 203 | 0 | 203 |
| 8 | 24 | 3 | workers | 23 | 0 | 23 |
| 8 | 24 | 4 | workers | 23 | 0 | 23 |
| 8 | 25 | 1 | workers | 127 | 0 | 127 |
| 8 | 25 | 2 | workers | 514 | 0 | 514 |
| 8 | 25 | 3 | workers | 57 | 0 | 57 |
| 8 | 25 | 4 | workers | 57 | 0 | 57 |
| 9 | 1 | 1 | workers | 185 | 95 | 280 |
| 9 | 1 | 2 | workers | 751 | 144 | 895 |
| 9 | 1 | 3 | workers | 84 | 15 | 99 |
| 9 | 1 | 4 | workers | 84 | 15 | 99 |
| 9 | 2 | 1 | workers | 82 | 79 | 161 |
| 9 | 2 | 2 | workers | 333 | 119 | 452 |
| 9 | 2 | 3 | workers | 37 | 12 | 49 |
| 9 | 2 | 4 | workers | 37 | 12 | 49 |
| 9 | 3 | 1 | ot_1 | 0 | 2 | 2 |
| 9 | 3 | 1 | workers | 112 | 20 | 132 |
| 9 | 3 | 2 | ot_1 | 0 | 5 | 5 |
| 9 | 3 | 2 | ot_2 | 0 | 1 | 1 |
| 9 | 3 | 2 | workers | 458 | 28 | 486 |
| 9 | 3 | 3 | workers | 51 | 3 | 54 |
| 9 | 3 | 4 | workers | 51 | 3 | 54 |
| 9 | 4 | 1 | ot_1 | 0 | 11 | 11 |
| 9 | 4 | 1 | ot_2 | 0 | 6 | 6 |
| 9 | 4 | 1 | workers | 120 | 62 | 182 |
| 9 | 4 | 2 | ot_1 | 0 | 16 | 16 |
| 9 | 4 | 2 | ot_2 | 0 | 14 | 14 |
| 9 | 4 | 2 | workers | 488 | 90 | 578 |
| 9 | 4 | 3 | workers | 55 | 13 | 68 |
| 9 | 4 | 4 | workers | 55 | 13 | 68 |
| 9 | 5 | 1 | workers | 8 | 0 | 8 |
| 9 | 5 | 2 | workers | 32 | 0 | 32 |
| 9 | 5 | 3 | workers | 4 | 0 | 4 |
| 9 | 5 | 4 | workers | 4 | 0 | 4 |
| 9 | 6 | 1 | ot_1 | 0 | 13 | 13 |
| 9 | 6 | 1 | ot_2 | 0 | 22 | 22 |
| 9 | 6 | 1 | workers | 107 | 71 | 178 |
| 9 | 6 | 2 | workers | 433 | 166 | 599 |
| 9 | 6 | 3 | workers | 48 | 17 | 65 |
| 9 | 6 | 4 | workers | 48 | 17 | 65 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 9 | 7 | 1 | workers | 52 | 29 | 81 |
| 9 | 7 | 2 | workers | 209 | 44 | 253 |
| 9 | 7 | 3 | workers | 23 | 5 | 28 |
| 9 | 7 | 4 | workers | 23 | 5 | 28 |
| 9 | 8 | 1 | workers | 47 | 40 | 87 |
| 9 | 8 | 2 | workers | 190 | 60 | 250 |
| 9 | 8 | 3 | workers | 22 | 6 | 28 |
| 9 | 8 | 4 | workers | 22 | 6 | 28 |
| 9 | 9 | 1 | workers | 50 | 104 | 154 |
| 9 | 9 | 2 | workers | 202 | 157 | 359 |
| 9 | 9 | 3 | workers | 23 | 16 | 39 |
| 9 | 9 | 4 | workers | 23 | 16 | 39 |
| 9 | 10 | 1 | ot_1 | 0 | 3 | 3 |
| 9 | 10 | 1 | workers | 48 | 76 | 124 |
| 9 | 10 | 2 | ot_1 | 0 | 5 | 5 |
| 9 | 10 | 2 | workers | 192 | 114 | 306 |
| 9 | 10 | 3 | workers | 21 | 12 | 33 |
| 9 | 10 | 4 | workers | 21 | 12 | 33 |
| 9 | 11 | 1 | workers | 16 | 43 | 59 |
| 9 | 11 | 2 | workers | 63 | 65 | 128 |
| 9 | 11 | 3 | workers | 7 | 7 | 14 |
| 9 | 11 | 4 | workers | 7 | 7 | 14 |
| 9 | 12 | 1 | workers | 82 | 0 | 82 |
| 9 | 12 | 2 | workers | 334 | 0 | 334 |
| 9 | 12 | 3 | workers | 37 | 0 | 37 |
| 9 | 12 | 4 | workers | 37 | 0 | 37 |
| 9 | 13 | 1 | ot_1 | 0 | 1 | 1 |
| 9 | 13 | 1 | workers | 120 | 51 | 171 |
| 9 | 13 | 2 | ot_1 | 0 | 3 | 3 |
| 9 | 13 | 2 | workers | 486 | 76 | 562 |
| 9 | 13 | 3 | workers | 54 | 8 | 62 |
| 9 | 13 | 4 | workers | 54 | 8 | 62 |
| 9 | 14 | 1 | workers | 34 | 72 | 106 |
| 9 | 14 | 2 | workers | 138 | 108 | 246 |
| 9 | 14 | 3 | workers | 15 | 11 | 26 |
| 9 | 14 | 4 | workers | 15 | 11 | 26 |
| 9 | 15 | 1 | workers | 65 | 55 | 120 |
| 9 | 15 | 2 | workers | 262 | 83 | 345 |
| 9 | 15 | 3 | workers | 29 | 8 | 37 |
| 9 | 15 | 4 | workers | 29 | 8 | 37 |
| 9 | 16 | 1 | workers | 93 | 79 | 172 |
| 9 | 16 | 2 | workers | 377 | 119 | 496 |
| 9 | 16 | 3 | workers | 42 | 12 | 54 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 9 | 16 | 4 | workers | 42 | 12 | 54 |
| 9 | 17 | 1 | workers | 42 | 46 | 88 |
| 9 | 17 | 2 | ot_1 | 0 | 2 | 2 |
| 9 | 17 | 2 | workers | 174 | 68 | 242 |
| 9 | 17 | 3 | workers | 19 | 7 | 26 |
| 9 | 17 | 4 | workers | 19 | 7 | 26 |
| 9 | 18 | 1 | workers | 62 | 33 | 95 |
| 9 | 18 | 2 | workers | 250 | 50 | 300 |
| 9 | 18 | 3 | workers | 28 | 5 | 33 |
| 9 | 18 | 4 | workers | 28 | 5 | 33 |
| 9 | 19 | 1 | workers | 0 | 74 | 74 |
| 9 | 19 | 2 | workers | 0 | 111 | 111 |
| 9 | 19 | 3 | workers | 0 | 11 | 11 |
| 9 | 19 | 4 | workers | 0 | 11 | 11 |
| 9 | 20 | 1 | ot_1 | 0 | 13 | 13 |
| 9 | 20 | 1 | ot_2 | 0 | 8 | 8 |
| 9 | 20 | 1 | workers | 0 | 74 | 74 |
| 9 | 20 | 2 | ot_1 | 0 | 21 | 21 |
| 9 | 20 | 2 | ot_2 | 0 | 11 | 11 |
| 9 | 20 | 2 | workers | 0 | 112 | 112 |
| 9 | 20 | 3 | ot_1 | 0 | 2 | 2 |
| 9 | 20 | 3 | ot_2 | 0 | 1 | 1 |
| 9 | 20 | 3 | workers | 0 | 11 | 11 |
| 9 | 20 | 4 | ot_1 | 0 | 2 | 2 |
| 9 | 20 | 4 | ot_2 | 0 | 1 | 1 |
| 9 | 20 | 4 | workers | 0 | 11 | 11 |
| 9 | 21 | 1 | workers | 113 | 0 | 113 |
| 9 | 21 | 2 | workers | 459 | 0 | 459 |
| 9 | 21 | 3 | workers | 51 | 0 | 51 |
| 9 | 21 | 4 | workers | 51 | 0 | 51 |
| 9 | 22 | 1 | workers | 119 | 90 | 209 |
| 9 | 22 | 2 | workers | 484 | 135 | 619 |
| 9 | 22 | 3 | workers | 54 | 14 | 68 |
| 9 | 22 | 4 | workers | 54 | 14 | 68 |
| 9 | 23 | 1 | workers | 10 | 0 | 10 |
| 9 | 23 | 2 | workers | 42 | 0 | 42 |
| 9 | 23 | 3 | workers | 5 | 0 | 5 |
| 9 | 23 | 4 | workers | 5 | 0 | 5 |
| 9 | 24 | 1 | workers | 50 | 0 | 50 |
| 9 | 24 | 2 | workers | 203 | 0 | 203 |
| 9 | 24 | 3 | workers | 23 | 0 | 23 |
| 9 | 24 | 4 | workers | 23 | 0 | 23 |
| 9 | 25 | 1 | workers | 127 | 0 | 127 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 9 | 25 | 2 | workers | 514 | 0 | 514 |
| 9 | 25 | 3 | workers | 57 | 0 | 57 |
| 9 | 25 | 4 | workers | 57 | 0 | 57 |
| 10 | 1 | 1 | workers | 185 | 95 | 280 |
| 10 | 1 | 2 | workers | 751 | 144 | 895 |
| 10 | 1 | 3 | workers | 84 | 15 | 99 |
| 10 | 1 | 4 | workers | 84 | 15 | 99 |
| 10 | 2 | 1 | workers | 82 | 79 | 161 |
| 10 | 2 | 2 | workers | 333 | 119 | 452 |
| 10 | 2 | 3 | workers | 37 | 12 | 49 |
| 10 | 2 | 4 | workers | 37 | 12 | 49 |
| 10 | 3 | 1 | workers | 112 | 20 | 132 |
| 10 | 3 | 2 | workers | 458 | 28 | 486 |
| 10 | 3 | 3 | workers | 51 | 3 | 54 |
| 10 | 3 | 4 | workers | 51 | 3 | 54 |
| 10 | 4 | 1 | workers | 120 | 62 | 182 |
| 10 | 4 | 2 | workers | 488 | 90 | 578 |
| 10 | 4 | 3 | workers | 55 | 13 | 68 |
| 10 | 4 | 4 | workers | 55 | 13 | 68 |
| 10 | 5 | 1 | workers | 8 | 0 | 8 |
| 10 | 5 | 2 | workers | 32 | 0 | 32 |
| 10 | 5 | 3 | workers | 4 | 0 | 4 |
| 10 | 5 | 4 | workers | 4 | 0 | 4 |
| 10 | 6 | 1 | ot_1 | 0 | 2 | 2 |
| 10 | 6 | 1 | workers | 107 | 71 | 178 |
| 10 | 6 | 2 | workers | 433 | 166 | 599 |
| 10 | 6 | 3 | workers | 48 | 17 | 65 |
| 10 | 6 | 4 | workers | 48 | 17 | 65 |
| 10 | 7 | 1 | workers | 52 | 29 | 81 |
| 10 | 7 | 2 | workers | 209 | 44 | 253 |
| 10 | 7 | 3 | workers | 23 | 5 | 28 |
| 10 | 7 | 4 | workers | 23 | 5 | 28 |
| 10 | 8 | 1 | workers | 47 | 40 | 87 |
| 10 | 8 | 2 | workers | 190 | 60 | 250 |
| 10 | 8 | 3 | workers | 22 | 6 | 28 |
| 10 | 8 | 4 | workers | 22 | 6 | 28 |
| 10 | 9 | 1 | workers | 50 | 104 | 154 |
| 10 | 9 | 2 | workers | 202 | 157 | 359 |
| 10 | 9 | 3 | workers | 23 | 16 | 39 |
| 10 | 9 | 4 | workers | 23 | 16 | 39 |
| 10 | 10 | 1 | workers | 48 | 76 | 124 |
| 10 | 10 | 2 | workers | 192 | 114 | 306 |
| 10 | 10 | 3 | workers | 21 | 12 | 33 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 10 | 10 | 4 | workers | 21 | 12 | 33 |
| 10 | 11 | 1 | workers | 16 | 43 | 59 |
| 10 | 11 | 2 | workers | 63 | 65 | 128 |
| 10 | 11 | 3 | workers | 7 | 7 | 14 |
| 10 | 11 | 4 | workers | 7 | 7 | 14 |
| 10 | 12 | 1 | workers | 82 | 0 | 82 |
| 10 | 12 | 2 | workers | 334 | 0 | 334 |
| 10 | 12 | 3 | workers | 37 | 0 | 37 |
| 10 | 12 | 4 | workers | 37 | 0 | 37 |
| 10 | 13 | 1 | workers | 120 | 51 | 171 |
| 10 | 13 | 2 | workers | 486 | 76 | 562 |
| 10 | 13 | 3 | workers | 54 | 8 | 62 |
| 10 | 13 | 4 | workers | 54 | 8 | 62 |
| 10 | 14 | 1 | workers | 34 | 72 | 106 |
| 10 | 14 | 2 | workers | 138 | 108 | 246 |
| 10 | 14 | 3 | workers | 15 | 11 | 26 |
| 10 | 14 | 4 | workers | 15 | 11 | 26 |
| 10 | 15 | 1 | workers | 65 | 55 | 120 |
| 10 | 15 | 2 | workers | 262 | 83 | 345 |
| 10 | 15 | 3 | workers | 29 | 8 | 37 |
| 10 | 15 | 4 | workers | 29 | 8 | 37 |
| 10 | 16 | 1 | workers | 93 | 79 | 172 |
| 10 | 16 | 2 | workers | 377 | 119 | 496 |
| 10 | 16 | 3 | workers | 42 | 12 | 54 |
| 10 | 16 | 4 | workers | 42 | 12 | 54 |
| 10 | 17 | 1 | workers | 42 | 46 | 88 |
| 10 | 17 | 2 | workers | 174 | 68 | 242 |
| 10 | 17 | 3 | workers | 19 | 7 | 26 |
| 10 | 17 | 4 | workers | 19 | 7 | 26 |
| 10 | 18 | 1 | workers | 62 | 33 | 95 |
| 10 | 18 | 2 | workers | 250 | 50 | 300 |
| 10 | 18 | 3 | workers | 28 | 5 | 33 |
| 10 | 18 | 4 | workers | 28 | 5 | 33 |
| 10 | 19 | 1 | workers | 0 | 74 | 74 |
| 10 | 19 | 2 | workers | 0 | 111 | 111 |
| 10 | 19 | 3 | workers | 0 | 11 | 11 |
| 10 | 19 | 4 | workers | 0 | 11 | 11 |
| 10 | 20 | 1 | workers | 0 | 74 | 74 |
| 10 | 20 | 2 | workers | 0 | 112 | 112 |
| 10 | 20 | 3 | workers | 0 | 11 | 11 |
| 10 | 20 | 4 | workers | 0 | 11 | 11 |
| 10 | 21 | 1 | workers | 113 | 0 | 113 |
| 10 | 21 | 2 | workers | 459 | 0 | 459 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 10 | 21 | 3 | workers | 51 | 0 | 51 |
| 10 | 21 | 4 | workers | 51 | 0 | 51 |
| 10 | 22 | 1 | workers | 119 | 90 | 209 |
| 10 | 22 | 2 | workers | 484 | 135 | 619 |
| 10 | 22 | 3 | workers | 54 | 14 | 68 |
| 10 | 22 | 4 | workers | 54 | 14 | 68 |
| 10 | 23 | 1 | workers | 10 | 0 | 10 |
| 10 | 23 | 2 | workers | 42 | 0 | 42 |
| 10 | 23 | 3 | workers | 5 | 0 | 5 |
| 10 | 23 | 4 | workers | 5 | 0 | 5 |
| 10 | 24 | 1 | workers | 50 | 0 | 50 |
| 10 | 24 | 2 | workers | 203 | 0 | 203 |
| 10 | 24 | 3 | workers | 23 | 0 | 23 |
| 10 | 24 | 4 | workers | 23 | 0 | 23 |
| 10 | 25 | 1 | workers | 127 | 0 | 127 |
| 10 | 25 | 2 | workers | 514 | 0 | 514 |
| 10 | 25 | 3 | workers | 57 | 0 | 57 |
| 10 | 25 | 4 | workers | 57 | 0 | 57 |
| 11 | 1 | 1 | ot_1 | 0 | 17 | 17 |
| 11 | 1 | 1 | workers | 185 | 95 | 280 |
| 11 | 1 | 2 | hire | 1 | 0 | 1 |
| 11 | 1 | 2 | ot_1 | 0 | 27 | 27 |
| 11 | 1 | 2 | workers | 751 | 144 | 895 |
| 11 | 1 | 3 | hire | 1 | 0 | 1 |
