PROGRAMMING ASSIGNMENT 1

DUE: Wednesday, October 28, 5 PM. DO NOT COPY. ACKNOWLEDGE YOUR SOURCES.

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

1. **[20 marks] Dynamic programming for two suitcases.** In this programming assignment you will implement your algorithm for the Two Suitcases problem from Assignment 4, Question 2. Here is a reminder of the question:

    Suppose you are moving for a co-op job, and want to choose which of your belongings to take. You have two suitcases. Suitcase 1 will be weighed and there is a limit of W kilos. Suitcase 2 won't be weighed, but it has a volume limit of V. Suitcase 1 has no volume limit and suitcase 2 has no weight limit. You have n items, $1, 2, \ldots, n$. Item i has weight $w_i \in \mathbb{N}$, volume $v_i \in \mathbb{N}$, and benefit b_i if you take it with you. You can leave items behind.

    The goal is to find the maximum benefit by choosing items for the two suitcases while observing weight limit W for suitcase 1 and volume limit V for suitcase 2.

    In notation, you want sets $S_1, S_2 \subseteq \{1, 2, \ldots, n\}$ such that

    1. $\sum_{i \in S_1} w_i \leq W$,
    2. $\sum_{i \in S_2} v_i \leq V$, and
    3. $B = \sum_{i \in S_1 \cup S_2} b_i$ is maximized.

    Implement a dynamic programming algorithm to find the maximum benefit B **and** the items in each suitcase given by S_1 and S_2. You can write your code in C++ or Python.

    Your program will be tested to see if your output produces the maximum value for B and satisfies the constraints on S_1 and S_2 (note that the sets need not be unique, so we will not test for an exact match). We will also check how much time your program takes, to disqualify brute-force solutions.

    Input and Output. Your program should read from standard input and write to standard output. The input consists of four lines:

    1. 3 whitespace delimited positive integers which are n, W and V.
    2. n whitespace delimited positive integers $w(i)$ for $i = 1, 2, \ldots, n$.
    3. n whitespace delimited positive integers $v(i)$ for $i = 1, 2, \ldots, n$.
    4. n whitespace delimited positive integers $b(i)$ for $i = 1, 2, \ldots, n$.

    The output contains three lines:

    1. the maximum value B.
    2. a list of whitespace delimited numbers indicating the items in subset $S_1 \subseteq \{1, \ldots, n\}$.
    3. a list of whitespace delimited numbers indicating the items in subset $S_2 \subseteq \{1, \ldots, n\}$.

    You still need to print an empty line if $S_1 = \emptyset$ or $S_2 = \emptyset$.

Examples.

(a) For the following input:

```
2 10 40
8 7
20 56
3 11
```

the output is:

```
14
2
1
```

(b) For the following input:

```
5 5 5
2 6 5 6 2
6 5 4 5 1
4 6 7 4 2
```

the output is:

```
13
1
3
```

another valid output is:

```
13
3
2
```

Submission Guidelines.

- Programming assignments are submitted through Marmoset:
  https://marmoset.student.cs.uwaterloo.ca/

- Your program will be compiled / run in the student Linux environment using the command "g++ -std=c++14" for C++ and "python3" for Python.

- Submit one zip file containing only your code file which must be named "prog1.cpp" or "prog1.py".

- Your score will be the score of your best submission, which depends on a set of secret testcases. You will be able to see only small testcases as a sanity check for your code.

- Finally, you can expect the input to satisfy that
  - $n \leq 10^6$, $W \leq 10^3$, $V \leq 10^3$, $nWV \leq 10^8$, and
  - $w_i, v_i, b_i \leq 10^3$ for $i = 1, \ldots, n$.