METU
Computer Engineering

Balloons Assignment

(Deadline: 21 April 2022, Thursday, 23:59)

CENG 316
Practice of Algorithms
Spring 2021-2022

1 Problem

We play a game where we are given a sequence of n balloons, placed neatly from left to right. Each balloon is assigned a fixed integer value, which is printed on the balloon. We are also given a needle to pop the balloons. When we pop a balloon, we earn a reward equal to the value of the popped balloon multiplied by the values of the two adjacent balloons. The popped balloon is removed from the sequence for good. To deal with the edge cases, we assume that the leftmost and rightmost balloons in the sequence are adjacent (as if the sequence was a circular sequence).

Write a program that finds the maximum total reward achievable in this game. Your program should also print the popping order that yields this reward. (If there are multiple orders, printing any is sufficient.) When specifying a balloon to be popped, your program should should mention its position (from the left) among the sequence of balloons. Notice that the positions of the balloons are subject to change as the balloons are popped and removed from the sequence.

2 Input Format

$\texttt{N}$
$\texttt{V}_1\ \texttt{V}_2\ \ldots\ \texttt{V}_N$

$\texttt{N}$ = The number of balloons in the initial sequence.
$\texttt{V}_i$ = The value of the ith balloon from the left in the initial sequence.

3 Output Format

$\texttt{R}$
$\texttt{P}_1\ \texttt{P}_2\ \ldots\ \texttt{P}_N$

$\texttt{R}$ = The maximum total reward achievable in the game.
$\texttt{P}_i$ = The position (as a 1-based index starting from the left) of the balloon to pop at the ith step.

4 Limits

$1 \leq \texttt{N} \leq 500$
$1 \leq \texttt{V}_i \leq 100$
Time Limit: 1 second
Memory Limit: 256 MB

5 Clarifications

- A solution with $O(N^3)$ time is expected to get full points in this assignment. However, you can get partial points for a slower solution. If you are unable to code the expected solution, please code a slower one.
- Your solution is expected as a C++ program source named `balloons.cpp` that reads from the standard input and writes to the standard output.
- It is OK to copy code from the sample codes we shared in our course website in ODTÜClass for this assignment.

- You are supposed to submit your code via ODTÜClass, via an auto-grader that we will prepare in the next few days.
- The grade from the auto-grader is not final. We can later do further evaluations of your code and adjust your grade. Solutions that do not attempt a “reasonable solution” to the given task may lose points.
- We will compile your code on g++ with options: `-std=c++17 -O2 -lm -Wall -Wextra -Wpedantic`
- Late submissions are not allowed.

6 Examples

Sample Input 1

```
6
7 4 8 5 6 3
```

Sample Output 1

```
1886
4 5 4 2 1 1
# Initial sequence:   7 4 8 5 6 3.
# 4th balloon popped. Brings a reward of 8 * 5 * 6 = 240.
# Follow-up sequence: 7 4 8 6 3.
# 5th balloon popped. Brings a reward of 6 * 3 * 7 = 126.
# Follow-up sequence: 7 4 8 6.
# 4th balloon popped. Brings a reward of 8 * 6 * 7 = 336.
# Follow-up sequence: 7 4 8.
# 2nd balloon popped. Brings a reward of 7 * 4 * 8 = 224.
# Follow-up sequence: 7 8.
# 1st balloon popped. Brings a reward of 8 * 7 * 8 = 448.
# Follow-up sequence: 8.
# 1st balloon popped. Brings a reward of 8 * 8 * 8 = 512.
# No balloons left.
```

Sample Input 2

```
10
6 9 2 3 1 7 6 5 7 3
```

Sample Output 2

```
3030
3 4 3 5 6 1 4 3 2 1
# Initial sequence:   6 9 2 3 1 7 6 5 7 3.
# 3rd balloon popped. Brings a reward of 9 * 2 * 3 = 54.
# Follow-up sequence: 6 9 3 1 7 6 5 7 3.
# 4th balloon popped. Brings a reward of 3 * 1 * 7 = 21.
# Follow-up sequence: 6 9 3 7 6 5 7 3.
# 3rd balloon popped. Brings a reward of 9 * 3 * 7 = 189.
# Follow-up sequence: 6 9 7 6 5 7 3.
# 5th balloon popped. Brings a reward of 6 * 5 * 7 = 210.
# Follow-up sequence: 6 9 7 6 7 3.
# 6th balloon popped. Brings a reward of 7 * 3 * 6 = 126.
# Follow-up sequence: 6 9 7 6 7.
# 1st balloon popped. Brings a reward of 7 * 6 * 9 = 378.
# Follow-up sequence: 9 7 6 7.
# 4th balloon popped. Brings a reward of 6 * 7 * 9 = 378.
# Follow-up sequence: 9 7 6.
# 3rd balloon popped. Brings a reward of 7 * 6 * 9 = 378.
# Follow-up sequence: 9 7.
# 2nd balloon popped. Brings a reward of 9 * 7 * 9 = 567.
# Follow-up sequence: 9.
# 1st balloon popped. Brings a reward of 9 * 9 * 9 = 729.
# No balloons left.
```