| 11 | 1 | 3 | ot_1 | 0 | 2 | 2 |
| 11 | 1 | 3 | workers | 84 | 15 | 99 |
| 11 | 1 | 4 | hire | 1 | 0 | 1 |
| 11 | 1 | 4 | ot_1 | 0 | 2 | 2 |
| 11 | 1 | 4 | workers | 84 | 15 | 99 |
| 11 | 2 | 1 | ot_1 | 0 | 10 | 10 |
| 11 | 2 | 1 | workers | 82 | 79 | 161 |
| 11 | 2 | 2 | hire | 1 | 0 | 1 |
| 11 | 2 | 2 | ot_1 | 0 | 16 | 16 |
| 11 | 2 | 2 | workers | 333 | 119 | 452 |
| 11 | 2 | 3 | hire | 1 | 0 | 1 |
| 11 | 2 | 3 | ot_1 | 0 | 1 | 1 |
| 11 | 2 | 3 | workers | 37 | 12 | 49 |
| 11 | 2 | 4 | hire | 1 | 0 | 1 |
| 11 | 2 | 4 | ot_1 | 0 | 1 | 1 |
| 11 | 2 | 4 | workers | 37 | 12 | 49 |
| 11 | 3 | 1 | hire | 1 | 0 | 1 |
| 11 | 3 | 1 | ot_1 | 0 | 3 | 3 |
| 11 | 3 | 1 | workers | 112 | 20 | 132 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 11 | 3 | 2 | ot_1 | 0 | 5 | 5 |
| 11 | 3 | 2 | ot_2 | 0 | 4 | 4 |
| 11 | 3 | 2 | workers | 458 | 28 | 486 |
| 11 | 3 | 3 | hire | 1 | 0 | 1 |
| 11 | 3 | 3 | workers | 51 | 3 | 54 |
| 11 | 3 | 4 | hire | 1 | 0 | 1 |
| 11 | 3 | 4 | workers | 51 | 3 | 54 |
| 11 | 4 | 1 | ot_1 | 0 | 11 | 11 |
| 11 | 4 | 1 | ot_2 | 0 | 17 | 17 |
| 11 | 4 | 1 | workers | 120 | 62 | 182 |
| 11 | 4 | 2 | hire | 1 | 0 | 1 |
| 11 | 4 | 2 | ot_1 | 0 | 16 | 16 |
| 11 | 4 | 2 | ot_2 | 0 | 30 | 30 |
| 11 | 4 | 2 | workers | 488 | 90 | 578 |
| 11 | 4 | 3 | ot_1 | 0 | 1 | 1 |
| 11 | 4 | 3 | workers | 55 | 13 | 68 |
| 11 | 4 | 4 | ot_1 | 0 | 1 | 1 |
| 11 | 4 | 4 | workers | 55 | 13 | 68 |
| 11 | 5 | 1 | hire | 1 | 0 | 1 |
| 11 | 5 | 1 | workers | 8 | 0 | 8 |
| 11 | 5 | 2 | hire | 1 | 0 | 1 |
| 11 | 5 | 2 | workers | 32 | 0 | 32 |
| 11 | 5 | 3 | workers | 4 | 0 | 4 |
| 11 | 5 | 4 | workers | 4 | 0 | 4 |
| 11 | 6 | 1 | ot_1 | 0 | 13 | 13 |
| 11 | 6 | 1 | ot_2 | 0 | 25 | 25 |
| 11 | 6 | 1 | workers | 107 | 71 | 178 |
| 11 | 6 | 2 | workers | 433 | 166 | 599 |
| 11 | 6 | 3 | hire | 1 | 0 | 1 |
| 11 | 6 | 3 | workers | 48 | 17 | 65 |
| 11 | 6 | 4 | hire | 1 | 0 | 1 |
| 11 | 6 | 4 | workers | 48 | 17 | 65 |
| 11 | 7 | 1 | ot_1 | 0 | 2 | 2 |
| 11 | 7 | 1 | workers | 52 | 29 | 81 |
| 11 | 7 | 2 | ot_1 | 0 | 3 | 3 |
| 11 | 7 | 2 | workers | 209 | 44 | 253 |
| 11 | 7 | 3 | hire | 1 | 0 | 1 |
| 11 | 7 | 3 | workers | 23 | 5 | 28 |
| 11 | 7 | 4 | hire | 1 | 0 | 1 |
| 11 | 7 | 4 | workers | 23 | 5 | 28 |
| 11 | 8 | 1 | ot_1 | 0 | 2 | 2 |
| 11 | 8 | 1 | workers | 47 | 40 | 87 |
| 11 | 8 | 2 | hire | 1 | 0 | 1 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 11 | 8 | 2 | ot_1 | 0 | 4 | 4 |
| 11 | 8 | 2 | workers | 190 | 60 | 250 |
| 11 | 8 | 3 | workers | 22 | 6 | 28 |
| 11 | 8 | 4 | workers | 22 | 6 | 28 |
| 11 | 9 | 1 | ot_1 | 0 | 4 | 4 |
| 11 | 9 | 1 | workers | 50 | 104 | 154 |
| 11 | 9 | 2 | hire | 1 | 0 | 1 |
| 11 | 9 | 2 | ot_1 | 0 | 7 | 7 |
| 11 | 9 | 2 | workers | 202 | 157 | 359 |
| 11 | 9 | 3 | workers | 23 | 16 | 39 |
| 11 | 9 | 4 | workers | 23 | 16 | 39 |
| 11 | 10 | 1 | ot_1 | 0 | 3 | 3 |
| 11 | 10 | 1 | workers | 48 | 76 | 124 |
| 11 | 10 | 2 | ot_1 | 0 | 5 | 5 |
| 11 | 10 | 2 | workers | 192 | 114 | 306 |
| 11 | 10 | 3 | hire | 1 | 0 | 1 |
| 11 | 10 | 3 | workers | 21 | 12 | 33 |
| 11 | 10 | 4 | hire | 1 | 0 | 1 |
| 11 | 10 | 4 | workers | 21 | 12 | 33 |
| 11 | 11 | 1 | ot_1 | 0 | 2 | 2 |
| 11 | 11 | 1 | workers | 16 | 43 | 59 |
| 11 | 11 | 2 | ot_1 | 0 | 2 | 2 |
| 11 | 11 | 2 | workers | 63 | 65 | 128 |
| 11 | 11 | 3 | workers | 7 | 7 | 14 |
| 11 | 11 | 4 | workers | 7 | 7 | 14 |
| 11 | 12 | 1 | workers | 82 | 0 | 82 |
| 11 | 12 | 2 | workers | 334 | 0 | 334 |
| 11 | 12 | 3 | hire | 1 | 0 | 1 |
| 11 | 12 | 3 | workers | 37 | 0 | 37 |
| 11 | 12 | 4 | hire | 2 | 0 | 2 |
| 11 | 12 | 4 | workers | 37 | 0 | 37 |
| 11 | 13 | 1 | workers | 120 | 51 | 171 |
| 11 | 13 | 2 | hire | 1 | 0 | 1 |
| 11 | 13 | 2 | workers | 486 | 76 | 562 |
| 11 | 13 | 3 | hire | 1 | 0 | 1 |
| 11 | 13 | 3 | workers | 54 | 8 | 62 |
| 11 | 13 | 4 | hire | 1 | 0 | 1 |
| 11 | 13 | 4 | workers | 54 | 8 | 62 |
| 11 | 14 | 1 | ot_1 | 0 | 5 | 5 |
| 11 | 14 | 1 | workers | 34 | 72 | 106 |
| 11 | 14 | 2 | ot_1 | 0 | 8 | 8 |
| 11 | 14 | 2 | workers | 138 | 108 | 246 |
| 11 | 14 | 3 | hire | 1 | 0 | 1 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 11 | 14 | 3 | ot_1 | 0 | 1 | 1 |
| 11 | 14 | 3 | workers | 15 | 11 | 26 |
| 11 | 14 | 4 | hire | 1 | 0 | 1 |
| 11 | 14 | 4 | ot_1 | 0 | 1 | 1 |
| 11 | 14 | 4 | workers | 15 | 11 | 26 |
| 11 | 15 | 1 | ot_1 | 0 | 4 | 4 |
| 11 | 15 | 1 | workers | 65 | 55 | 120 |
| 11 | 15 | 2 | hire | 1 | 0 | 1 |
| 11 | 15 | 2 | ot_1 | 0 | 6 | 6 |
| 11 | 15 | 2 | workers | 262 | 83 | 345 |
| 11 | 15 | 3 | hire | 1 | 0 | 1 |
| 11 | 15 | 3 | ot_1 | 0 | 1 | 1 |
| 11 | 15 | 3 | workers | 29 | 8 | 37 |
| 11 | 15 | 4 | hire | 1 | 0 | 1 |
| 11 | 15 | 4 | ot_1 | 0 | 1 | 1 |
| 11 | 15 | 4 | workers | 29 | 8 | 37 |
| 11 | 16 | 1 | ot_1 | 0 | 10 | 10 |
| 11 | 16 | 1 | workers | 93 | 79 | 172 |
| 11 | 16 | 2 | hire | 1 | 0 | 1 |
| 11 | 16 | 2 | ot_1 | 0 | 16 | 16 |
| 11 | 16 | 2 | workers | 377 | 119 | 496 |
| 11 | 16 | 3 | hire | 1 | 0 | 1 |
| 11 | 16 | 3 | ot_1 | 0 | 1 | 1 |
| 11 | 16 | 3 | workers | 42 | 12 | 54 |
| 11 | 16 | 4 | hire | 1 | 0 | 1 |
| 11 | 16 | 4 | ot_1 | 0 | 1 | 1 |
| 11 | 16 | 4 | workers | 42 | 12 | 54 |
| 11 | 17 | 1 | hire | 1 | 0 | 1 |
| 11 | 17 | 1 | ot_1 | 0 | 2 | 2 |
| 11 | 17 | 1 | workers | 42 | 46 | 88 |
| 11 | 17 | 2 | hire | 1 | 0 | 1 |
| 11 | 17 | 2 | ot_1 | 0 | 5 | 5 |
| 11 | 17 | 2 | workers | 174 | 68 | 242 |
| 11 | 17 | 3 | hire | 1 | 0 | 1 |
| 11 | 17 | 3 | workers | 19 | 7 | 26 |
| 11 | 17 | 4 | hire | 1 | 0 | 1 |
| 11 | 17 | 4 | workers | 19 | 7 | 26 |
| 11 | 18 | 1 | workers | 62 | 33 | 95 |
| 11 | 18 | 2 | workers | 250 | 50 | 300 |
| 11 | 18 | 3 | workers | 28 | 5 | 33 |
| 11 | 18 | 4 | workers | 28 | 5 | 33 |
| 11 | 19 | 1 | ot_1 | 0 | 4 | 4 |
| 11 | 19 | 1 | workers | 0 | 74 | 74 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 11 | 19 | 2 | ot_1 | 0 | 7 | 7 |
| 11 | 19 | 2 | workers | 0 | 111 | 111 |
| 11 | 19 | 3 | ot_1 | 0 | 1 | 1 |
| 11 | 19 | 3 | workers | 0 | 11 | 11 |
| 11 | 19 | 4 | ot_1 | 0 | 1 | 1 |
| 11 | 19 | 4 | workers | 0 | 11 | 11 |
| 11 | 20 | 1 | ot_1 | 0 | 13 | 13 |
| 11 | 20 | 1 | ot_2 | 0 | 15 | 15 |
| 11 | 20 | 1 | workers | 0 | 74 | 74 |
| 11 | 20 | 2 | ot_1 | 0 | 21 | 21 |
| 11 | 20 | 2 | ot_2 | 0 | 23 | 23 |
| 11 | 20 | 2 | workers | 0 | 112 | 112 |
| 11 | 20 | 3 | ot_1 | 0 | 2 | 2 |
| 11 | 20 | 3 | ot_2 | 0 | 3 | 3 |
| 11 | 20 | 3 | workers | 0 | 11 | 11 |
| 11 | 20 | 4 | ot_1 | 0 | 2 | 2 |
| 11 | 20 | 4 | ot_2 | 0 | 3 | 3 |
| 11 | 20 | 4 | workers | 0 | 11 | 11 |
| 11 | 21 | 1 | workers | 113 | 0 | 113 |
| 11 | 21 | 2 | workers | 459 | 0 | 459 |
| 11 | 21 | 3 | hire | 1 | 0 | 1 |
| 11 | 21 | 3 | workers | 51 | 0 | 51 |
| 11 | 21 | 4 | hire | 1 | 0 | 1 |
| 11 | 21 | 4 | workers | 51 | 0 | 51 |
| 11 | 22 | 1 | ot_1 | 0 | 1 | 1 |
| 11 | 22 | 1 | workers | 119 | 90 | 209 |
| 11 | 22 | 2 | hire | 1 | 0 | 1 |
| 11 | 22 | 2 | ot_1 | 0 | 3 | 3 |
| 11 | 22 | 2 | workers | 484 | 135 | 619 |
| 11 | 22 | 3 | hire | 1 | 0 | 1 |
| 11 | 22 | 3 | workers | 54 | 14 | 68 |
| 11 | 22 | 4 | hire | 1 | 0 | 1 |
| 11 | 22 | 4 | workers | 54 | 14 | 68 |
| 11 | 23 | 1 | hire | 1 | 0 | 1 |
| 11 | 23 | 1 | workers | 10 | 0 | 10 |
| 11 | 23 | 2 | workers | 42 | 0 | 42 |
| 11 | 23 | 3 | workers | 5 | 0 | 5 |
| 11 | 23 | 4 | workers | 5 | 0 | 5 |
| 11 | 24 | 1 | workers | 50 | 0 | 50 |
| 11 | 24 | 2 | hire | 1 | 0 | 1 |
| 11 | 24 | 2 | workers | 203 | 0 | 203 |
| 11 | 24 | 3 | workers | 23 | 0 | 23 |
| 11 | 24 | 4 | workers | 23 | 0 | 23 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 11 | 25 | 1 | workers | 127 | 0 | 127 |
| 11 | 25 | 2 | hire | 1 | 0 | 1 |
| 11 | 25 | 2 | workers | 514 | 0 | 514 |
| 11 | 25 | 3 | hire | 1 | 0 | 1 |
| 11 | 25 | 3 | workers | 57 | 0 | 57 |
| 11 | 25 | 4 | hire | 4 | 0 | 4 |
| 11 | 25 | 4 | workers | 57 | 0 | 57 |
| 12 | 1 | 1 | ot_1 | 34 | 17 | 51 |
| 12 | 1 | 1 | ot_2 | 153 | 0 | 153 |
| 12 | 1 | 1 | workers | 185 | 95 | 280 |
| 12 | 1 | 2 | ot_1 | 141 | 27 | 168 |
| 12 | 1 | 2 | ot_2 | 626 | 0 | 626 |
| 12 | 1 | 2 | workers | 752 | 144 | 896 |
| 12 | 1 | 3 | ot_1 | 15 | 2 | 17 |
| 12 | 1 | 3 | ot_2 | 68 | 0 | 68 |
| 12 | 1 | 3 | workers | 85 | 15 | 100 |
| 12 | 1 | 4 | ot_1 | 15 | 2 | 17 |
| 12 | 1 | 4 | ot_2 | 68 | 0 | 68 |
| 12 | 1 | 4 | workers | 85 | 15 | 100 |
| 12 | 2 | 1 | ot_1 | 15 | 10 | 25 |
| 12 | 2 | 1 | ot_2 | 68 | 0 | 68 |
| 12 | 2 | 1 | workers | 82 | 79 | 161 |
| 12 | 2 | 2 | ot_1 | 62 | 16 | 78 |
| 12 | 2 | 2 | ot_2 | 277 | 0 | 277 |
| 12 | 2 | 2 | workers | 334 | 119 | 453 |
| 12 | 2 | 3 | ot_1 | 7 | 1 | 8 |
| 12 | 2 | 3 | ot_2 | 30 | 0 | 30 |
| 12 | 2 | 3 | workers | 38 | 12 | 50 |
| 12 | 2 | 4 | ot_1 | 7 | 1 | 8 |
| 12 | 2 | 4 | ot_2 | 30 | 0 | 30 |
| 12 | 2 | 4 | workers | 38 | 12 | 50 |
| 12 | 3 | 1 | ot_1 | 21 | 3 | 24 |
| 12 | 3 | 1 | ot_2 | 93 | 0 | 93 |
| 12 | 3 | 1 | workers | 113 | 20 | 133 |
| 12 | 3 | 2 | ot_1 | 85 | 5 | 90 |
| 12 | 3 | 2 | ot_2 | 381 | 4 | 385 |
| 12 | 3 | 2 | workers | 458 | 28 | 486 |
| 12 | 3 | 3 | ot_1 | 9 | 0 | 9 |
| 12 | 3 | 3 | ot_2 | 41 | 0 | 41 |
| 12 | 3 | 3 | workers | 52 | 3 | 55 |
| 12 | 3 | 4 | ot_1 | 9 | 0 | 9 |
| 12 | 3 | 4 | ot_2 | 41 | 0 | 41 |
| 12 | 3 | 4 | workers | 52 | 3 | 55 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 12 | 4 | 1 | ot_1 | 22 | 11 | 33 |
| 12 | 4 | 1 | ot_2 | 99 | 17 | 116 |
| 12 | 4 | 1 | workers | 120 | 62 | 182 |
| 12 | 4 | 2 | ot_1 | 91 | 16 | 107 |
| 12 | 4 | 2 | ot_2 | 406 | 30 | 436 |
| 12 | 4 | 2 | workers | 489 | 90 | 579 |
| 12 | 4 | 3 | ot_1 | 9 | 1 | 10 |
| 12 | 4 | 3 | ot_2 | 45 | 0 | 45 |
| 12 | 4 | 3 | workers | 55 | 13 | 68 |
| 12 | 4 | 4 | ot_1 | 9 | 1 | 10 |
| 12 | 4 | 4 | ot_2 | 45 | 0 | 45 |
| 12 | 4 | 4 | workers | 55 | 13 | 68 |
