Lab1 – Athlete Schedule
Deadline: 2022/11/10 23:55

Lab1 Objective

This lab is to exercise the concept of dynamic programming (DP).

Introduction

You are an athlete preparing for an important contest, and you have to manage your rest and practice schedule to get highest performance $\mathbf{P}$ in next couple of days.

You have $\mathbf{N}$ days to prepare for the contest. However, you only have the choice of either rest or practice in a single day.

For each day you practice, you will increase some constant number $\mathbf{A}$ to the number of performance $\mathbf{P}$.

However, without resting for $\mathbf{X}$ days in a row, including the day that you have decided to practice, in the end of the day the number of your performance $\mathbf{P}$ will be decreased by $\mathbf{X^2 * B}$, where $\mathbf{B}$ is some constant factor.

Additionally, if you have decided to rest in the $\mathbf{i}$'th day, you would start to think that your performance will get worse, and the number of performance $\mathbf{P}$ will be decreased by $\mathbf{R[i]}$.

Note that your performance could be negative in the end.

Input

Example (input.txt)

```
3 4 3          // N, A, B
7 0 4          // R[i]
```

Explanation of the Input

- First line represents the factors N, A, B.
- Second line, it will give N numbers representing the R[0], R[i], … , R[N-1].

Note: All the factors in input file are integer.

Output

Output Format (output.txt)

```
2              // P
1 0 1          // 0 for rest, 1 for practice
```

Explanation of the Output

- First line output maximum performance.
- Second line output the rest and practice schedule of each day.
- The performance originally is 0.
- If there are two or more ways to reach the maximum performance, select one of them will be correct.

Environment

1. Linux (Please make sure your code is available on our linux server. If it cannot be executed, you will get zero point!!)
2. Please use argc and argv to read input and output files or you will get fail in this lab.

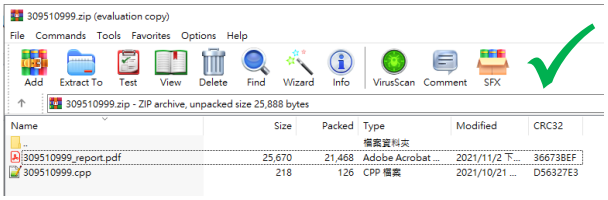
⚠Notice

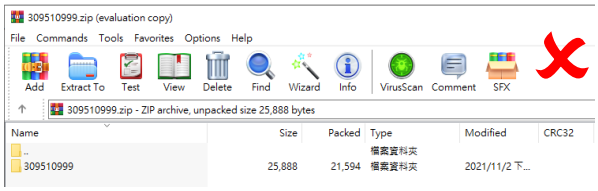
1. Do not print anything on the terminal! **(-5%)**
2. Please check the output format! TAs will use the command: *diff -b <file> <file>*
   If output format is not correct, you will get fail in this lab.

Submission

Please upload the following materials in a .zip file (e.g. **<student_ID>.zip**) to New E3 by the deadline, specifying your student ID in the subject field. (If your submission file is not .zip file, you will get zero point!!)

1. Source code (.cpp) (only one!!)
2. Report
   ⚠ Please check the .zip file with correct format as following figures. **(-5%)**





Evaluation

1. You **MUST WRITE YOUR OWN CODE**. Copying codes may get **FAIL** in this course.
2. For each case, it will be regarded as "failed" if you use more than time limit.
3. Naming rule.
   A. Compile procedure: **g++ -std=C++11 <student_ID>.cpp -o Lab1**
   B. Execution procedure: **./Lab1 [input] [output]**
      For example: *./Lab1 case1.txt output.txt*
   C. Source code: **<student_ID>.cpp**
      For example: *109511999.cpp*
   D. Report: **<student_ID>_report.pdf**
      For example: *109511999_report.pdf*
   E. .zip file: **<student_ID>.zip** (compress your source code and report)
      For example: *109511999.zip*

   ⚠ Naming Error: **-5%** per file

4. Grading (Time limit: 30 seconds for each case)
   A. Small case (x5) $[10^0 \leq N \leq 10^4]$ $[0 \leq A, B, R[i] \leq 10^4]$ 50%
   B. Big case (x3) $[10^4 \leq N \leq 10^6]$ $[0 \leq A, B, R[i] \leq 10^6]$ 30%
      - Correct answer 15%
      - Timing performance (if the answer is correct) 15%

C. Report 20%

- No more than 2 page
- Your report must contain:
  i. Time complexity analysis
  ii. The flow chart of you program

Due date

- Due date：2022/11/10 23:55
- **Penalty of 10%** of the total score per day for the first four days (weekend included) and will not be accepted afterwards