

CS29002 Algorithms Laboratory

LAB TEST (ODD PC)

Date: 09–November–2016

You are given two (unsorted) arrays $A = (a_0, a_1, a_2, \dots, a_{n-1})$ and $B = (b_0, b_1, b_2, \dots, b_{n-1})$, each consisting of n integers (positive, negative, or zero). You are also given a target sum T (a positive integer). Your task is to find disjoint subsets I, J of indices satisfying $I \cup J = \{0, 1, 2, \dots, n-1\}$ such that

$$\sum_{i \in I} a_i + \sum_{j \in J} b_j = T,$$

or to report that no such I, J exist. You solve this problem using a dynamic-programming algorithm.

Part 1: Indicate what problems and subproblems are relevant in this context, and propose a recursive formulation of the problem in terms of the subproblems. Write the formula as a comment near the beginning of your program.

Part 2: Write a function `issolvable(A, B, n, T)` to find out whether an index set I satisfying the above property exists. Print the decision (*solution exists* or *solution does not exist*), and return 1 or 0 accordingly. You may pass additional parameters to the function, that would be needed in Part 3.

Part 3: Assume that the output of Part 2 is *solution exists*. Write a function `printsoln()` with appropriate parameters to print T as a sum of the chosen a_i and b_j values. Indicate clearly which summand comes from which of the two input arrays A and B . Present your output in a format shown in the sample output. The function should also compute and print the individual sums $sum_1 = \sum_{i \in I} a_i$ and $sum_2 = \sum_{j \in J} b_j$.

The `main()` function

- Read n , the elements of the arrays A and B , and the target sum T from the user.
 - Call `issolvable` to decide whether T can be expressed as a sum of suitably chosen a_i and b_j values.
 - If the output of Part 2 is *solution does not exist*, exit. Otherwise, call `printsoln` to print the chosen a_i and b_j values (and the individual sums).
-

Sample output

```
n = 10
A = 79 -89 -85 94 74 12 -84 70 -21 22
B = -87 -10 62 -33 -39 23 15 30 72 48
T = 123

+++ Solution exists

A : 79 -85 74 12 70 -21 22
B : -10 -33 15
Sum1 = 151, Sum2 = -28
```

Submit a single C/C++ source file. Do not use global/static variables.

Write your name, roll number, and PC number in a comment at the top of your program.

CS29002 Algorithms Laboratory

LAB TEST (EVEN PC)

Date: 09–November–2016

You are given two (unsorted) arrays $X = (x_0, x_1, x_2, \dots, x_{n-1})$ and $Y = (y_0, y_1, y_2, \dots, y_{n-1})$, each consisting of n positive integers. For each $i \in \{0, 1, 2, \dots, n-1\}$, you make one of the two choices: $z_i = x_i$ or $z_i = -y_i$. Your task is to find out whether for some choices, we have

$$z_0 + z_1 + z_2 + \dots + z_{n-1} = 0.$$

You solve this problem using a dynamic-programming algorithm.

Part 1: Indicate what problems and subproblems are relevant in this context, and propose a recursive formulation of the problem in terms of the subproblems. Write the formula as a comment near the beginning of your program.

Part 2: Write a function `choiceexists(X, Y, n)` in order to find out whether a choice of the z_i values satisfying the above property exists. Print the decision (*choice exists* or *choice does not exist*), and return 1 or 0 accordingly. You may pass additional parameters to the function, that would be needed in Part 3.

Part 3: Assume that the output of Part 2 is *choice exists*. Write a function `printchoice()` with appropriate parameters to print the z_i values chosen. Present your output in a format shown in the sample output. Your function should also compute and print the individual sums of the elements chosen from the two input arrays.

The `main()` function

- Read n , and the elements of the arrays A and B from the user.
- Call `choiceexists` to decide whether z_i values can be so chosen to have $z_0 + z_1 + z_2 + \dots + z_{n-1} = 0$.
- If the output of Part 2 is *choice does not exist*, exit. Otherwise, call `printchoice` to print the chosen z_i values (and the individual sums).

Sample output

```
n = 10
X = 34 31 26 11 58 39 83 62 81 95
Y = 53 72 43 27 46 6 32 55 56 37

+++ Solution exists

Z = -53 -72 26 -27 58 -6 -32 62 81 -37
Sum of elements chosen from X is 227
Sum of elements chosen from Y is 227
```

Submit a single C/C++ source file. Do not use global/static variables.

Write your name, roll number, and PC number in a comment at the top of your program.
