Testing

Professor Qichao Dong

CS 2400

5 May, 2024

Introduction

The Lopez Urban Farm is a Community Wellness Project by Community Four Partners in collaboration with Pomona Unified School District. Their mission is to grow the seeds of a sustainable and just future. They achieve this by ensuring access to locally grown food, educating the community, and creating green spaces open to all. The farm uses little technology which limits their efficiency. In this paper we present FarmMainClass as a solution to this problem. FarmMainClass is an application that organizes the farm's inventory and needed resources. The program then returns whether purchasing the items is suggested.

The remainder of this paper is organized as follows. Section 2 analyzes literature review that provides evidence as to why systems programming can be beneficial towards farms. In Sections 3 and 4 we discuss the methodology of the FarmMainClass and resulting data. Improvements of the FarmMainClass and a brief continued discussion of literature review is highlighted in Section 5. Conclusions and future work is outlined in Section 6.

Literature Reviews

Farming is one of America's major industries that contributes heavily towards the U.S economy and technology has been known to make jobs easier compared to how they were in the past. Systems programming can be a powerful ally for farmers through data-management programs allowing farmers to know when it is optimal for them to harvest, sell, and buy crops. There have been many instances in the past where systems programming has been beneficial towards farms and farmers. In an article written by Vienna McGrain titled “Agricultural data-tracking system helps farmers maximize output” it discusses a management software tool Jaime Sonneville, an apple farmer, built. This software tool is known as Agri-Trak and it is used to collect valuable data of the farm to help farmers make data-driven decisions. Furthermore, an article titled “Farm Management: Unleashing the Power of Agriculture Software" written by Meredith Mejia also touches on aspects in which systems programming can help farms. This article explores the importance of how farm management systems help farmers keep track of their land and their resources. These examples demonstrate how impactful systems programming, specifically dealing with resource management, can be in farming and agriculture. Using data management software can alleviate some of the burden farmers have to face when it comes to strategic decision-making which can allow them to focus on other aspects of their farm.

Methodology

The methodology used in this project revolves around the development and implementation of FarmMainClass with some implemented interfaces. At the core of FarmMainClass. Some data structures utilized include:

HashMap: A hash map (map, dictionary) data structure is used to store and manage resource information, including resource names, prices, current quantities, and quantities needed. This structure allows for O(1) time complexity when adding or removing items.

LinkedList: A linked list data structure is utilized to maintain a prioritized list of resources based on their importance. This list is dynamically updated as resource information changes, enabling the farm to focus on purchasing the most critical resources first.

Sorting and Prioritization Algorithm: A sorting algorithm is implemented to prioritize resources based on their importance, which is calculated using a combination of the farm's budget, resource prices, and quantities needed. This algorithm ensures that the most crucial resources are addressed first, optimizing resource allocation and maximizing the farm's efficiency.

A resizable-array-bag was also implemented to keep the list of all resource names only. In FarmMainClass, a command-line interface (CLI) was implemented that allows farm staff to interact with the application. The main functionalities of the application include:

Adding Resources: Users can add new resources to the system by providing the resource name, price, current quantity, and quantity needed.

Removing Resources: Existing resources can be removed from the system if they are no longer required or relevant.

Editing Resources: Users can modify the details of existing resources, such as price, current quantity, and quantity needed.

Printing Resource Information: The application can display a comprehensive list of all resources, including their names, prices, current quantities, and quantities needed.

Calculating Total Resource Costs: FarmMainClass calculates the total cost of acquiring the necessary quantities of each resource based on their prices and quantities needed.

Prioritizing Resources: The application generates a prioritized list of resources based on their importance, taking into account the farm's budget and resource requirements. This list suggests which resources should be purchased first and which can be deferred.

Budget Management: Users can input the farm's total budget, which is used by the application to calculate resource importance and prioritize resource acquisition accordingly.

To calculate the importance of certain items, the first step was calculating the total purchase cost for each resource. This was achieved by iterating through the resourceQuantityNeeded map, which contained the required quantities for each resource. For each resource, the current quantity (resourceQuantityCurrent) was subtracted from the needed quantity, and the resulting purchase quantity was multiplied by the resource price (resources) to obtain the total purchase cost. These values were then stored in the totalPurchaseCost map.

Next, the resources were sorted in descending order based on their total purchase cost. This was accomplished by creating a list of the totalPurchaseCost map entries and sorting them using the comparingByValue method with a reverse order comparator.

The algorithm then calculates the total cost that could be covered within the given budget. It iteratively went through the sorted resources, adding their total cost to the totalCostWithinBudget variable as long as the cost did not exceed the remaining budget. Finally, the importance of each resource was determined. For resources with a total cost less than or equal to the budget, the importance was calculated as the ratio of the resource's total cost to the total cost within the budget, multiplied by 100 to obtain a percentage. For resources with a total cost exceeding the budget, the importance was calculated as the ratio of the remaining budget to the resource's total cost, multiplied by 100.

The calculated resource importances were then stored in the resourceImportance map, and the updateResourcePriorityList() method was called to update the resource priority list accordingly. Although the values may vary, the suggestions for purchase are recognizable and consistent with input data. This methodology ensured simple implementation, yet an efficient approach to determining the importance of resources based on the given budget, in the main class.

