COMP2034 – C++ PROGRAMMING

Mohammad Aazam

University of Nottingham (Malaysia)

CW2: TABLE OF CONTENTS

- **Problem:** Slide 3
- **Solution and CW Requirements:** Slide 4 – 8
- **Submission Guidelines:** Slide 9
- **Grading Policy:** Slide 10
- **Closing Remarks:** Slide 11

SMART WASTE MANAGEMENT

- PROBLEM: currently, waste management is not fully automatic, being collected mostly when its overdue or earlier, costing the waste collectors resources unnecessarily

- Impact of this problem:
  - If waste is not collected on time
    - Bacterial growth, insects, and other health hazards
    - Foul smell
    - Looks bad
  - If waste is collected before it should be
    - Consumes waste collector’s resources (such as fuel, time, wages) unnecessarily
    - The next visit might be ill-timed

SOLUTION

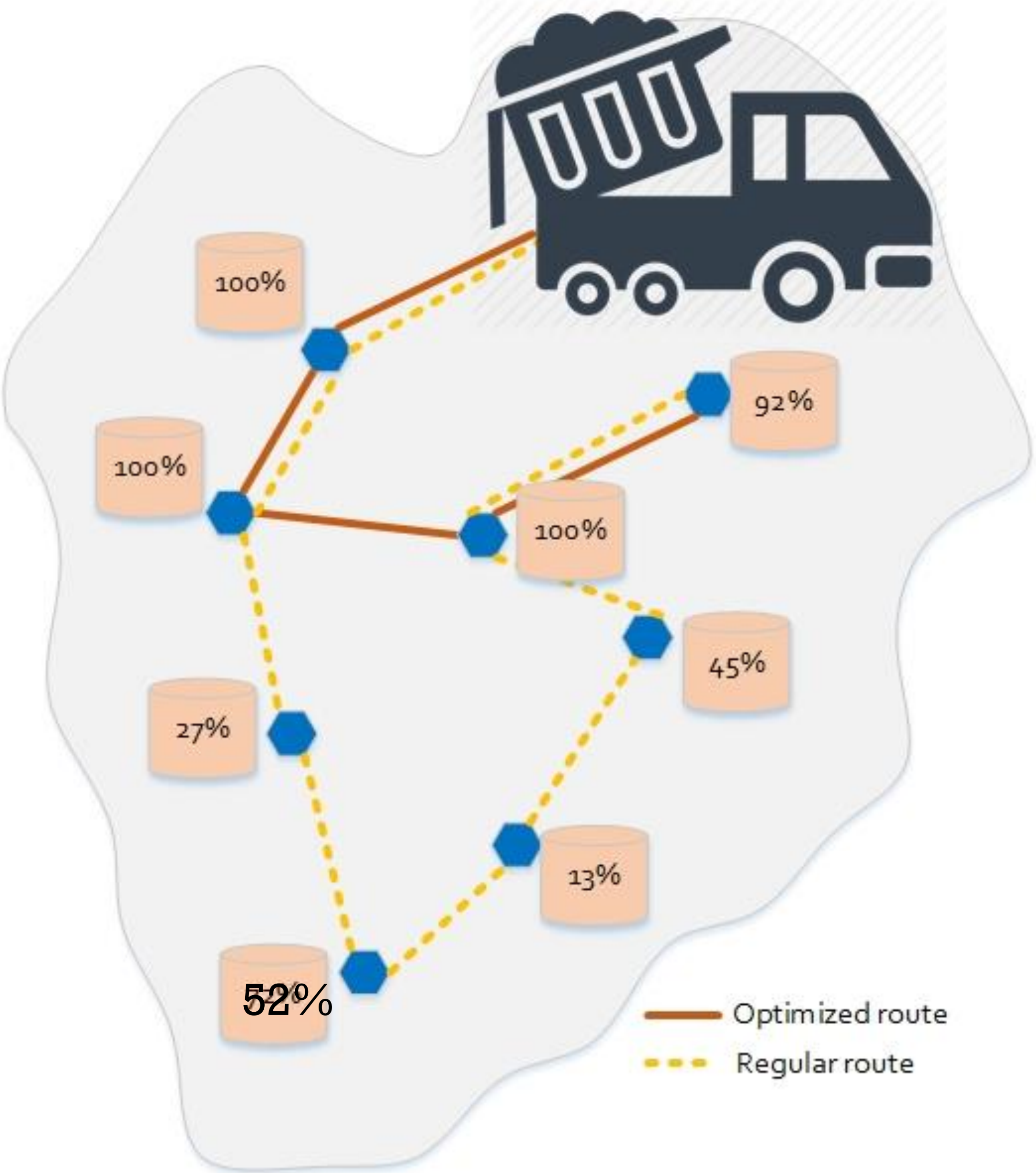
- Automatic waste status update
  - To create an automatic system that updates the waste management server about the waste statuses of bins in a city
  - The waste collectors can determine an optimized route based on the waste statuses and collect waste only and exactly when it is needed
- ASSUMPTIONS: We assume that each waste collection location has sensor-equipped waste bin that can transmit its waste status periodically (e.g., through WiFi) to the waste management server