| 12 | 5 | 1 | ot_1 | 1 | 0 | 1 |
| 12 | 5 | 1 | ot_2 | 5 | 0 | 5 |
| 12 | 5 | 1 | workers | 9 | 0 | 9 |
| 12 | 5 | 2 | ot_1 | 6 | 0 | 6 |
| 12 | 5 | 2 | ot_2 | 27 | 0 | 27 |
| 12 | 5 | 2 | workers | 33 | 0 | 33 |
| 12 | 5 | 3 | ot_2 | 3 | 0 | 3 |
| 12 | 5 | 3 | workers | 4 | 0 | 4 |
| 12 | 5 | 4 | ot_2 | 3 | 0 | 3 |
| 12 | 5 | 4 | workers | 4 | 0 | 4 |
| 12 | 6 | 1 | ot_1 | 20 | 13 | 33 |
| 12 | 6 | 1 | ot_2 | 87 | 25 | 112 |
| 12 | 6 | 1 | workers | 107 | 71 | 178 |
| 12 | 6 | 2 | ot_1 | 81 | 0 | 81 |
| 12 | 6 | 2 | ot_2 | 360 | 0 | 360 |
| 12 | 6 | 2 | workers | 433 | 166 | 599 |
| 12 | 6 | 3 | ot_1 | 9 | 0 | 9 |
| 12 | 6 | 3 | ot_2 | 39 | 0 | 39 |
| 12 | 6 | 3 | workers | 49 | 17 | 66 |
| 12 | 6 | 4 | ot_1 | 9 | 0 | 9 |
| 12 | 6 | 4 | ot_2 | 39 | 0 | 39 |
| 12 | 6 | 4 | workers | 49 | 17 | 66 |
| 12 | 7 | 1 | ot_1 | 9 | 5 | 14 |
| 12 | 7 | 1 | ot_2 | 41 | 23 | 64 |
| 12 | 7 | 1 | workers | 52 | 29 | 81 |
| 12 | 7 | 2 | ot_1 | 39 | 8 | 47 |
| 12 | 7 | 2 | ot_2 | 173 | 35 | 208 |
| 12 | 7 | 2 | workers | 209 | 44 | 253 |
| 12 | 7 | 3 | ot_1 | 4 | 0 | 4 |
| 12 | 7 | 3 | ot_2 | 18 | 3 | 21 |
| 12 | 7 | 3 | workers | 24 | 5 | 29 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 12 | 7 | 4 | ot_1 | 4 | 0 | 4 |
| 12 | 7 | 4 | ot_2 | 18 | 3 | 21 |
| 12 | 7 | 4 | workers | 24 | 5 | 29 |
| 12 | 8 | 1 | ot_1 | 8 | 7 | 15 |
| 12 | 8 | 1 | ot_2 | 38 | 27 | 65 |
| 12 | 8 | 1 | workers | 47 | 40 | 87 |
| 12 | 8 | 2 | ot_1 | 35 | 11 | 46 |
| 12 | 8 | 2 | ot_2 | 158 | 42 | 200 |
| 12 | 8 | 2 | workers | 191 | 60 | 251 |
| 12 | 8 | 3 | ot_1 | 4 | 1 | 5 |
| 12 | 8 | 3 | ot_2 | 16 | 4 | 20 |
| 12 | 8 | 3 | workers | 22 | 6 | 28 |
| 12 | 8 | 4 | ot_1 | 4 | 1 | 5 |
| 12 | 8 | 4 | ot_2 | 16 | 4 | 20 |
| 12 | 8 | 4 | workers | 22 | 6 | 28 |
| 12 | 9 | 1 | ot_1 | 9 | 19 | 28 |
| 12 | 9 | 1 | ot_2 | 41 | 74 | 115 |
| 12 | 9 | 1 | workers | 50 | 104 | 154 |
| 12 | 9 | 2 | ot_1 | 38 | 29 | 67 |
| 12 | 9 | 2 | ot_2 | 168 | 112 | 280 |
| 12 | 9 | 2 | workers | 203 | 157 | 360 |
| 12 | 9 | 3 | ot_1 | 4 | 3 | 7 |
| 12 | 9 | 3 | ot_2 | 18 | 11 | 29 |
| 12 | 9 | 3 | workers | 23 | 16 | 39 |
| 12 | 9 | 4 | ot_1 | 4 | 3 | 7 |
| 12 | 9 | 4 | ot_2 | 18 | 11 | 29 |
| 12 | 9 | 4 | workers | 23 | 16 | 39 |
| 12 | 10 | 1 | ot_1 | 9 | 14 | 23 |
| 12 | 10 | 1 | ot_2 | 38 | 61 | 99 |
| 12 | 10 | 1 | workers | 48 | 76 | 124 |
| 12 | 10 | 2 | ot_1 | 36 | 21 | 57 |
| 12 | 10 | 2 | ot_2 | 159 | 94 | 253 |
| 12 | 10 | 2 | workers | 192 | 114 | 306 |
| 12 | 10 | 3 | ot_1 | 4 | 1 | 5 |
| 12 | 10 | 3 | ot_2 | 17 | 9 | 26 |
| 12 | 10 | 3 | workers | 22 | 12 | 34 |
| 12 | 10 | 4 | ot_1 | 4 | 1 | 5 |
| 12 | 10 | 4 | ot_2 | 17 | 9 | 26 |
| 12 | 10 | 4 | workers | 22 | 12 | 34 |
| 12 | 11 | 1 | ot_1 | 3 | 8 | 11 |
| 12 | 11 | 1 | ot_2 | 11 | 29 | 40 |
| 12 | 11 | 1 | workers | 16 | 43 | 59 |
| 12 | 11 | 2 | ot_1 | 11 | 12 | 23 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 12 | 11 | 2 | ot_2 | 50 | 44 | 94 |
| 12 | 11 | 2 | workers | 63 | 65 | 128 |
| 12 | 11 | 3 | ot_1 | 1 | 0 | 1 |
| 12 | 11 | 3 | ot_2 | 5 | 4 | 9 |
| 12 | 11 | 3 | workers | 7 | 7 | 14 |
| 12 | 11 | 4 | ot_1 | 1 | 0 | 1 |
| 12 | 11 | 4 | ot_2 | 5 | 4 | 9 |
| 12 | 11 | 4 | workers | 7 | 7 | 14 |
| 12 | 12 | 1 | ot_1 | 15 | 0 | 15 |
| 12 | 12 | 1 | ot_2 | 68 | 0 | 68 |
| 12 | 12 | 1 | workers | 82 | 0 | 82 |
| 12 | 12 | 2 | ot_1 | 62 | 0 | 62 |
| 12 | 12 | 2 | ot_2 | 277 | 0 | 277 |
| 12 | 12 | 2 | workers | 334 | 0 | 334 |
| 12 | 12 | 3 | ot_1 | 7 | 0 | 7 |
| 12 | 12 | 3 | ot_2 | 30 | 0 | 30 |
| 12 | 12 | 3 | workers | 38 | 0 | 38 |
| 12 | 12 | 4 | ot_1 | 7 | 0 | 7 |
| 12 | 12 | 4 | ot_2 | 29 | 0 | 29 |
| 12 | 12 | 4 | workers | 39 | 0 | 39 |
| 12 | 13 | 1 | ot_1 | 22 | 0 | 22 |
| 12 | 13 | 1 | ot_2 | 98 | 0 | 98 |
| 12 | 13 | 1 | workers | 120 | 51 | 171 |
| 12 | 13 | 2 | ot_1 | 91 | 0 | 91 |
| 12 | 13 | 2 | ot_2 | 405 | 0 | 405 |
| 12 | 13 | 2 | workers | 487 | 76 | 563 |
| 12 | 13 | 3 | ot_1 | 10 | 0 | 10 |
| 12 | 13 | 3 | ot_2 | 44 | 0 | 44 |
| 12 | 13 | 3 | workers | 55 | 8 | 63 |
| 12 | 13 | 4 | ot_1 | 10 | 0 | 10 |
| 12 | 13 | 4 | ot_2 | 44 | 0 | 44 |
| 12 | 13 | 4 | workers | 55 | 8 | 63 |
| 12 | 14 | 1 | ot_1 | 6 | 13 | 19 |
| 12 | 14 | 1 | ot_2 | 27 | 52 | 79 |
| 12 | 14 | 1 | workers | 34 | 72 | 106 |
| 12 | 14 | 2 | ot_1 | 25 | 20 | 45 |
| 12 | 14 | 2 | ot_2 | 113 | 80 | 193 |
| 12 | 14 | 2 | workers | 138 | 108 | 246 |
| 12 | 14 | 3 | ot_1 | 3 | 2 | 5 |
| 12 | 14 | 3 | ot_2 | 11 | 8 | 19 |
| 12 | 14 | 3 | workers | 16 | 11 | 27 |
| 12 | 14 | 4 | ot_1 | 3 | 2 | 5 |
| 12 | 14 | 4 | ot_2 | 11 | 8 | 19 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 12 | 14 | 4 | workers | 16 | 11 | 27 |
| 12 | 15 | 1 | ot_1 | 12 | 10 | 22 |
| 12 | 15 | 1 | ot_2 | 53 | 30 | 83 |
| 12 | 15 | 1 | workers | 65 | 55 | 120 |
| 12 | 15 | 2 | ot_1 | 49 | 15 | 64 |
| 12 | 15 | 2 | ot_2 | 218 | 46 | 264 |
| 12 | 15 | 2 | workers | 263 | 83 | 346 |
| 12 | 15 | 3 | ot_1 | 5 | 1 | 6 |
| 12 | 15 | 3 | ot_2 | 23 | 5 | 28 |
| 12 | 15 | 3 | workers | 30 | 8 | 38 |
| 12 | 15 | 4 | ot_1 | 5 | 1 | 6 |
| 12 | 15 | 4 | ot_2 | 23 | 5 | 28 |
| 12 | 15 | 4 | workers | 30 | 8 | 38 |
| 12 | 16 | 1 | ot_1 | 17 | 10 | 27 |
| 12 | 16 | 1 | ot_2 | 76 | 0 | 76 |
| 12 | 16 | 1 | workers | 93 | 79 | 172 |
| 12 | 16 | 2 | ot_1 | 70 | 16 | 86 |
| 12 | 16 | 2 | ot_2 | 314 | 0 | 314 |
| 12 | 16 | 2 | workers | 378 | 119 | 497 |
| 12 | 16 | 3 | ot_1 | 8 | 1 | 9 |
| 12 | 16 | 3 | ot_2 | 34 | 0 | 34 |
| 12 | 16 | 3 | workers | 43 | 12 | 55 |
| 12 | 16 | 4 | ot_1 | 8 | 1 | 9 |
| 12 | 16 | 4 | ot_2 | 34 | 0 | 34 |
| 12 | 16 | 4 | workers | 43 | 12 | 55 |
| 12 | 17 | 1 | ot_1 | 8 | 2 | 10 |
| 12 | 17 | 1 | ot_2 | 35 | 0 | 35 |
| 12 | 17 | 1 | workers | 43 | 46 | 89 |
| 12 | 17 | 2 | ot_1 | 32 | 5 | 37 |
| 12 | 17 | 2 | ot_2 | 144 | 0 | 144 |
| 12 | 17 | 2 | workers | 175 | 68 | 243 |
| 12 | 17 | 3 | ot_1 | 3 | 0 | 3 |
| 12 | 17 | 3 | ot_2 | 15 | 0 | 15 |
| 12 | 17 | 3 | workers | 20 | 7 | 27 |
| 12 | 17 | 4 | ot_1 | 3 | 0 | 3 |
| 12 | 17 | 4 | ot_2 | 15 | 0 | 15 |
| 12 | 17 | 4 | workers | 20 | 7 | 27 |
| 12 | 18 | 1 | ot_1 | 11 | 0 | 11 |
| 12 | 18 | 1 | ot_2 | 50 | 0 | 50 |
| 12 | 18 | 1 | workers | 62 | 33 | 95 |
| 12 | 18 | 2 | ot_1 | 46 | 0 | 46 |
| 12 | 18 | 2 | ot_2 | 208 | 0 | 208 |
| 12 | 18 | 2 | workers | 250 | 50 | 300 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 12 | 18 | 3 | ot_1 | 5 | 0 | 5 |
| 12 | 18 | 3 | ot_2 | 23 | 0 | 23 |
| 12 | 18 | 3 | workers | 28 | 5 | 33 |
| 12 | 18 | 4 | ot_1 | 5 | 0 | 5 |
| 12 | 18 | 4 | ot_2 | 23 | 0 | 23 |
| 12 | 18 | 4 | workers | 28 | 5 | 33 |
| 12 | 19 | 1 | ot_1 | 0 | 13 | 13 |
| 12 | 19 | 1 | ot_2 | 0 | 43 | 43 |
| 12 | 19 | 1 | workers | 0 | 74 | 74 |
| 12 | 19 | 2 | ot_1 | 0 | 20 | 20 |
| 12 | 19 | 2 | ot_2 | 0 | 66 | 66 |
| 12 | 19 | 2 | workers | 0 | 111 | 111 |
| 12 | 19 | 3 | ot_1 | 0 | 2 | 2 |
| 12 | 19 | 3 | ot_2 | 0 | 7 | 7 |
| 12 | 19 | 3 | workers | 0 | 11 | 11 |
| 12 | 19 | 4 | ot_1 | 0 | 2 | 2 |
| 12 | 19 | 4 | ot_2 | 0 | 7 | 7 |
| 12 | 19 | 4 | workers | 0 | 11 | 11 |
| 12 | 20 | 1 | ot_1 | 0 | 13 | 13 |
| 12 | 20 | 1 | ot_2 | 0 | 15 | 15 |
| 12 | 20 | 1 | workers | 0 | 74 | 74 |
| 12 | 20 | 2 | ot_1 | 0 | 21 | 21 |
| 12 | 20 | 2 | ot_2 | 0 | 23 | 23 |
| 12 | 20 | 2 | workers | 0 | 112 | 112 |
| 12 | 20 | 3 | ot_1 | 0 | 2 | 2 |
| 12 | 20 | 3 | ot_2 | 0 | 3 | 3 |
| 12 | 20 | 3 | workers | 0 | 11 | 11 |
| 12 | 20 | 4 | ot_1 | 0 | 2 | 2 |
| 12 | 20 | 4 | ot_2 | 0 | 3 | 3 |
| 12 | 20 | 4 | workers | 0 | 11 | 11 |
| 12 | 21 | 1 | ot_1 | 21 | 0 | 21 |
| 12 | 21 | 1 | ot_2 | 93 | 0 | 93 |
| 12 | 21 | 1 | workers | 113 | 0 | 113 |
| 12 | 21 | 2 | ot_1 | 86 | 0 | 86 |
| 12 | 21 | 2 | ot_2 | 381 | 0 | 381 |
| 12 | 21 | 2 | workers | 459 | 0 | 459 |
| 12 | 21 | 3 | ot_1 | 9 | 0 | 9 |
| 12 | 21 | 3 | ot_2 | 41 | 0 | 41 |
| 12 | 21 | 3 | workers | 52 | 0 | 52 |
| 12 | 21 | 4 | ot_1 | 9 | 0 | 9 |
| 12 | 21 | 4 | ot_2 | 41 | 0 | 41 |
| 12 | 21 | 4 | workers | 52 | 0 | 52 |
| 12 | 22 | 1 | ot_1 | 22 | 16 | 38 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 12 | 22 | 1 | ot_2 | 99 | 62 | 161 |
| 12 | 22 | 1 | workers | 119 | 90 | 209 |
| 12 | 22 | 2 | ot_1 | 90 | 25 | 115 |
| 12 | 22 | 2 | ot_2 | 403 | 95 | 498 |
| 12 | 22 | 2 | workers | 485 | 135 | 620 |
| 12 | 22 | 3 | ot_1 | 10 | 2 | 12 |
| 12 | 22 | 3 | ot_2 | 44 | 9 | 53 |
| 12 | 22 | 3 | workers | 55 | 14 | 69 |
| 12 | 22 | 4 | ot_1 | 10 | 2 | 12 |
| 12 | 22 | 4 | ot_2 | 44 | 9 | 53 |
| 12 | 22 | 4 | workers | 55 | 14 | 69 |
| 12 | 23 | 1 | ot_1 | 2 | 0 | 2 |
| 12 | 23 | 1 | ot_2 | 7 | 0 | 7 |
| 12 | 23 | 1 | workers | 11 | 0 | 11 |
| 12 | 23 | 2 | ot_1 | 7 | 0 | 7 |
| 12 | 23 | 2 | ot_2 | 34 | 0 | 34 |
| 12 | 23 | 2 | workers | 42 | 0 | 42 |
| 12 | 23 | 3 | ot_2 | 3 | 0 | 3 |
| 12 | 23 | 3 | workers | 5 | 0 | 5 |
| 12 | 23 | 4 | ot_2 | 3 | 0 | 3 |
| 12 | 23 | 4 | workers | 5 | 0 | 5 |
| 12 | 24 | 1 | ot_1 | 9 | 0 | 9 |
| 12 | 24 | 1 | ot_2 | 41 | 0 | 41 |
| 12 | 24 | 1 | workers | 50 | 0 | 50 |
| 12 | 24 | 2 | ot_1 | 38 | 0 | 38 |
| 12 | 24 | 2 | ot_2 | 168 | 0 | 168 |
| 12 | 24 | 2 | workers | 204 | 0 | 204 |
| 12 | 24 | 3 | ot_1 | 3 | 0 | 3 |
| 12 | 24 | 3 | ot_2 | 18 | 0 | 18 |
| 12 | 24 | 3 | workers | 23 | 0 | 23 |
| 12 | 24 | 4 | ot_1 | 3 | 0 | 3 |
| 12 | 24 | 4 | ot_2 | 18 | 0 | 18 |
| 12 | 24 | 4 | workers | 23 | 0 | 23 |
| 12 | 25 | 1 | ot_1 | 23 | 0 | 23 |
| 12 | 25 | 1 | ot_2 | 104 | 0 | 104 |
| 12 | 25 | 1 | workers | 127 | 0 | 127 |
| 12 | 25 | 2 | ot_1 | 96 | 0 | 96 |
| 12 | 25 | 2 | ot_2 | 427 | 0 | 427 |
| 12 | 25 | 2 | workers | 515 | 0 | 515 |
| 12 | 25 | 3 | ot_1 | 10 | 0 | 10 |
| 12 | 25 | 3 | ot_2 | 47 | 0 | 47 |
| 12 | 25 | 3 | workers | 58 | 0 | 58 |
| 12 | 25 | 4 | ot_1 | 11 | 0 | 11 |