A comparator was utilized to reverse the order of cost-ordered resources in a map, which allows change of the underlying object type, unlike a hashmap. It is faster than a stack-queue or double-stack configuration and safer to use. A HashMap was utilized to keep the total price of a resource with it's quantity requested, importance, or each individual value pair. InputMismatchException was utilized to check a few locations for bad user inputs. Lastly, the ArrayList and List implementation with LinkedList let us create the ordered list of resources. As mentioned, CLI was implemented using java.util.Scanner. Basic methods around these structures allow the user to edit or remove resources, by utilizing checks to ensure the proper functionality of the program without any errors. After each edit for example, the total price of resources are recalculated. If no resource is available, the code will not perform actions on empty or null data structures.

Results

FarmMainClass holds significant potential for optimizing inventory management and resource allocation at the Lopez Urban Farm. Through the utilization of a hash map-based algorithm, the application has the capacity to streamline operations and reduce manual labor associated with inventory counting and resource tracking.

One anticipated outcome of implementing FarmMainClass is the reduction in time spent on inventory management tasks. Prior to integrating the application into farm operations, staff dedicated substantial hours to manually counting and organizing inventory. However, with the introduction of FarmMainClass, the expectation is that this process will be expedited, allowing staff to allocate their time and efforts more efficiently.

Moreover, FarmMainClass is expected to provide valuable insights into resource utilization and replenishment. By analyzing data collected through the application, farm managers anticipate being able to identify patterns in resource consumption and accurately predict when restocking is required. This proactive approach to resource management is expected to prevent stockouts and optimize resource allocation, minimizing waste and maximizing efficiency.

Additionally, FarmMainClass is anticipated to facilitate better decision-making processes at the Lopez Urban Farm. With access to real-time data on inventory levels and resource availability, farm managers are expected to make more informed decisions regarding purchasing and resource allocation. This data-driven approach is anticipated to enhance the overall productivity and sustainability of the farm operation.

In summary, the potential benefits of implementing FarmMainClass at the Lopez Urban Farm include improved efficiency, reduced manual labor, and enhanced decision-making processes. Moving forward, further research and refinement of the application are essential to fully realize these anticipated benefits and contribute to the advancement of sustainable agriculture practices.

Discussions

Although the program alone can improve efficiency at the Lopez Urban Farm by cutting down on counting inventory and informing its workers when they should restock, given more time, it could be improved to be more user-friendly, better tailored to the farm, and to include additional capabilities. In its current state, the workers and volunteers of the farm would interact with the FarmMainClass through a command-line interface which can be difficult for anyone unfamiliar with programming as they would have to learn the basics of terminals and Java program compilation. After this initial hurdle, the program will be easy for the workers of the farm to run and use for their resource management, but a user interface (UI) would cut corners of any learning curves.

In this user interface, anyone operating FarmMainClass would view something similar to what they would see in the terminal once they run the program, several options which include adding a resource, printing the total price of the resources, etc., but could also instead run the application and perform necessary operations with a simple click of a button. Further price analysis programming can utilize low-demand and low prices of resources to factor in for calculating the importance of buying a particular resource. This is a math-heavy route but saves more money in the long term than the current implementation.

Another way in which to improve the program would be to allow users to divide and manage their budget based on the different priorities of the farm. For instance, one of the options presented to the user once they open the program could be to categorize the budget and from there they could add the resources into their respective category and be able to prioritize resources through the program just as they would with the original FarmMainClass. This implementation would be more effective than the current program as the Lopez Urban Farm already categorizes their budget and therefore makes for a better fit for the operations of the farm.

Compared to other data management programs mentioned in articles listed in the literature review, the FarmMainClass could further ease the work of those managing the farm by expanding to other areas such as financial management as discussed by Meredith Mejia in “Farm Management: Unleashing the Power of Agriculture Software.” In the article, the author introduces different types of farm management softwares, one of which is financial and accounting softwares which “monitor cash flow, store important documents, track their inventory levels, create invoices and handle other aspects of business finance” (Mejia, 2023). As the FarmMainClass would already aid the Lopez Urban Farm financially by informing users which resources to prioritize based on their budget, costs, and current stock, expanding its capabilities to those mentioned by Mejia could further lessen the work and time normally allocated towards accounting and allow the Lopez Urban Farm to prioritize their resources to other means which could improve production. However, the financial and accounting softwares which the author shares support that our program is already effective in helping the farm make financial decisions as it is similar to those she listed. Overall, although certain aspects could be improved, the FarmMainClass is an efficient program which will promote productivity on the farm.

Conclusions

This paper has presented the methodology and implementation of the FarmMainClass. A hash map is at the core of the FarmMainClass along with other interfaces such as LinkedList and a Sorting and Prioritizing Algorithm. With the utilization of these data structures FarmMainClass is able to store and manage inputted resource information. With this resource information the program can efficiently organize stock and provide information on what is necessary for the farm to purchase at the moment.

The potential benefits of implementing the FarmMainClass at the Lopez Urban Farm include a reduced amount of manual labor, enhanced decision-making processes, and most importantly improved efficiency due to a more organized inventory.

Moving forward, additional research of the program is essential. The FarmMainClass can be further developed in order to fully realize anticipated benefits such as resource tracking, resource management, and program driven decisions. The FarmMainClass is not exclusive to the Lopez Urban Farm and its impact can be distributed to a number of places around the world.