SOLUTION DESCRIPTION

- Create a class named WasteLocations in which 7 or more locations are saved
- Assign a fixed, pre-defined distance to each location
  - We assume that this distance of each location is from the waste collector
- Every time your program is run, each waste location should get assigned with a random waste level between 0 – 100 (this shows waste level in %)
- If the waste level of a waste location is >= 40%, create a map of all those locations that waste collector has to visit
- Determine cost in terms of time, fuel consumption, and wages on the basis of the below data
  - Time: 1.5 minute per km; fuel consumption: 1.5 RM per km; wage: 5.77 RM per hour
  - Based on this info, create your algorithm that should make sense and determine correct results
  - This is your algorithm. Highlight it through comments and explain your logic (if necessary)
- Display the so far information (waste locations, distances, waste levels, all costs (and cumulative cost) as well as save this information in a file (e.g., .txt file, giving it name WL_InitialRoute)

SOLUTION DESCRIPTION

- Create another class OptimizedRoute
- In this class, determine if any waste level in WasteLocations class is < 60%, ignore that location. Now recalculate all the cost and display all the new information as well as the previous one. Save this data in a file named OptimizedRoute (can be a .txt file)
  - Also display how much cost is being saved through the optimized route
- User should be able to interact with your displayed information by approving the optimized route information (e.g, Press 'Y' to proceed with the optimized route, otherwise the original route will be followed)
- Perform inheritance wherever necessary in your program
- We need solution to the said problem. Do it using your C++ knowledge in however way you think necessary/productive/efficient.
  - Make use of inheritance, polymorphism, file I/O, exception handling, whatever you feel appropriate and applicable


100%
100%
92%
100%
45%
27%
13%
52%
Optimized route
Regular route

SUMMARIZING CW2 REQUIREMENTS

- Location database of different locations is in WasteLocations class
- Each waste location is assigned with a waste status through a random number generator
- A class (WasteLocations or any other in your program) determines a route based on >=40% waste
- A class OptimizedRoute determines optimized route based on >=60% waste
- In un-optimized as well as optimized route, costs are determined and properly saved in a file and information is displayed
- User can choose between optimized or un-optimized route and appropriate messages can be displayed

SUBMISSION

Fulfill all the requirements as mentioned on all Solution Description slides.

- Submit the following as one consolidated zipped file
  - 1. C++ source code (runnable and complete)
  - 2. Screenshots of all significant output (screenshots should be separate so that we can zoom in and see the info)
    - Make sure your output shows your own name wherever applicable and specified in the details
    - Make sure you write comments against each significant activity (statement, class, method, algorithm)
      - Not necessarily with every line, but each significant part
  - 3. Report (just a brief one, but no minimum length requirement) that contains:
    - How was your overall experience with this coursework? This includes any feedback you'd like to give
    - What significant difficulties you faced while doing this coursework, and why?
    - What new things you've learned and the potential benefits of this work?
    - References (Include, if applicable, any helpful sources. This is optional only if no helpful sources to cite.)
- Coursework 2 is due on Thursday 25th April 2022 (5 pm, Malaysia Time)
- Submission should be named as below:
  - *StudentName_StudentID_*CW2

GRADING

- Weightage: 20%
- Grading will be done based on the following:
  - **Accuracy and completeness of code:** 7 marks
  - **Output, user-friendliness and overall presentation:** 7 marks
  - **Code commenting and cleanliness:** 3 marks
  - **Report:** 3 marks

Enjoy the learning process.
Do not take stress or overburden yourselves.
Do take care of your health.

EXCITED?

- I hope this will be an interesting exercise, letting you apply your C++ programming knowledge to real-world scenario(s)
- You can think beyond this coursework and foresee how you can further extend such kind of a solution and what are its use-cases in the real-world
- This could be implemented in your locality, city, or this entire country
- This could even be a worldwide thing
- YOU CAN DO IT!
- All the best

Focus more on learning rather than grades.