| | | | | | | |
|---|---|---|---|---|---|---|
| 12 | 25 | 4 | ot_2 | 43 | 0 | 43 |
| 12 | 25 | 4 | workers | 61 | 0 | 61 |

## C. *Gurobi Optimization Code*

# Coppel Forecasting & Labor Optimization Tool

### Christian Gatmaitan ¶

Import necessary packages

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.tsa.api import ExponentialSmoothing, SimpleExpSmoothing, Holt
import gurobipy as gp
from gurobipy import GRB
from gurobipy import quicksum
import math
```

# Optimization Model

Load Data from Forecast, 'Capstone_Forecast_Coppel' must be run before this file

```
forecast = pd.read_csv("Forecast_Clothing2020.csv", index_col = 0)
```

```
forecast_furniture = pd.read_csv("Forecast_Furniture2020.csv", index_col = 0)
```

```
workforce = pd.read_csv("DC_workforce.csv", index_col = 0)
```

Add operational values

```
#create all input variables
dc = 25; #number of DCs
w = 4; #types of workers
m = w+1; #used for loop
t = 12; #number of time periods
o = t+1;
c_w = 11982; #Cost of employee ($/person/month)
c_f = 3.5*c_w; #cost of firing one employee ($/person)
c_h = 1668*(c_w/2); #cost of hiring one employee ($/person)
c_ot1 = c_w * 2; #Cost of overtime type 1 ($/hour)
c_ot2 = c_w * 3; #Cost of overtime type 2 ($/hour)
c_t = 930; #cost of transferring employee
c_c = 300000; #Cost of contractor ($/unit), set intentionally high to not utilize contractors
h = 48*4; #number monthly hours an employee works according to shift (hours/person)
m_ot1 = 9*4; #Max hours of OT1 allowed per employee per month (hrs/person)
m_ot2 = 40*4; #Max hours of OT2 allowed per employee per month (hrs/person)
#input cycle time data
l1 = 1/(.0122); #units/hr clothing picking
l2 = 1/(.0497); #units/hr clothing transfer
l3 = 1/(.0055556); #units/hr clothing load
l4 = 1/(.0055556); #units/hr clothing deliver
l5 = 1/(.044722); #units/hr furniture picking
l6 = 1/(.0675); #units/hr furniture transfer
l7 = 1/(.006944); #units/hr furniture load
l8 = 1/(.006944); #units/hr furniture deliver
l = [l1, l2, l3, l4, l5, l6, l7, l8];
col_names = ["Worker1", "Worker2", "Worker3", "Worker4", "Hire1", "Hire2",  "Hire3", "Hire4", "Fire1", "Fire2", "Fire3", "Fir
"Transfer2","Transfer3","Transfer4", "Contract1", "Contract2","Contract3","Contract4", "DC and Cost"];
```

```
d = forecast
dc_counter = 1
y = forecast_furniture
```

## Run loop that creates operational plan for every DC

```
output = pd.DataFrame()
```

```
output_furniture = pd.DataFrame()
```

```
for a in range(0,dc):
    # Initialize Model
    m = gp.Model("headcount")
    m.Params.timelimit = 3600.0     # set maximum runtime in seconds
    m.Params.MIPGap = 0.025         # set maximum allowable optimality gap (0.01 = 1%)
    #define variables
    workers = dict(
        [((i,k), m.addVar(vtype=GRB.INTEGER, lb=0.0, name="workers"))
         for i in range(1,w+1)
         for k in range(1,t+1)])
    hire = dict(
        [((i,k), m.addVar(vtype=GRB.INTEGER, lb=0.0, name="hire"))
         for i in range(1,w+1)
         for k in range(1,t+1)])
    fire = dict(
        [((i,k), m.addVar(vtype=GRB.INTEGER, lb=0.0, name="fire"))
         for i in range(1,w+1)
         for k in range(1,t+1)])
    ot_1 = dict(
        [((i,k), m.addVar(vtype=GRB.CONTINUOUS, lb=0.0, name="ot_1"))
         for i in range(1,w+1)
         for k in range(1,t+1)])
    ot_2 = dict(
        [((i,k), m.addVar(vtype=GRB.CONTINUOUS, lb=0.0, name="ot_2"))
         for i in range(1,w+1)
         for k in range(1,t+1)])
    transfer = dict(
        [((i,k), m.addVar(vtype=GRB.INTEGER, lb=0.0, name="transfer"))
         for i in range(1,w+1)
         for k in range(1,t+1)])
    contractunits = dict(
        [((i,k), m.addVar(vtype=GRB.INTEGER, lb=0.0, name="contractunits"))
         for i in range(1,w+1)
         for k in range(1,t+1)])
    m.update()
# Set objective (Minimize Total Cost)
m.setObjective(
    sum([hire[i, k] * c_h
         for i in range(1,w+1)
         for k in range(1,t+1)])
    +sum([workers[i, k] * c_w
          for i in range(1,w+1)
          for k in range(1,t+1)])
    +sum([fire[i, k] * c_f
          for i in range(1,w+1)
          for k in range(1,t+1)])
    +sum([ot_1[i, k] * c_ot1
          for i in range(1,w+1)
          for k in range(1,t+1)])
    +sum([ot_2[i, k] * c_ot2
          for i in range(1,w+1)
          for k in range(1,t+1)])
    +sum([transfer[i, k] * c_t
          for i in range(1,w+1)
          for k in range(1,t+1)])
    +sum([contractunits[i, k] * c_c
          for i in range(1,w+1)
          for k in range(1,t+1)])
    ,
    GRB.MINIMIZE)
#add constraints
for k in range(1, w+1):
    for i in range(1,t+1):
        m.addConstr((d.iloc[i-1, a] - ((workers[k,i]+ot_1[k,i]+ot_2[k,i])*h)*l[k-1] - contractunits[k,i])<= 0, name = 'De
        m.addConstr((ot_1[k,i]*h)-(m_ot1*workers[k,i]) <= 0, name = 'OT Type 1 Regulation')
        m.addConstr((ot_2[k,i]*h)-(m_ot2*workers[k,i]) <= 0, name = 'OT Type 2 Regulation')
    for i in range(1,t):
        m.addConstr((workers[1, i+1] == workers[1, i]+hire[1, i]-fire[1, i]), name = 'Workforce Consistency1')
        m.addConstr((workers[2, i+1] == workers[2, i]+hire[2, i]-fire[2, i]), name = 'Workforce Consistency2')
        m.addConstr((workers[3, i+1] == workers[3, i]+hire[3, i]-fire[3, i] + transfer[3, i]), name = 'Workforce Consiste
        m.addConstr((workers[4, i+1] == workers[4, i]+hire[4, i]-fire[4, i] - transfer[3, i]), name = 'Workforce Consiste
    m.addConstr((workers[1,1]>=workforce.iloc[a,1]), name = 'Initial Workforce Consistency3')
    m.addConstr((workers[2,1]>=workforce.iloc[a,2]), name = 'Initial Workforce Consistency4')
    m.addConstr((workers[3,1]>=workforce.iloc[a,5]), name = 'Initial Workforce Consistency5')
    m.addConstr((workers[4,1]>=workforce.iloc[a,6]), name = 'Initial Workforce Consistency6')
    m.addConstr((hire[1,1]==workers[1,1] - workforce.iloc[a,1]), name = 'Initial Workforce Consistency7')
    m.addConstr((hire[2,1]==workers[2,1] - workforce.iloc[a,2]), name = 'Initial Workforce Consistency8')
    m.addConstr((hire[3,1]==workers[3,1] - workforce.iloc[a,5]), name = 'Initial Workforce Consistency9')
    m.addConstr((hire[4,1]==workers[4,1] - workforce.iloc[a,6]), name = 'Initial Workforce Consistency10')
    m.addConstr((transfer[4, i] == 0), name = 'Transfer Constraint')
```

```
    # Optimize model
    m.optimize()
    #print results
    print('Minimal cost for DC ', a,'=' , m.ObjVal)
    v = m.getVars()
    results = pd.DataFrame(v)
    for i in range(0,len(v)):
        results['t'] = 1
        results['DC'] = 1
        results['Variable'] = 1
        results['Operation'] = 1
        results['Value_Clothing'] = 1

    for i in range(0,len(v)):
        results['Variable'][i] = v[i].varName
        results['Value_Clothing'][i] = v[i].x
        results['DC'][i] = a+1
        g = i/12
        g = math.floor(g) + 1
        r = math.floor(i/48)
        results['Operation'][i] = g-(r*4)
        e = math.floor(i/12)
        results['t'][i] = (i + 1) - (12*e)
    results.drop(0,inplace = True, axis = 1)
    output = output.append(results)
```

