BLG336E - Analysis of Algorithms II

Homework 3

Due Date: 16.05.2023, 23:30 PM

1 Implementation (70 Points)

You are the head of a theater club that wants to produce a movie. However, your current equipment is mostly old, and your possible contracts such as actors/actresses or the catering firm will be made with nonprofessionals. Therefore, you want to organize a tour to collect money for the shortcomings mentioned.

There are places where each one has different-sized salons and these salons have a daily schedule. You collected the availability intervals off the places and need to come up with a plan that maximizes the number of chairs.

After your tour ends, you spend the collected money on upgrading your equipment and/or contracts. Each upgrade will have a price and value that represent the positive effect on the movie (you can think of it as a technical quality increase or a morale boost for your team). You need to spend your money such that you maximize the value.

The input and output formats are given below. In short, you need to do;

1. Find the best possible schedule for each place
2. Find the best possible plan for your tour
3. Find the most valuable upgrades you can afford

For the homework, you should consider the Weighted Interval Scheduling and Knapsack problems to solve the given tasks.

2 Compile & Run

Implement your solution in **main.cpp** file. Your programs should compile using the following commands:

g++ -std=c++11 -Wall -Werror main.cpp -o main

Also, your programs should run using the following command respectively:

./main <**caseNo**> where caseNo is the example case ID.

2.1 Inputs

You are going to use four input files for each example. The contents of the input files will be as follows:

daily_schedule.txt

```
<str: PlaceName> <str: SalonName> <str: StartAt> <str: EndAt>
Example_Place_Name Example_Salon_Name 8:15 11:00
...
```

capacity.txt

```
<str: PlaceName> <str: SalonName> <int: Capacity>
Example_Place_Name Example_Salon_Name 80
...
```

availability_intervals.txt

```
<str: PlaceName> <str: StartingDate> <str: EndingDate>
Example_Place_Name 30.04 2.05
...
```

assets.txt

```
<str: Asset> <float: Price> <float: Value>
Example_Asset 123 1.2
...
```

NOTE: Each file has a header line.

The directory hierarchy will be as follows;

```
inputs
├── case_1
│   ├── daily_schedule.txt
│   ├── capacity.txt
│   ├── availability_intervals.txt
│   └── assets.txt
└── case_2
    └── ...
```

2.2 Outputs

For the outputs, you need to save the files according to the directory hierarchy below(similar to the input part);

```
outputs
├── case_1
│   ├── best_for_each_place.txt
│   ├── best_tour.txt
│   └── upgrade_list.txt
└── case_2
    └── ...
```

Rough formats for each output file are explained as follows;

best_for_each_place.txt

```
Example_Place_Name —> 123
Example_Place_Name Example_Salon_Name 8:15 11:00
Example_Place_Name Example_Salon_Name2 12.30 13:45
...

Example_Place_Name2 —> 456
Example_Place_Name2 Example_Salon_Name3 8:45 11:30
Example_Place_Name2 Example_Salon_Name4 12.30 13:15
...

...
```

You need to print the Weighted Interval Scheduling result for each place with the selected sessions printed below it.

best_tour.txt

```
Total Revenue —> 1234
Example_Place_Name 30.04 2.05
Example_Place_Name2 13.05 21.05
...
```

You need to print the Weighted Interval Scheduling result and the selected time intervals printed below (similar to the previous one).

upgrade_list.txt

```
Total Value —> 32
Example_Asset
Example_Asset2
...
```

You need to print the Knapsack result and the selected assets printed below.

3 Report (30 Points)

Please prepare a report and discuss the following items:

1. Explain your code and your solution.
   (a) Write your pseudo-code for each part. Try to keep your pseudo-code human-readable. Code snippets will not be graded.
   (b) Show the time complexity of your algorithms on the pseudo-code.
2. In the homework, you are given two restrictions;
   (a) You need to be in one place for the whole day (For example you can't be in place_A for 9:00-13:00 and place_B for 13:30-17:00 in one day.)
   (b) You can not break the determined available time intervals (For example you can't be in place_A for 15.04-17.04 and be in another place after 17.04 if place_A is available for the time period 15.04-19.04. You have to be in place_A for the full period.)

   How would the result be if you discard these constraints for the case_1? First, break the second restriction. Then, discard the first one also.

4 Grading Details

- There are 4 test scenarios in total, 2 of them are public test scenarios and 2 of them are hidden test scenarios. You can check your solutions with public cases during development.
- After the deadline, the hidden test scenarios also will be shared with you.
- You are required to write clean and readable code. A penalty of up to 10 points will be conducted in that case.

Important Notes:

- Please be aware of the deadline.
- Interactions among individuals are prohibited.
- It is prohibited to share or copy any code from your classmates or from the Internet. You have to submit your own, individual project.
- Please submit your homework through **only** Ninova.
- All your code must be written in C++, and should be compiled and run on the docker image shared under homework files(.devcontainer/Dockerfile) using g++. You can use the C++ Standard Template Library (STL). Your code must compile without any errors; otherwise, you may get a grade of zero on the assignment.
- When you write your code, try to follow an object-oriented methodology with well-chosen variables, methods, and class names and comments where necessary.
- In the report, you should show and explain how the algorithm works to find solutions to the given questions with meaningful explanations. Do not just tell the story please (for the first part).
- You should be aware that Ninova's clock may not be synchronized with your computer, watch, or cell phone. If you have submitted to Ninova once and want to make any changes, you should do it before the Ninova submission system closes. Your changes will not be accepted by e-mail.
- Send an email to **saritas21@itu.edu.tr** or **karatepe22@itu.edu.tr** for your questions.