CSCI 3901 Project High-level breakdown

Shubham Mishra (B00917146)

What comes into the program?

- Postal codes where each postal code contains:
    - postal code identifier
    - number of people living in the postal code
    - area covered by postal code
- Power distribution hubs where each hub contains:
    - Hub identifier
    - Coordinate location of hub in world in UTM coordinates
    - Set of postal codes served by the hub
- Hub damage information
    - Estimated hours needed to repair the hub identified by the Hub identifier, if there is any damage
- Hub repair information
    - Hub identifier of the repaired hub
    - Employee id of employee who did the repair
    - Repair time
    - If hub is fully repaired and ready to go

What transformation do I need to make?

- None

What part of data is processed right away?

- There are 3 types of methods:
    - **Data entry methods**: addPostalCode, addDistributionHub, hubDamage, hubRepair
    - **Reporting methods**: peopleOutOfService , mostDamagedPostalCodes , fixOrder, rateOfServiceRestoration, underservedPostalByPopulation, underservedPostalByArea
    - **Planning methods:** repairPan
- The data input to the reporting and planning methods is processed right away and the answer is returned back.

What part of data do I need to keep longer?

- The data input through data entry methods is needed to keep longer.
- New Postal code information is stored in database/data structure.
- New Power distribution hub information is stored in database/data structure.
- Hub damage information is stored in database/data structure.

- Hub repair information is logged into a file and database/data structure updated if needed.

What goes out of program?

- If the input postal code/hub information was recorded into the system.
- If the input hub information was recorded into the system.
- The number of people who are out of service.
- The list of “limit” postal codes that require most repairs.
- The list of “limit” most significant (measured by number of people who regain service per hour of repair) hubs to fix.
- The list of hours of repairs needed across whole system. List entry x reports the number of hours of repairs needed across the whole system before (x * increment) percent of province’s population has power.
- The list of hubs and their impacts that an employee should follow to do repairs to hubs given the start hub, the end hub, max time of repair allowed, and other conditions given in problem statement.
- The list of “limit” most underserved postal coded by population as well as by area in descending order of services needed.

What assumptions can I make?

- The input postal code identifier will not overlap with existing postal codes identifiers.
- The postal codes are stable. Once created, we won’t delete a postal code.
- Employee would report the repair via their cell phone before leaving power distribution hub.
- I can choose type of exception handling that would be appropriate as per circumstances.

What constraints exist?

- One hub can serve multiple postal codes and one postal code can be served by multiple hubs.
- All the methods of PowerService class that return some information, need to fetch the latest data stored in database/data structure for computation.
- The information managed by the PowerService class should survive between execution of programs that use PowerService class.

Are there strange cases to handle?

- Input validation for all the methods
- Boundary cases for all the methods
- The hub identifier passed in reporting and planning methods does not exist in database

What is important for solution to do?

- The solution should be able to store the information about postal codes and hubs even after finishing the program.
- It should accurately report the information as requested (as per output requirements).

Who are the users and how will they use it?

- The employees of the central power utility office for recording postal code and distribution hub information
- The field employees of the central power utility for reporting the hours of repair of a hub and if it is fully repaired before leaving power distribution hub.

What is the target environment?

- Should be portable across Windows, Mac OS, or Linux.
- Should be portable to at least Android on mobile phones as Android mobile applications also use Java. This program app would be required by field employees on their mobile.

How stable are the requirements?

- More reporting and planning methods could be added to PowerService class to extract other pieces of information or analysis of data in future.