```
output.to_csv('Optimization_Clothing.csv')
```

```
for a in range(0,dc):
    # Initialize Model
    m = gp.Model("headcount")
    m.Params.timelimit = 3600.0     # set maximum runtime in seconds
    m.Params.MIPGap = 0.025         # set maximum allowable optimality gap (0.01 = 1%)
    #define variables
    workers = dict(
        [((i,k), m.addVar(vtype=GRB.INTEGER, lb=0.0, name="workers"))
         for i in range(1,w+1)
         for k in range(1,t+1)])
    hire = dict(
        [((i,k), m.addVar(vtype=GRB.INTEGER, lb=0.0, name="hire"))
         for i in range(1,w+1)
         for k in range(1,t+1)])
    fire = dict(
        [((i,k), m.addVar(vtype=GRB.INTEGER, lb=0.0, name="fire"))
         for i in range(1,w+1)
         for k in range(1,t+1)])
    ot_1 = dict(
        [((i,k), m.addVar(vtype=GRB.CONTINUOUS, lb=0.0, name="ot_1"))
         for i in range(1,w+1)
         for k in range(1,t+1)])
    ot_2 = dict(
        [((i,k), m.addVar(vtype=GRB.CONTINUOUS, lb=0.0, name="ot_2"))
         for i in range(1,w+1)
         for k in range(1,t+1)])
    transfer = dict(
        [((i,k), m.addVar(vtype=GRB.INTEGER, lb=0.0, name="transfer"))
         for i in range(1,w+1)
         for k in range(1,t+1)])
```

```
contractunits = dict(
    [((i,k), m.addVar(vtype=GRB.INTEGER, lb=0.0, name="contractunits"))
     for i in range(1,w+1)
     for k in range(1,t+1)])
m.update()
# Set objective (Minimize Total Cost)
m.setObjective(
    sum([hire[i, k] * c_h
         for i in range(1,w+1)
         for k in range(1,t+1)])
    +sum([workers[i, k] * c_w
          for i in range(1,w+1)
          for k in range(1,t+1)])
    +sum([fire[i, k] * c_f
          for i in range(1,w+1)
          for k in range(1,t+1)])
    +sum([ot_1[i, k] * c_ot1
          for i in range(1,w+1)
          for k in range(1,t+1)])
    +sum([ot_2[i, k] * c_ot2
          for i in range(1,w+1)
          for k in range(1,t+1)])
    +sum([transfer[i, k] * c_t
          for i in range(1,w+1)
          for k in range(1,t+1)])
    +sum([contractunits[i, k] * c_c
          for i in range(1,w+1)
          for k in range(1,t+1)])
    ,
    GRB.MINIMIZE)
#add constraints
for k in range(1, w+1):
    for i in range(1,t+1):
        m.addConstr((y.iloc[i-1, a] - ((workers[k,i]+ot_1[k,i]+ot_2[k,i])*h)*l[k+3] - contractunits[k,i])<= 0, name = 'De
        m.addConstr((ot_1[k,i]*h)-(m_ot1*workers[k,i]) <= 0, name = 'OT Type 1 Regulation')
        m.addConstr((ot_2[k,i]*h)-(m_ot2*workers[k,i]) <= 0, name = 'OT Type 2 Regulation')
    for i in range(1,t):
        m.addConstr((workers[1, i+1] == workers[1, i]+hire[1, i]-fire[1, i]), name = 'Workforce Consistency1')
        m.addConstr((workers[2, i+1] == workers[2, i]+hire[2, i]-fire[2, i]), name = 'Workforce Consistency2')
        m.addConstr((workers[3, i+1] == workers[3, i]+hire[3, i]-fire[3, i] + transfer[3, i]), name = 'Workforce Consiste
        m.addConstr((workers[4, i+1] == workers[4, i]+hire[4, i]-fire[4, i] - transfer[3, i]), name = 'Workforce Consiste
    m.addConstr((workers[1,1]>=workforce.iloc[a,3]), name = 'Initial Workforce Consistency3')
    m.addConstr((workers[2,1]>=workforce.iloc[a,4]), name = 'Initial Workforce Consistency4')
    m.addConstr((workers[3,1]>=workforce.iloc[a,5]), name = 'Initial Workforce Consistency5')
    m.addConstr((workers[4,1]>=workforce.iloc[a,6]), name = 'Initial Workforce Consistency6')
    m.addConstr((hire[1,1]==workers[1,1] - workforce.iloc[a,3]), name = 'Initial Workforce Consistency7')
    m.addConstr((hire[2,1]==workers[2,1] - workforce.iloc[a,4]), name = 'Initial Workforce Consistency8')
    m.addConstr((hire[3,1]==workers[3,1] - workforce.iloc[a,5]), name = 'Initial Workforce Consistency9')
    m.addConstr((hire[4,1]==workers[4,1] - workforce.iloc[a,6]), name = 'Initial Workforce Consistency10')
    m.addConstr((transfer[4, i] == 0), name = 'Transfer Constraint')
# Optimize model
m.optimize()
#print results
print('Minimal cost for DC ', a,'=' , m.ObjVal)
v = m.getVars()
results = pd.DataFrame(v)
for i in range(0,len(v)):
    results['t'] = 1
    results['DC'] = 1
    results['Variable'] = 1
    results['Operation'] = 1
    results['Value_Furniture'] = 1

for i in range(0,len(v)):
    results['Variable'][i] = v[i].varName
    results['Value_Furniture'][i] = v[i].x
    results['DC'][i] = a+1
    g = i/12
    g = math.floor(g) + 1
    r = math.floor(i/48)
    results['Operation'][i] = g-(r*4)
    e = math.floor(i/12)
    results['t'][i] = (i + 1) - (12*e)
results.drop(0,inplace = True, axis = 1)
output_furniture = output_furniture.append(results)
```

```
output_furniture.to_csv('Optimization_Furniture.csv')
```

```
final_output = pd.merge(output, output_furniture,  how='left', left_on=['Variable','DC', 'Operation', 't'], right_on = ['Vari
```

```
output_pivot = final_output.pivot_table(index=['t', 'DC', 'Operation', 'Variable'], values=['Value_Clothing', 'Value_Furnitur
```

```
output_pivot['Value_Total'] = output_pivot['Value_Clothing'] + output_pivot['Value_Furniture']
```

```
output_pivot
```

```
output_final = output_pivot[output_pivot.Value_Total != 0]
```

```
output_final.to_csv('Optimization_Coppel.csv')
```

# D. *Forecasting Script*

## Raw Data Upload & Pivot Creation

### Christian Gatmaitan

Import Necessary Packages

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import math
```

In [2]:
```
from statsmodels.tsa.api import ExponentialSmoothing, SimpleExpSmoothing, Holt
```

Load data files

In [3]:
```
#import data
df_clothing17 = pd.read_csv("Ventas2017_CUML_.txt", sep='|', encoding='latin-1')
```

In [4]:
```
df_clothing18 = pd.read_csv("Ventas_Ropa_2018.txt", sep='|', encoding='latin-1')
```

In [5]:
```
df_clothing19 = pd.read_csv("Ventas_Ropa_2019 (2).txt", sep='|', encoding='latin-1')
```

Rename file headers to be consistent

In [6]:
```
df_clothing17.rename(columns = {'Departamento':'NumDepto', 'Clase':'Numclase', 'Familia':'NumFamilia', 'NumTienda':'numtienda
```

In [ ]:
```
DCnames = df_clothing17.pivot_table(index=['Bodega', 'NomBodega'], values=['VtaUnidades'], aggfunc='sum')df_clothing17
```

Pivot data on Year, Month, and DC by units sold

In [7]:
```
clothes2017 = df_clothing17.pivot_table(index=['Anio', 'Mes', 'Bodega'], values=['VtaUnidades'], aggfunc='sum')
```

In [8]:
```
clothes2018 = df_clothing18.pivot_table(index=['Anio', 'Mes', 'Bodega'], values=['VtaUnidades'], aggfunc='sum')
```

In [9]:
```
clothes2019 = df_clothing19.pivot_table(index=['Anio', 'Mes', 'Bodega'], values=['VtaUnidades'], aggfunc='sum')
```

In [10]:
```
clothes2017 = clothes2017.reset_index()
```

In [11]:
```
clothes2018 = clothes2018.reset_index()
```

In [12]:
```
clothes2019 = clothes2019.reset_index()
```

Write outputs to csv

[13]:
```
clothes2017.to_csv("clothing2017.csv")
```

[14]:
```
clothes2018.to_csv("clothing2018.csv")
```

[15]:
```
clothes2019.to_csv("clothing2019.csv")
```

Import furniture Data

[64]:
```
df_furniture = pd.read_csv("Ventas_2017,2018,2019_DCF.txt", sep='|', encoding='latin-1')
```

Convert weeks to months

[65]:
```
df_furniture['Semanafloat'] = df_furniture['Semana'].astype(float)
```

[22]:
```
df_furniture['Mes'] = np.ceil(df_furniture['Semanafloat']/4.345)
```

Pivot Data on Year, Month, DC by Units Sold

[19]:
```
df_furniture = df_furniture.pivot_table(index=['Anio', 'Mes', 'Bodega'], values=['Cantidad'], aggfunc='sum')
```

[20]:
```
df_furniture = df_furniture.reset_index()
```

Write results to CSV

[21]:
```
df_furniture.to_csv("furniture.csv")
```

# Coppel Forecasting & Labor Optimization Tool

## Christian Gatmaitan

Import necessary packages

[1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from statsmodels.tsa.api import ExponentialSmoothing, SimpleExpSmoothing, Holt
import gurobipy as gp
from gurobipy import GRB
from gurobipy import quicksum
```

Read past sales data files created from Pivot Creation file

[2]:
```
#read data for furniture and clothing
df_2017 = pd.read_csv("clothing2017.csv", index_col = 0)
df_2018 = pd.read_csv("clothing2018.csv", index_col = 0)
df_2019 = pd.read_csv("clothing2019.csv", index_col = 0)
df_furniture = pd.read_csv("furniture.csv", index_col = 0)
```

Update DC column in furniture file due to difference in naming convention

[409]:
```
df_furniture['Bodega'] = df_furniture['Bodega']-30000 #update Bodega for furniture
```

## Create forecast for clothing

Merge all historical data into one dataframe

[410]:
```
#merge 2017 and 2018 dataframes
df_201718 = df_2017.append(df_2018)
```

[411]:
```
#merge 2017-18 w/ 2019
df_allsales = df_201718.append(df_2019)
```

Convert to English and create datetime column that merges date, month, and year

[412]:
```
df_allsales['Day'] = 1
df_allsales['Month'] = df_allsales['Mes']
df_allsales['Year'] = df_allsales['Anio']
```

[413]:
```
df_allsales['Datetime'] = pd.to_datetime(df_allsales[['Year', 'Month', 'Day']])
```

[414]:
```
df_allsales['Count']= df_allsales['VtaUnidades']
```

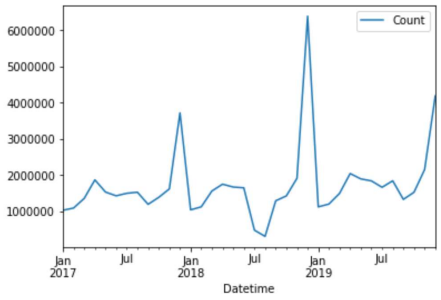
[415]:
```
df_allsales.drop(['Anio', 'Mes', 'Day','Month','Year','VtaUnidades'], axis=1, inplace = True)
```

Create a variable for sales at each distribution center

[416]:
```
dc1 = df_allsales[df_allsales.Bodega == 1]
dc2 = df_allsales[df_allsales.Bodega == 2]
dc3 = df_allsales[df_allsales.Bodega == 3]
dc4 = df_allsales[df_allsales.Bodega == 4]
dc5 = df_allsales[df_allsales.Bodega == 5]
dc6 = df_allsales[df_allsales.Bodega == 6]
dc7 = df_allsales[df_allsales.Bodega == 7]
dc8 = df_allsales[df_allsales.Bodega == 8]
dc9 = df_allsales[df_allsales.Bodega == 9]
dc10 = df_allsales[df_allsales.Bodega == 10]
dc11 = df_allsales[df_allsales.Bodega == 11]
dc12 = df_allsales[df_allsales.Bodega == 12]
dc13 = df_allsales[df_allsales.Bodega == 13]
dc14 = df_allsales[df_allsales.Bodega == 14]
dc15 = df_allsales[df_allsales.Bodega == 15]
dc16 = df_allsales[df_allsales.Bodega == 16]
dc17 = df_allsales[df_allsales.Bodega == 17]
dc18 = df_allsales[df_allsales.Bodega == 18]
dc19 = df_allsales[df_allsales.Bodega == 19]
dc20 = df_allsales[df_allsales.Bodega == 20]
dc21 = df_allsales[df_allsales.Bodega == 21]
dc22 = df_allsales[df_allsales.Bodega == 22]
dc23 = df_allsales[df_allsales.Bodega == 23]
dc24 = df_allsales[df_allsales.Bodega == 24]
dc25 = df_allsales[df_allsales.Bodega == 25]
dc26 = df_allsales[df_allsales.Bodega == 26]
dc27 = df_allsales[df_allsales.Bodega == 27]
```

Test plot to get basic understanding of historical demand

[417]:
```
dc1.plot.line(x = 'Datetime',
              y = 'Count')
plt.show()
```




## Create Forecasts by DC

[418]:
```
data = dc1.Count.values #load historical data for dc
index = pd.date_range(start='2017', end='2020', freq='M') #set date range
fulldata = pd.Series(data, index) #create series off data and timeframe

#use exponential smoothign package to try all 4 Holt-Winter methods
fit1 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='add').fit(use_boxcox=True)
fit2 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='mul').fit(use_boxcox=True)
fit3 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='add', damped=True).fit(use_boxcox=True)
fit4 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='mul', damped=True).fit(use_boxcox=True)
results=pd.DataFrame(index=[r"$\alpha$",r"$\beta$",r"$\phi$",r"$\gamma$",r"$l_0$","$b_0$","SSE"])
params = ['smoothing_level', 'smoothing_slope', 'damping_slope', 'smoothing_seasonal', 'initial_level', 'initial_slope']
results["Additive"]       = [fit1.params[p] for p in params] + [fit1.sse]
results["Multiplicative"] = [fit2.params[p] for p in params] + [fit2.sse]
results["Additive Dam"]   = [fit3.params[p] for p in params] + [fit3.sse]
results["Multiplica Dam"] = [fit4.params[p] for p in params] + [fit4.sse]

ax = fulldata.plot(figsize=(10,6), marker='o', color='black', title="Forecasts from Holt-Winters' multiplicative method" )
ax.set_ylabel("Sales")
ax.set_xlabel("Date")
fit1.fittedvalues.plot(ax=ax, style='--', color='red')
fit2.fittedvalues.plot(ax=ax, style='--', color='green')

