CS 341, Winter 2022 R. Peng, A. Storjohann

PROGRAMMING ASSIGNMENT 1

DUE: Wednesday, February 16, 11:59 PM. **No late submissions allowed.**
DO NOT COPY. ACKNOWLEDGE YOUR SOURCES.

Please read `http://www.student.cs.uwaterloo.ca/~cs341` for general instructions and policies.

Problem Description

Problem Statement

For this assignment, you are asked to implement an algorithm to solve Problem 2 of Assignment 3. Let us recall the problem:

There are n trails around the campus that you'd like to explore, each will bring you a happiness value of $h[i]$ (but only once, otherwise it gets repetitive).
The melting snow brings one major concern. Data compiled from `https://goose-watch.uwaterloo.ca/` suggest that trail i will become inaccessible $d[i]$ days from today ($1 \leq d[i] \leq n$).
Given that you want to explore at most one trail per day (without repetitions), compute the maximum happiness that can be achieved while avoiding geese encounters.

For the programming assignment, you also need to output a trail exploration schedule from day 1 (today) to day n that maximizes happiness.

Input Format

The first line of input consists of integer n, the number of trails. The second line of input consists of n integers $d[1], \ldots, d[n]$ ($1 \leq d[i] \leq n$), each separated by one space, where $d[i]$ indicates the last day on which trail i will be accessible. So, trail i is accessible only from day 1 to day $d[i]$. The third line of input consists of n integers $h[1], \ldots, h[n]$, each separated by one space, where $h[i]$ is the happiness value that exploring trail i brings.

Output Format

Output two lines. On the first line, output an integer, the greatest total happiness value attainable. On the second line, output an optimal exploration schedule as n space-separated integers. The i-th integer should either be the index of the trail (between 1 and n inclusive) that you want to explore on the i-th day, or -1 indicating that no trail is to be explored on that day.
If there are multiple schedules that achieve the greatest happiness value, output any of them.

Constraints

For all test cases, $1 \leq n \leq 100000, 1 \leq d[i] \leq n, 1 \leq h[i] \leq 10^9$.
Note that the constraints on $h[i]$ are slightly different from that in Assignment 3.
Hint 1: an $O(n^2)$ algorithm may not be efficient enough to pass all cases.
Hint 2: the answer may exceed 32-bit integer range.

Sample Input 1

```
4
3 1 2 4
1 11 10 20
```

Sample Output 1

```
42
2 3 1 4
```

Sample Input 2

```
4
1 1 1 1
1 11 10 20
```

Sample Output 2

```
20
4 -1 -1 -1
```

Sample Input 3

```
7
1 2 2 4 4 4 6
1 30 30 20 10 10 10
```

Sample Output 3

```
100
2 3 6 4 -1 7 -1
```

Submission Instructions

- Submit your solution on Marmoset.
- You can choose to code in either C++ or Python.
- Name your program `prog1.cpp/prog1.py`.
- **Time limit**: 2 seconds (C++) / 6 seconds (Python) for each test case.

- Compilation command for C++: `g++ -std=c++14 prog1.cpp -O3 -o prog1`.
- Execution command for Python: `python3 prog1.py`.
- Read from standard input and write to standard output.
- There will be several test cases, worth a total of 20 points. The public tests are worth 6 points and the secret tests are worth 14 points. The public tests (input only, not the answer) will be made available under a separate file.
- We will take the submission with the highest score. Please, however, refrain from excessive submissions.
- General collaboration policy applies. Please acknowledge your collaborator(s) by adding a comment in the beginning of your code.
- FAQ and updates will be posted on Piazza when necessary.