

CSC 210 Data Structures  
Final Exam

**Basic Instructions:**

---

1. In every file submitted you **MUST** place the following comments:
  - a. File Name.
  - b. Full name
2. Each student is required to submit the assignment on iLearn.
3. Please download the support files - if any - provided with this assignment and use them when implementing your project.
4. **A plagiarism detection tool will be used to check for code cheating among students. If a case of plagiarism is detected you will fail the exam and the course.**
5. Submission details:
  - a. The file name is very important and should follow the following format:  
**[YOUR\_ID]@Final.zip**
  - b. You should submit the assignment through iLearn: Submit the zip file.
6. **Failure to follow the above instructions will result in point deductions.**

## CSC 210: Final Exam

### Part One: Food bank system simulation

During the past weeks of this pandemic, food banks across the country are doing what they do best – feeding people in need within their communities. In this time of uncertainty, feeding those who are vulnerable to ensure that this pandemic does not perpetuate inequity or deepen the burdens often borne disproportionately by communities of color and low-income people.

This project is an opportunity to use what we have learned during this course to relate to building a food bank system. While in reality, food bank systems are mostly implemented as a web-based application or a mobile app, in this project we will try to build a desktop based application (Java application) of a specific food bank location. The simulation of food bank will cover basic entities such as HR and food supply.

You will be provided with basic guidelines that describes how the system works. ***If there is any detail that is not explicitly described then, it is left intentionally for your own judgment.***

#### Project Components:

1. Employee: There is two types of employees, full time employees and volunteers. The main difference between the two types is the way you calculate salaries. Full time employees salaries are calculated as follows :

Number of years \* 1.05 + basic salary

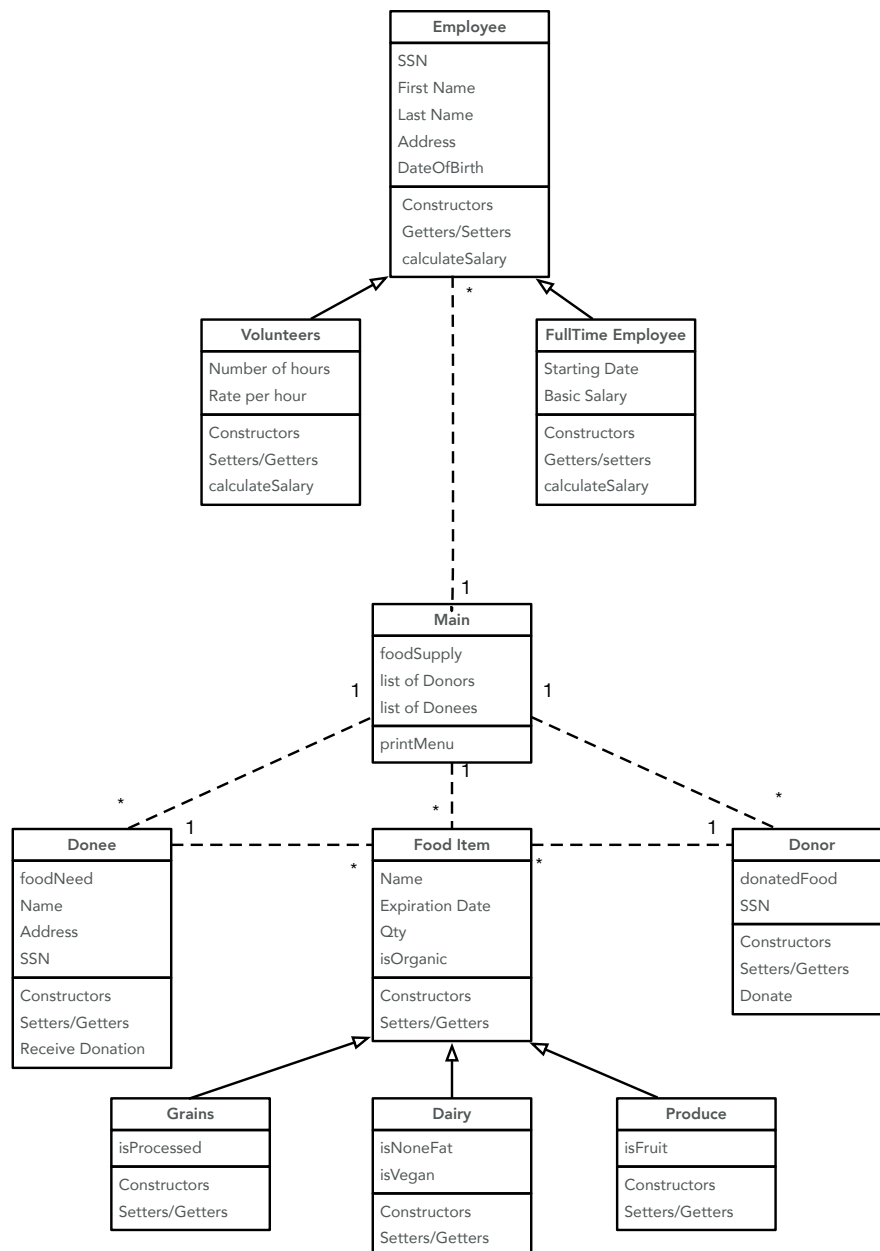
Where volunteers are not supposed to be paid but as an encouragement, the food bank pay them per hour as follows:

Number of hours \* rate per hour.

2. Food items: There are three types of food items, produce, dairy and grains.
3. Donor: This food bank asks donors to provide two information, social security number and food items they are willing to donate. *Hint: Use a list to hold food items.*
4. Donee: Each donee are asked to provide his/her name, address, social security number and food needs. *Hint: Use a list to hold food items.*
5. Main: This is the driver class (contains the main functions). This class contains:
  - a. list of donors: you can add new donors based on use choice.
  - b. list of donees: you can add new donors based on use choice.
  - c. current supply of food items (list): The supply will be increased whenever a donor **donates** food, and decreased whenever a donee **receives** his/ her food needs.
  - d. Print Menu to the use to choose which operation to execute.

You will be provided with a skeleton project (starter code). Download the zip file and continue building the project based on the requirements described. To give you a wholistic view of the project, follows a UML diagram of the classes hierarchy and relations.

## Project UML Class Diagram



### Program Run Sample:

When the user runs your program the following menu should appear:

```
[1] To add a full time employee information
[2] To add a volunteer information
[3] To add Donor information
[4] To add Donee information
[5] To receive donation from a specific Donor
[6] To send donation to a specific Donee
[7] To print employees information & salaries for this cycle
[8] To print food pantry current supply
[9] To Exit
Enter your choice:
```

[1] User will be used to provide information about the full time employee. Create the object and add it to the list of employees.

[2] User will be used to provide information about the volunteer. Create the object and add it to the list of employees.

[3] User will be used to provide information about the Donor. One of details you need to ask is the number of food items he/she is willing to donate and then ask for the each food item detailed information. When are you done, create the object and add it to the list of Donors.

[4] User will be used to provide information about the Donee. One of details you need to ask is the number of food items he/she needs to donate and then ask for the each food item detailed information. When are you done, create the object and add it to the list of Donees.

[5] Ask for the Donor SSN, retrieve the object from the list of Donors and then call the method: donate() which will add the food items to the food supply list.

[6] Ask for the Donee SSN, retrieve the object from the list of Donees and then call the method: receiveDonation() which will - based on the need - decrease the food items in the food supply list.

[7] Print information of all employees (full-time and volunteers), including the their salaries.

[8] Print food items detailed information the current food supply list.

[9] Exit

### **Part Two: Reflection Essay**

In this essay you will reflect in your experience working on this project. I am particularly interested in knowing the following:

- What design/ coding decisions you made while working on your project. There are decisions you had to take because it was not explicitly mentioned in the requirements of the project. Justify your decisions?
- Where did you encounter struggles? what did you do to deal with it?
- What did you learn ?
- Did you give your best effort? If not, why?
- What are things you did really well? What are things that you think can be improved ?
- Do you think remote-teaching affected your learning (negative or positive)? If so, how ?