#fit1.forecast(12).rename('Holt-Winters (add-add-seasonal)').plot(ax=ax, style='--', marker='o', color='red', legend=True)
#fit2.forecast(12).rename('Holt-Winters (add-mul-seasonal)').plot(ax=ax, style='--', marker='o', color='green', legend=True)

plt.show()
print("Figure: Forecasting Coppel Sales for the next year using Holt-Winters method with both additive and multiplicative sea

results
results1 = fit2.forecast(12).values
results
```

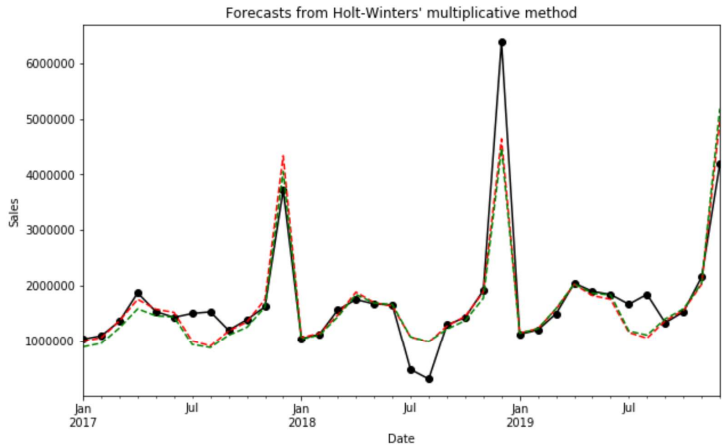



Figure: Forecasting Coppel Sales for the next year using Holt-Winters method with both additive and multiplicative seasonality.

Calculate Error to determine best fit

[419]:
```
#Calculate Error of each fit and write to CSV
df_fit = pd.DataFrame(fit1.fittedvalues)
df_fit['Fit2'] = fit2.fittedvalues
df_fit['Fit3'] = fit3.fittedvalues
df_fit['Fit4'] = fit4.fittedvalues
df_fit = df_fit.applymap(str)
df_fit['Fit1'] = df_fit[0].str.split(' ').str[0]
df_fit['Fit2'] = df_fit['Fit2'].str.split(' ').str[0]
df_fit['Fit3'] = df_fit['Fit3'].str.split(' ').str[0]
df_fit['Fit4'] = df_fit['Fit4'].str.split(' ').str[0]
df_fit = df_fit.applymap(float)
df_fit['Actual'] = fulldata.values
count = pd.DataFrame.count(df_fit)
df_fit['Error1'] = (abs(df_fit['Fit1']-df_fit['Actual'])/df_fit['Actual'])*100
df_fit['MAPE1'] = sum(df_fit['Error1'])/(count[0])

df_fit['Error2'] = (abs(df_fit['Fit2']-df_fit['Actual'])/df_fit['Actual'])*100
df_fit['MAPE2'] = sum(df_fit['Error2'])/(count[0])

df_fit['Error3'] = (abs(df_fit['Fit3']-df_fit['Actual'])/df_fit['Actual'])*100
df_fit['MAPE3'] = sum(df_fit['Error3'])/(count[0])

df_fit['Error4'] = (abs(df_fit['Fit4']-df_fit['Actual'])/df_fit['Actual'])*100
df_fit['MAPE4'] = sum(df_fit['Error4'])/(count[0])
df_fit.to_csv('MAPE.csv')
```

[420]:
```
#same for DC2
data = dc2.Count.values
index = pd.date_range(start='2017', end='2020', freq='M')
fulldata = pd.Series(data, index)

fit1 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='add').fit(use_boxcox=True)
fit2 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='mul').fit(use_boxcox=True)
fit3 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='add', damped=True).fit(use_boxcox=True)
fit4 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='mul', damped=True).fit(use_boxcox=True)
results=pd.DataFrame(index=[r"$\alpha$",r"$\beta$",r"$\phi$",r"$\gamma$",r"$l_0$","$b_0$","SSE"])
params = ['smoothing_level', 'smoothing_slope', 'damping_slope', 'smoothing_seasonal', 'initial_level', 'initial_slope']
results["Additive"]       = [fit1.params[p] for p in params] + [fit1.sse]
results["Multiplicative"] = [fit2.params[p] for p in params] + [fit2.sse]
results["Additive Dam"]   = [fit3.params[p] for p in params] + [fit3.sse]
results["Multiplica Dam"] = [fit4.params[p] for p in params] + [fit4.sse]

ax = fulldata.plot(figsize=(10,6), marker='o', color='black', title="Forecasts from Holt-Winters' multiplicative method" )
ax.set_ylabel("Sales")
ax.set_xlabel("Date")
fit1.fittedvalues.plot(ax=ax, style='--', color='red')
fit2.fittedvalues.plot(ax=ax, style='--', color='green')

fit1.forecast(12).rename('Holt-Winters (add-add-seasonal)').plot(ax=ax, style='--', marker='o', color='red', legend=True)
fit2.forecast(12).rename('Holt-Winters (add-mul-seasonal)').plot(ax=ax, style='--', marker='o', color='green', legend=True)

plt.show()
print("Figure: Forecasting Coppel Sales for the next year using Holt-Winters method with both additive and multiplicative sea

results
results2 = fit2.forecast(12).values
results2
```

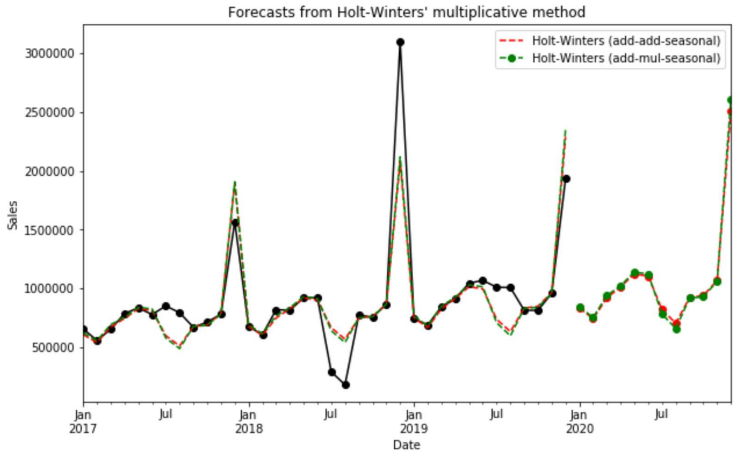



Figure: Forecasting Coppel Sales for the next year using Holt-Winters method with both additive and multiplicative seasonality.

420]:
```
array([ 845311.98014748,  752663.84064513,  939219.41582212,
       1023968.84328097, 1139798.19097027, 1121739.58364396,
        782275.27947501,  658628.66879958,  917615.84287607,
        928025.52105461, 1060085.12281721, 2605934.10369977])
```

[421]:
```
#same for DC3
data = dc3.Count.values
index = pd.date_range(start='2017', end='2020', freq='M')
fulldata = pd.Series(data, index)

fit1 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='add').fit(use_boxcox=True)
fit2 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='mul').fit(use_boxcox=True)
fit3 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='add', damped=True).fit(use_boxcox=True)
fit4 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='mul', damped=True).fit(use_boxcox=True)
results=pd.DataFrame(index=[r"$\alpha$",r"$\beta$",r"$\phi$",r"$\gamma$",r"$l_0$","$b_0$","SSE"])
params = ['smoothing_level', 'smoothing_slope', 'damping_slope', 'smoothing_seasonal', 'initial_level', 'initial_slope']
results["Additive"]       = [fit1.params[p] for p in params] + [fit1.sse]
results["Multiplicative"] = [fit2.params[p] for p in params] + [fit2.sse]
results["Additive Dam"]   = [fit3.params[p] for p in params] + [fit3.sse]
results["Multiplica Dam"] = [fit4.params[p] for p in params] + [fit4.sse]

ax = fulldata.plot(figsize=(10,6), marker='o', color='black', title="Forecasts from Holt-Winters' multiplicative method" )
ax.set_ylabel("Sales")
ax.set_xlabel("Date")
fit1.fittedvalues.plot(ax=ax, style='--', color='red')
fit2.fittedvalues.plot(ax=ax, style='--', color='green')

fit1.forecast(12).rename('Holt-Winters (add-add-seasonal)').plot(ax=ax, style='--', marker='o', color='red', legend=True)
fit2.forecast(12).rename('Holt-Winters (add-mul-seasonal)').plot(ax=ax, style='--', marker='o', color='green', legend=True)

plt.show()
print("Figure: Forecasting Coppel Sales for the next year using Holt-Winters method with both additive and multiplicative sea

results
results3 = fit2.forecast(12).values
results3
```

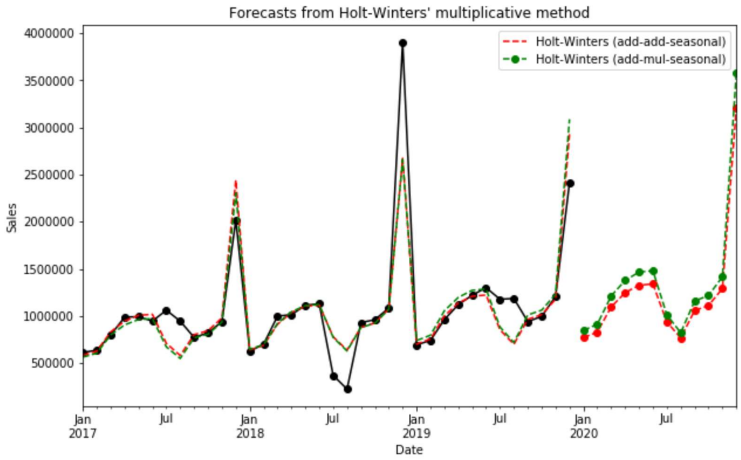



Figure: Forecasting Coppel Sales for the next year using Holt-Winters method with both additive and multiplicative seasonality.

21]:
```
array([ 846555.25847647,  911839.49109609, 1212441.4625222 ,
       1376860.43552741, 1465306.3679274 , 1480432.50043106,
       1012699.80786216,  821490.29382579, 1158301.68384173,
       1217087.52576968, 1415100.48064288, 3575508.93422427])
```

[422]:
```
#same for DC4
data = dc4.Count.values
index = pd.date_range(start='2017', end='2020', freq='M')
fulldata = pd.Series(data, index)

fit1 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='add').fit(use_boxcox=True)
fit2 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='mul').fit(use_boxcox=True)
fit3 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='add', damped=True).fit(use_boxcox=True)
fit4 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='mul', damped=True).fit(use_boxcox=True)
results=pd.DataFrame(index=[r"$\alpha$",r"$\beta$",r"$\phi$",r"$\gamma$",r"$l_0$","$b_0$","SSE"])
params = ['smoothing_level', 'smoothing_slope', 'damping_slope', 'smoothing_seasonal', 'initial_level', 'initial_slope']
results["Additive"]       = [fit1.params[p] for p in params] + [fit1.sse]
results["Multiplicative"] = [fit2.params[p] for p in params] + [fit2.sse]
results["Additive Dam"]   = [fit3.params[p] for p in params] + [fit3.sse]
results["Multiplica Dam"] = [fit4.params[p] for p in params] + [fit4.sse]

ax = fulldata.plot(figsize=(10,6), marker='o', color='black', title="Forecasts from Holt-Winters' multiplicative method" )
ax.set_ylabel("Sales")
ax.set_xlabel("Date")
fit1.fittedvalues.plot(ax=ax, style='--', color='red')
fit2.fittedvalues.plot(ax=ax, style='--', color='green')

fit1.forecast(12).rename('Holt-Winters (add-add-seasonal)').plot(ax=ax, style='--', marker='o', color='red', legend=True)
fit2.forecast(12).rename('Holt-Winters (add-mul-seasonal)').plot(ax=ax, style='--', marker='o', color='green', legend=True)

plt.show()
print("Figure: Forecasting Coppel Sales for the next year using Holt-Winters method with both additive and multiplicative sea

results
results4 = fit2.forecast(12).values
results4
```

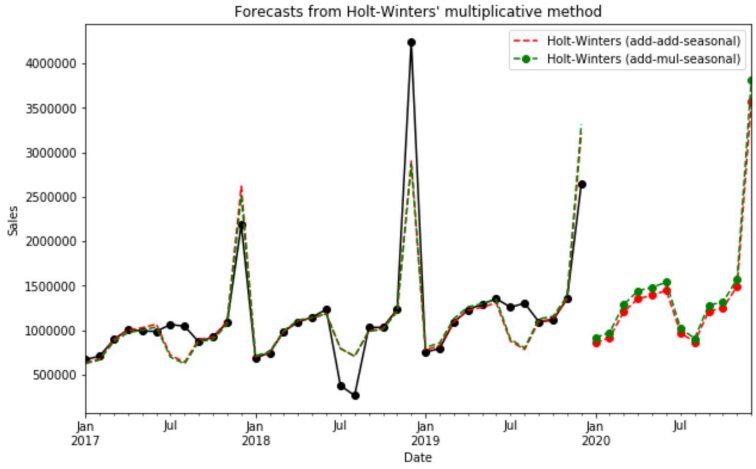



Figure: Forecasting Coppel Sales for the next year using Holt-Winters method with both additive and multiplicative seasonali
ty.

']: array([ 914983.72170579,  973679.77497416, 1288337.52160042,
       1442485.66757471, 1482026.36729394, 1538549.65150034,
       1021274.17518183,  906347.85422417, 1282077.15343847,
       1314730.2696541 , 1575678.09703124, 3813703.83683655])

[423]:
```
#same for DC5
data = dc5.Count.values
index = pd.date_range(start='2017', end='2020', freq='M')
fulldata = pd.Series(data, index)

fit1 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='add').fit(use_boxcox=True)
fit2 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='mul').fit(use_boxcox=True)
fit3 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='add', damped=True).fit(use_boxcox=True)
fit4 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='mul', damped=True).fit(use_boxcox=True)
results=pd.DataFrame(index=[r"$\alpha$",r"$\beta$",r"$\phi$",r"$\gamma$",r"$l_0$","$b_0$","SSE"])
params = ['smoothing_level', 'smoothing_slope', 'damping_slope', 'smoothing_seasonal', 'initial_level', 'initial_slope']
results["Additive"]       = [fit1.params[p] for p in params] + [fit1.sse]
results["Multiplicative"] = [fit2.params[p] for p in params] + [fit2.sse]
results["Additive Dam"]   = [fit3.params[p] for p in params] + [fit3.sse]
results["Multiplica Dam"] = [fit4.params[p] for p in params] + [fit4.sse]

ax = fulldata.plot(figsize=(10,6), marker='o', color='black', title="Forecasts from Holt-Winters' multiplicative method" )
ax.set_ylabel("Sales")
ax.set_xlabel("Date")
fit1.fittedvalues.plot(ax=ax, style='--', color='red')
fit2.fittedvalues.plot(ax=ax, style='--', color='green')

fit1.forecast(12).rename('Holt-Winters (add-add-seasonal)').plot(ax=ax, style='--', marker='o', color='red', legend=True)
fit2.forecast(12).rename('Holt-Winters (add-mul-seasonal)').plot(ax=ax, style='--', marker='o', color='green', legend=True)

plt.show()
print("Figure: Forecasting Coppel Sales for the next year using Holt-Winters method with both additive and multiplicative sea

