



ACM-ICPC AFGHANISTAN
ACM-ICPC First Training BootCamp
Hosted by Kateb University

Third Session

Location: Kateb University

Date: 26/7/2018

Time: 01:30 pm

1) Sum of N elements

You have giving a list of space separated integers, your job is to write a program to print the sum of these integers.

- **Constraints:**

Use recursive technique to solve this problem.

$$0 \leq N_i \leq 10^3$$

Input format:

Input will be one line that contains the n separated integers.

Output format:

Print the sum of the integers.

Input: 1 2 3 4 5

Output: 15

2) Sum of N elements

You have giving a list of space separated integers, your job is to write a program to print the sum of these integers.

- **Constraints:**

You are not allowed to use Array, LinkedList or Vector

$$0 \leq N_i \leq 10^3$$

Input: 1 2 3 4 5

Output: 15

3) Union, Intersection and Difference

Union

In this, we could simply add one Set with other. Since the Set will itself not allow any duplicate entries, we need not take care of the common values.

Intersection

We just need to retain the common values from both Sets.

Difference

We just need to remove all the values of one Set from the other.

Input format:

The first line is the number of test case, the next line is the value of set1 and the third line is the value of set2.

Output format:

Print the Test case number like Test case #number_of_test_case: then print the union, intersection and difference of two sets.

Input:

1

1,3,2,4,8,9,0

1,3,7,5,4,0,7,5

Output:

Test case #1:

Union of the two Set is: {0,1,2,3,4,5,7,8,9}

Intersection of the two Set: {0,1,3,4}

Difference of the two Set: {2,8,9}

4) Thief Man

A thief man is going in a store and want to steal the store's stuff but the thief man has a bag with limits weight that can carry the stuffs, help the thief man to select the stuffs to reach the maximum benefit.

Note: each stuff has the value and weight, thief man can take a piece of stuff.

Input format:

The first line is the number of test cases, after that we have three lines, the first one is the values of the stuff's price, the second line is the values of stuff's weight and the last

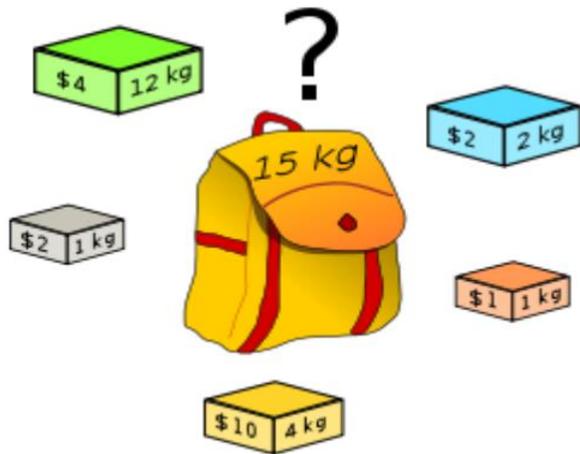
Output format:

Output the maximum benefit and the selected item with its weight. Like below:

Maximum benefit: 200

Stuff1: 100

stuff2: 5



input:

1

4,2,10,1,2

12,1,4,1,2

output:

17,33

Item1: 7

Item2: 1

Item3: 4

Item4: 1

Item5: 2

5) Merging and Sorting

Work with stack, merge and sort two stacks (stack1, stack2) into a new final stack(stack3).

Input Format:

First line of the input will contain the N space separated integers (first stack elements) and the second line of the input contain the M space separated integers (second stack elements).

Output format:

Print the new final stack(stack3) that is sorted and made by merging stack1 and stack2

Input:

Stack1: 9 4 2 1

Stack2: 8 17 3 10

Output:

Stack3: 1 2 3 4 8 9 10 17

6) Word Count

A teacher gives homework to their students, student should write paragraph about something that contains different words, write a program to check all the different words in a paragraph.

Input format:

The first line is the amount of test case, the next line is the amount of words and the last line is paragraph.

Output format:

Output if the student completed the amount word print true and the amount of word that student wrote, if student not completed the amount of word print false and the amount of word that student wrote.

Input:

1

8

Early to bed early to rise makes a man healthy wealthy wise

Output:

True 11

7) Check if the queue can be sorted

Write a program to check if a queue can be sorted into another queue. You are allowed to use only stack and queue data structures for this operation.

Input format:

The input will have one line that will contain the N space separated integers (first queue elements).

Output format:

Print True if the queue can be sorted into another queue using stack, otherwise print False.

Input: 5 1 2 3 4

Output: True

Input: 5 1 2 6 3 4

Output: False

8) Change the infix expressions to postfix expressions

Infix expression:

The expression of the form a op b. When an operator is in-between every pair of operands.

Postfix expression:

The expression of the form a b op. When an operator is followed for every pair of operands.

Input format:

The first line is the t number of test case. The next t lines are the expressions.

Output format:

Output the postfix expression of input infix.

Input:

2

abc++

ab*c+

Output:

(a+(b+c))

((a*b)+c)