results
results5 = fit2.forecast(12).values
results5
```

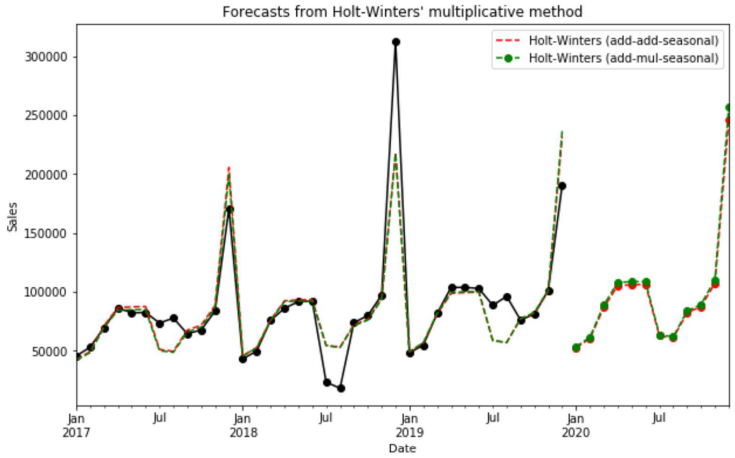



Figure: Forecasting Coppel Sales for the next year using Holt-Winters method with both additive and multiplicative seasonality.

23]:
```
array([ 53491.96509101,  61646.21098327,  88980.69634214, 108121.58816201,
       108685.36941624, 108729.64720132,  63764.69542718,  61883.70448845,
        83841.13970257,  89457.2555306 , 110499.31252407, 257453.04918276])
```

[424]:
```
#same for DC6
data = dc6.Count.values
index = pd.date_range(start='2017', end='2020', freq='M')
fulldata = pd.Series(data, index)

fit1 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='add').fit(use_boxcox=True)
fit2 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='mul').fit(use_boxcox=True)
fit3 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='add', damped=True).fit(use_boxcox=True)
fit4 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='mul', damped=True).fit(use_boxcox=True)
results=pd.DataFrame(index=[r"$\alpha$",r"$\beta$",r"$\phi$",r"$\gamma$",r"$l_0$","$b_0$","SSE"])
params = ['smoothing_level', 'smoothing_slope', 'damping_slope', 'smoothing_seasonal', 'initial_level', 'initial_slope']
results["Additive"]       = [fit1.params[p] for p in params] + [fit1.sse]
results["Multiplicative"] = [fit2.params[p] for p in params] + [fit2.sse]
results["Additive Dam"]   = [fit3.params[p] for p in params] + [fit3.sse]
results["Multiplica Dam"] = [fit4.params[p] for p in params] + [fit4.sse]

ax = fulldata.plot(figsize=(10,6), marker='o', color='black', title="Forecasts from Holt-Winters' multiplicative method" )
ax.set_ylabel("Sales")
ax.set_xlabel("Date")
fit1.fittedvalues.plot(ax=ax, style='--', color='red')
fit2.fittedvalues.plot(ax=ax, style='--', color='green')

fit1.forecast(12).rename('Holt-Winters (add-add-seasonal)').plot(ax=ax, style='--', marker='o', color='red', legend=True)
fit2.forecast(12).rename('Holt-Winters (add-mul-seasonal)').plot(ax=ax, style='--', marker='o', color='green', legend=True)

plt.show()
print("Figure: Forecasting Coppel Sales for the next year using Holt-Winters method with both additive and multiplicative sea

results
results6 = fit2.forecast(12).values
results6
```

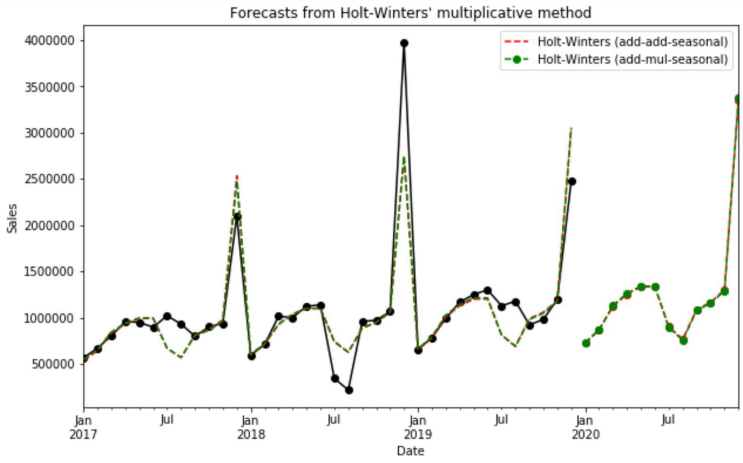



Figure: Forecasting Coppel Sales for the next year using Holt-Winters method with both additive and multiplicative seasonality.

24]:
```
array([ 729192.68737342,  872454.38297908, 1132982.21061834,
       1262843.33111656, 1340888.14696432, 1338019.51430761,
        893242.07635145,  755235.06158678, 1080406.86194971,
       1154396.58457513, 1291192.02482011, 3378026.45817887])
```

[425]:
```
#same for DC7
data = dc7.Count.values
index = pd.date_range(start='2017', end='2020', freq='M')
fulldata = pd.Series(data, index)

fit1 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='add').fit(use_boxcox=True)
fit2 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='mul').fit(use_boxcox=True)
fit3 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='add', damped=True).fit(use_boxcox=True)
fit4 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='mul', damped=True).fit(use_boxcox=True)
results=pd.DataFrame(index=[r"$\alpha$",r"$\beta$",r"$\phi$",r"$\gamma$",r"$l_0$","$b_0$","SSE"])
params = ['smoothing_level', 'smoothing_slope', 'damping_slope', 'smoothing_seasonal', 'initial_level', 'initial_slope']
results["Additive"]       = [fit1.params[p] for p in params] + [fit1.sse]
results["Multiplicative"] = [fit2.params[p] for p in params] + [fit2.sse]
results["Additive Dam"]   = [fit3.params[p] for p in params] + [fit3.sse]
results["Multiplica Dam"] = [fit4.params[p] for p in params] + [fit4.sse]

ax = fulldata.plot(figsize=(10,6), marker='o', color='black', title="Forecasts from Holt-Winters' multiplicative method" )
ax.set_ylabel("Sales")
ax.set_xlabel("Date")
fit1.fittedvalues.plot(ax=ax, style='--', color='red')
fit2.fittedvalues.plot(ax=ax, style='--', color='green')

fit1.forecast(12).rename('Holt-Winters (add-add-seasonal)').plot(ax=ax, style='--', marker='o', color='red', legend=True)
fit2.forecast(12).rename('Holt-Winters (add-mul-seasonal)').plot(ax=ax, style='--', marker='o', color='green', legend=True)

plt.show()
print("Figure: Forecasting Coppel Sales for the next year using Holt-Winters method with both additive and multiplicative sea

results
results7 = fit2.forecast(12).values
results7
```

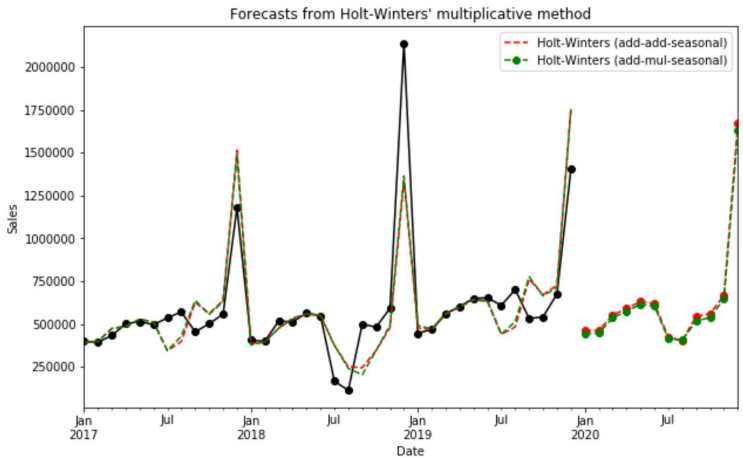



Figure: Forecasting Coppel Sales for the next year using Holt-Winters method with both additive and multiplicative seasonality.

:5]:
```
array([ 443574.22703354,  446950.73260554,  535200.48331427,
        574622.58247744,  615078.77738895,  604516.23166391,
        412415.28519196,  405243.55728202,  516069.70262719,
        538754.18606109,  645048.84092103, 1629773.75012489])
```

[426]:
```
#same for DC8
data = dc8.Count.values
index = pd.date_range(start='2017', end='2020', freq='M')
fulldata = pd.Series(data, index)

fit1 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='add').fit(use_boxcox=True)
fit2 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='mul').fit(use_boxcox=True)
fit3 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='add', damped=True).fit(use_boxcox=True)
fit4 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='mul', damped=True).fit(use_boxcox=True)
results=pd.DataFrame(index=[r"$\alpha$",r"$\beta$",r"$\phi$",r"$\gamma$",r"$l_0$","$b_0$","SSE"])
params = ['smoothing_level', 'smoothing_slope', 'damping_slope', 'smoothing_seasonal', 'initial_level', 'initial_slope']
results["Additive"]       = [fit1.params[p] for p in params] + [fit1.sse]
results["Multiplicative"] = [fit2.params[p] for p in params] + [fit2.sse]
results["Additive Dam"]   = [fit3.params[p] for p in params] + [fit3.sse]
results["Multiplica Dam"] = [fit4.params[p] for p in params] + [fit4.sse]

ax = fulldata.plot(figsize=(10,6), marker='o', color='black', title="Forecasts from Holt-Winters' multiplicative method" )
ax.set_ylabel("Sales")
ax.set_xlabel("Date")
fit1.fittedvalues.plot(ax=ax, style='--', color='red')
fit2.fittedvalues.plot(ax=ax, style='--', color='green')

fit1.forecast(12).rename('Holt-Winters (add-add-seasonal)').plot(ax=ax, style='--', marker='o', color='red', legend=True)
fit2.forecast(12).rename('Holt-Winters (add-mul-seasonal)').plot(ax=ax, style='--', marker='o', color='green', legend=True)

plt.show()
print("Figure: Forecasting Coppel Sales for the next year using Holt-Winters method with both additive and multiplicative sea

results
results8 = fit2.forecast(12).values
results8
```

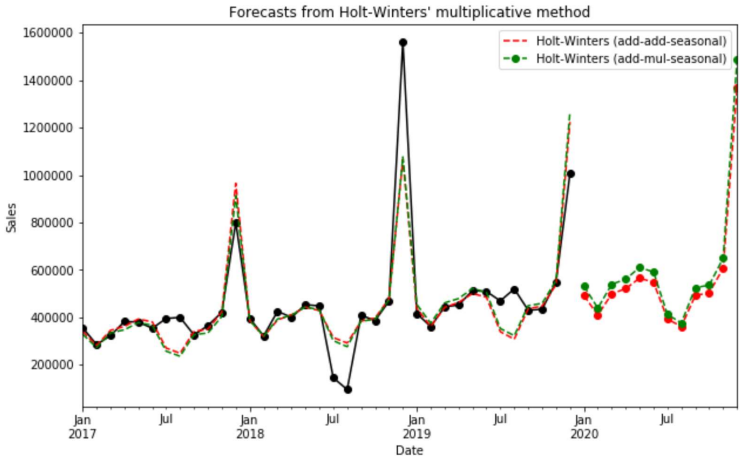



Figure: Forecasting Coppel Sales for the next year using Holt-Winters method with both additive and multiplicative seasonality.

126]:
```
array([ 532443.46292986,  439346.46289892,  539770.31450524,
        561297.88885389,  610676.93864705,  591818.8871553 ,
        412246.71149288,  376809.10965182,  525413.50630122,
        536302.27347755,  649753.5121795 , 1489065.4501512 ])
```

[427]:
```
#same for DC9
data = dc9.Count.values
index = pd.date_range(start='2017', end='2020', freq='M')
fulldata = pd.Series(data, index)

fit1 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='add').fit(use_boxcox=True)
fit2 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='mul').fit(use_boxcox=True)
fit3 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='add', damped=True).fit(use_boxcox=True)
fit4 = ExponentialSmoothing(fulldata, seasonal_periods=12, trend='add', seasonal='mul', damped=True).fit(use_boxcox=True)
results=pd.DataFrame(index=[r"$\alpha$",r"$\beta$",r"$\phi$",r"$\gamma$",r"$l_0$","$b_0$","SSE"])
params = ['smoothing_level', 'smoothing_slope', 'damping_slope', 'smoothing_seasonal', 'initial_level', 'initial_slope']
results["Additive"]       = [fit1.params[p] for p in params] + [fit1.sse]
results["Multiplicative"] = [fit2.params[p] for p in params] + [fit2.sse]
results["Additive Dam"]   = [fit3.params[p] for p in params] + [fit3.sse]
results["Multiplica Dam"] = [fit4.params[p] for p in params] + [fit4.sse]

ax = fulldata.plot(figsize=(10,6), marker='o', color='black', title="Forecasts from Holt-Winters' multiplicative method" )
ax.set_ylabel("Sales")
ax.set_xlabel("Date")
fit1.fittedvalues.plot(ax=ax, style='--', color='red')
fit2.fittedvalues.plot(ax=ax, style='--', color='green')

fit1.forecast(12).rename('Holt-Winters (add-add-seasonal)').plot(ax=ax, style='--', marker='o', color='red', legend=True)
fit2.forecast(12).rename('Holt-Winters (add-mul-seasonal)').plot(ax=ax, style='--', marker='o', color='green', legend=True)

plt.show()
print("Figure: Forecasting Coppel Sales for the next year using Holt-Winters method with both additive and multiplicative sea

results
results9 = fit2.forecast(12).values
results9
```

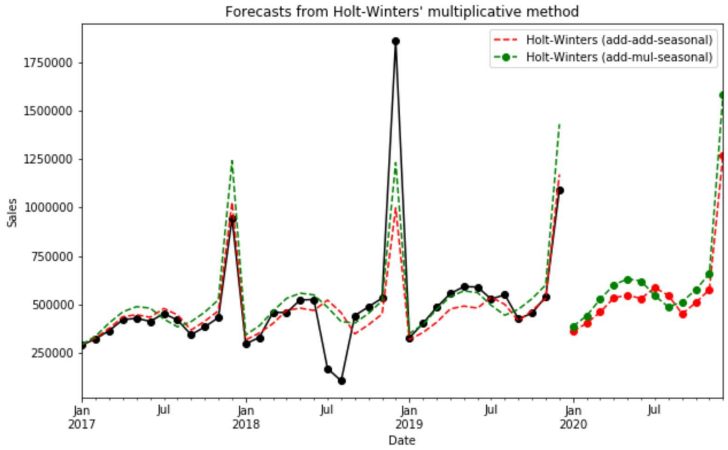



Figure: Forecasting Coppel Sales for the next year using Holt-Winters method with both additive and multiplicative seasonality.

[427]:
```
array([ 387342.06276662,  441299.35174721,  528061.95078795,
        600767.77444793,  632187.30305694,  620298.26828707,
        545749.08663091,  486658.4092315 ,  513838.59649823,
        575913.32606872,  657952.37670182, 1583524.8309289 ])
```

## Create dataframe for results and add each DC

[446]:
```
df_results = pd.DataFrame(results1)
df_results['DC1'] = (results1)
df_results['DC2'] = (results2)
df_results['DC3'] = (results3)
df_results['DC4'] = (results4)
df_results['DC5'] = (results5)
df_results['DC6'] = (results6)
df_results['DC7'] = (results7)
df_results['DC8'] = (results8)
df_results['DC9'] = (results9)
df_results['DC10'] = (results10)
df_results['DC11'] = (results11)
df_results['DC12'] = (results12)
df_results['DC13'] = (results13)
df_results['DC14'] = (results14)
df_results['DC15'] = (results15)
df_results['DC16'] = (results16)
df_results['DC17'] = (results17)
df_results['DC18'] = (results18)
df_results['DC19'] = (results19)
df_results['DC20'] = (results20)
df_results['DC21'] = (results21)
df_results['DC22'] = (results22)
df_results['DC23'] = (results23)
df_results['DC24'] = (results24)
df_results['DC25'] = (results25)
df_results['DC26'] = (results26)
df_results['DC27'] = (results27)
```

[447]:
```
df_results.drop(0,axis=1, inplace = True)
```

[448]:
```
df_results['Month'] = [1,2,3,4,5,6,7,8,9,10,11,12] #add column for month
```

[449]:
```
df_results.to_csv('Forecast_Clothing2020.csv') #write forecast to CSV
```

[535]:
```
df_results
```

Out[535]:

| | DC1 | DC2 | DC3 | DC4 | DC5 | DC6 | DC7 | DC8 | DC9 | DC10 |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 244355.250883 | 309917.541325 | 78158.995393 | 257645.226088 | 0 | 338023.266361 | 112480.878288 | 193568.088283 | 491186.485295 | 321588.692851 |
| 1 | 238388.441803 | 243928.468762 | 77964.565838 | 240208.061617 | 0 | 350054.780012 | 101886.628285 | 135940.142216 | 361638.474267 | 282384.024291 |
| 2 | 332786.007233 | 314422.013208 | 103691.640990 | 313655.973321 | 0 | 464021.105971 | 121090.903609 | 168308.554374 | 445610.848934 | 340174.999208 |
| 3 | 227663.350439 | 249101.417221 | 83747.268450 | 255490.896728 | 0 | 365037.644983 | 100408.028675 | 127002.516040 | 344837.343910 | 272742.481725 |
| 4 | 309908.610555 | 329513.806308 | 115775.208418 | 346682.996771 | 0 | 498608.545134 | 112480.878288 | 147495.663153 | 394421.684629 | 306311.374410 |
| 5 | 429196.152840 | 387618.299333 | 127465.477779 | 440164.918584 | 0 | 613182.811759 | 134560.123919 | 181473.207500 | 467378.483168 | 340174.999208 |
| 6 | 238588.412256 | 274590.512752 | 82881.621029 | 287292.053692 | 0 | 380595.134733 | 108753.923027 | 138817.520961 | 386597.690587 | 282384.024291 |
| 7 | 243044.063803 | 272567.262760 | 83032.663810 | 288606.689445 | 0 | 371793.971546 | 109542.056648 | 133826.527945 | 362264.029758 | 272742.481725 |
| 8 | 305244.032542 | 338353.840791 | 98654.928372 | 344583.210031 | 0 | 460647.375128 | 123510.833175 | 162758.915683 | 422450.808118 | 340174.999208 |
| 9 | 223635.133107 | 237470.444097 | 71178.816645 | 236751.393414 | 0 | 313667.929015 | 98467.088099 | 126048.812583 | 323032.890343 | 249791.414874 |
| 10 | 487133.466015 | 385528.193482 | 105974.375560 | 391861.347119 | 0 | 473434.567127 | 133703.019071 | 182221.385602 | 467303.900128 | 340174.999208 |

## Create forecast for furniture (repeat process with furniture data)

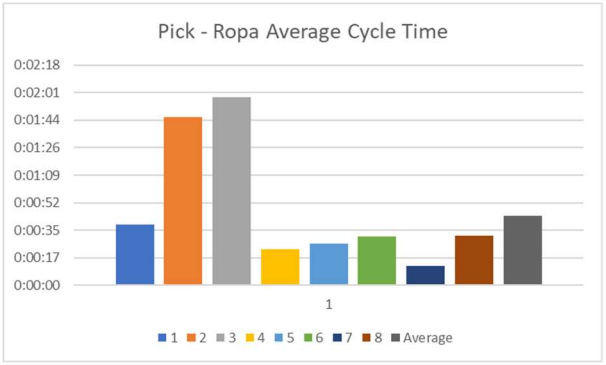
[450]:
```
df_allsales = df_furniture
```

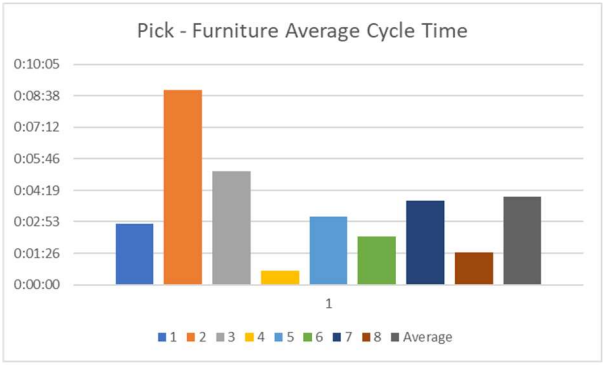
[451]:
```
df_allsales['Day'] = 1
df_allsales['Month'] = df_allsales['Mes']
df_allsales['Year'] = df_allsales['Anio']
```

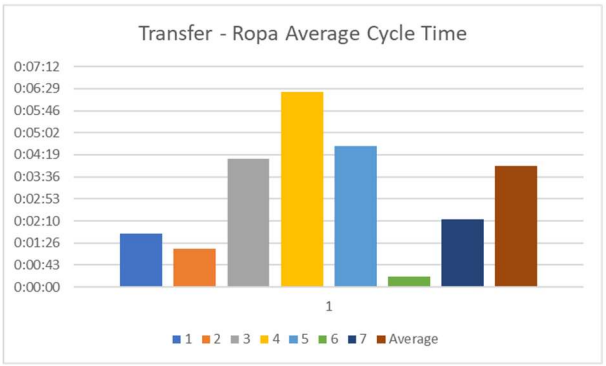
[452]:
```
df_allsales['Datetime'] = pd.to_datetime(df_allsales[['Year', 'Month', 'Day']])
```

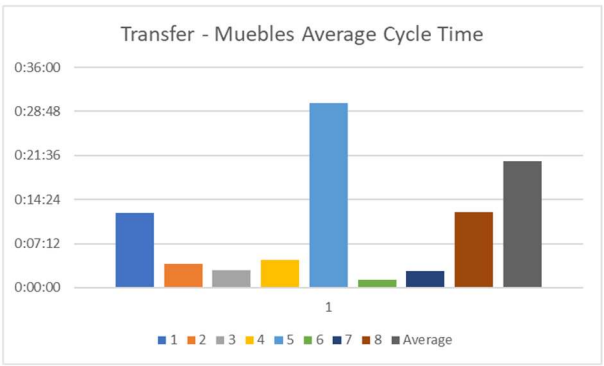
[453]:
```
df_allsales['Count']= df_allsales['Cantidad']
```

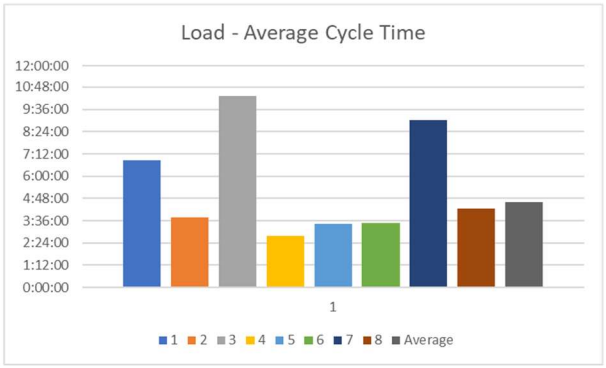
## ***E. Time Study Results***

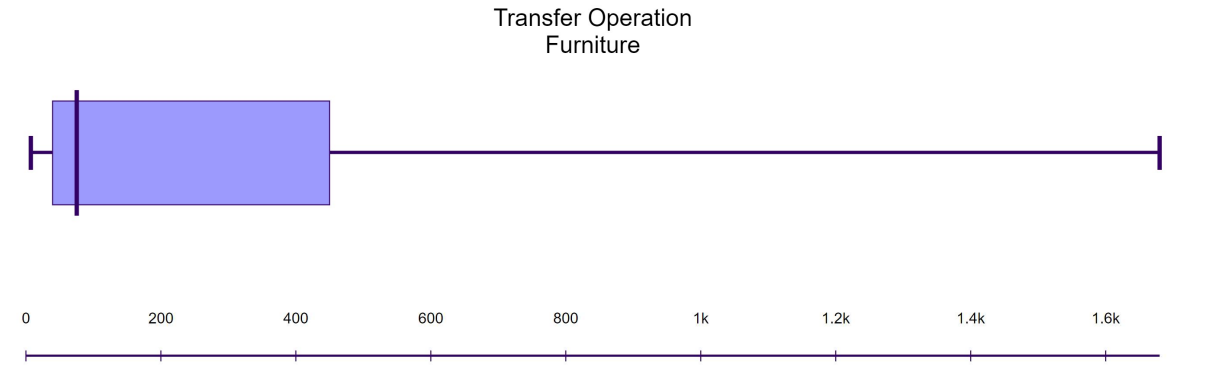


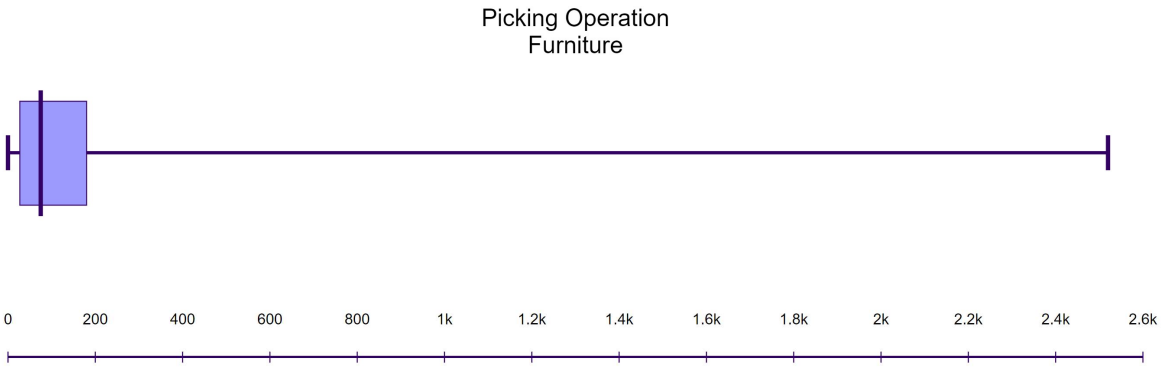



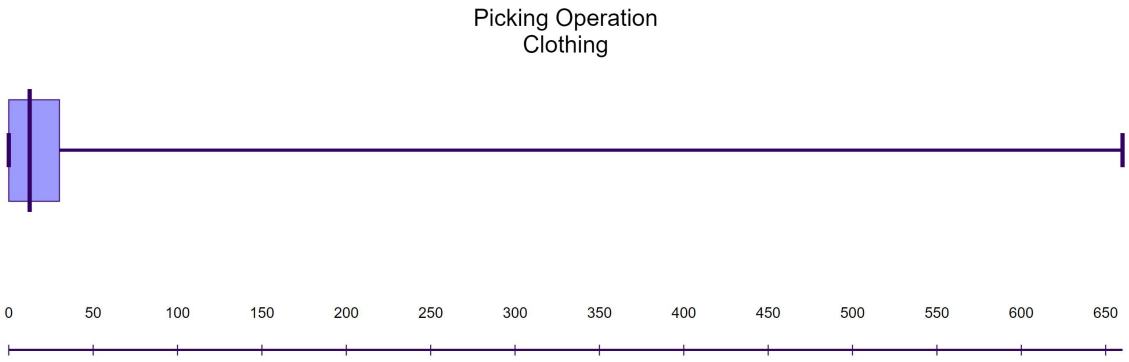




| Total Capacity - Clothing (Units & Workers) | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Distribution Center | W1 Headcount | W1 Capacity | W2 Headcount | W2 Capacity | W3 Headcount | W3 Capacity | W4 Headcount | W4 Capacity | Total Capacity |
| 1 | 107 | 1680873 | 119 | 459513 | 182 | 62899200 | 178 | 61516800 | 459513 |
| 2 | 127 | 1995055 | 93 | 359115 | 234 | 80870400 | 150 | 51840000 | 359115 |
| 6 | 168 | 2639127 | 129 | 498127 | 138 | 47692800 | 343 | 118540800 | 498127 |
| 7 | 72 | 1131055 | 55 | 212380 | 68 | 23500800 | 145 | 50112000 | 212380 |
| 8 | 93 | 1460945 | 78 | 301193 | 14 | 4838400 | 308 | 106444800 | 301193 |
| 11 | 61 | 958255 | 63 | 243272 | 107 | 36979200 | 76 | 26265600 | 243272 |
| 13 | 143 | 2246400 | 122 | 471097 | 242 | 83635200 | 91 | 31449600 | 471097 |
| 15 | 92 | 1445236 | 80 | 308916 | 66 | 22809600 | 310 | 107136000 | 308916 |
| 16 | 150 | 2356364 | 117 | 451790 | 108 | 37324800 | 242 | 83635200 | 451790 |
| 21 | 126 | 1979345 | 105 | 405453 | 105 | 36288000 | 181 | 62553600 | 405453 |
| 22 | 121 | 1900800 | 98 | 378422 | 114 | 39398400 | 132 | 45619200 | 378422 |
| 24 | 95 | 1492364 | 65 | 250994 | 95 | 32832000 | 95 | 32832000 | 250994 |
| 25 | 142 | 2230691 | 130 | 501989 | 190 | 65664000 | 135 | 46656000 | 501989 |
| 27 | 151 | 2372073 | 127 | 490404 | 128 | 44236800 | 128 | 44236800 | 490404 |

| Total Capacity - Furniture (Units & Workers) | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| Distribution Center | W1 Headcount | W1 Capacity | W2 Headcount | W2 Capacity | W3 Headcount | W3 Capacity | W4 Headcount | W4 Capacity | Total Capacity |
| 1 | 338 | 1451091 | 380 | 1080889 | 182 | 8985600 | 178 | 8788114 | 1080889 |
| 2 | 113 | 485128 | 119 | 338489 | 234 | 11552914 | 150 | 7405714 | 338489 |
| 6 | 69 | 296229 | 302 | 859022 | 138 | 6813257 | 343 | 16934400 | 296229 |
| 7 | 100 | 429317 | 105 | 298667 | 68 | 3357257 | 145 | 7158857 | 298667 |
| 8 | 61 | 261883 | 92 | 261689 | 14 | 691200 | 308 | 15206400 | 261689 |
| 11 | 176 | 755598 | 193 | 548978 | 107 | 5282743 | 76 | 3752229 | 548978 |
| 13 | 218 | 935911 | 238 | 676978 | 242 | 11947886 | 91 | 4492800 | 676978 |
| 15 | 131 | 562405 | 146 | 415289 | 66 | 3258514 | 310 | 15305143 | 415289 |
| 16 | 151 | 648268 | 193 | 548978 | 108 | 5332114 | 242 | 11947886 | 548978 |
| 21 | 43 | 184606 | 173 | 492089 | 105 | 5184000 | 181 | 8936229 | 184606 |
| 22 | 155 | 665441 | 161 | 457956 | 114 | 5628343 | 132 | 6517029 | 457956 |
| 24 | 118 | 506594 | 132 | 375467 | 95 | 4690286 | 95 | 4690286 | 375467 |
| 25 | 223 | 957376 | 234 | 665600 | 190 | 9380571 | 135 | 6665143 | 665600 |
| 27 | 161 | 691200 | 303 | 861867 | 128 | 6319543 | 128 | 6319543 | 691200 |