

Started on	Monday, 26 February 2024, 4:47 PM
State	Finished
Completed on	Monday, 26 February 2024, 6:02 PM
Time taken	1 hour 15 mins
Marks	20.00/20.00
Grade	10.00 out of 10.00 (100%)

Question 1

Correct

Mark 10.00 out of 10.00

This challenge is part of a tutorial track by [MyCodeSchool](#) and is accompanied by a video lesson.

Given a pointer to the head of a singly-linked list, print each **data** value from the reversed list. If the given list is empty, do not print anything.

Example

head* refers to the linked list with **data** values **1 → 2 → 3 → NULL**

Print the following:

3

2

1

Function Description

Complete the `reversePrint` function in the editor below.

`reversePrint` has the following parameters:

- `SinglyLinkedListNode` pointer `head`: a reference to the head of the list

Prints

The **data** values of each node in the reversed list.

Input Format

The first line of input contains ***t***, the number of test cases.

The input of each test case is as follows:

- The first line contains an integer ***n***, the number of elements in the list.
- Each of the next ***n*** lines contains a data element for a list node.

Constraints

- $1 \leq n \leq 1000$
- $1 \leq \text{list}[i] \leq 1000$, where ***list[i]*** is the ***i*th** element in the list.

Sample Input

```
3
5
16
12
4
2
5
3
7
3
9
5
5
1
18
3
13
```

Sample Output

```
5
2
4
12
16
9
3
7
13
3
18
1
5
```

Explanation

There are three test cases. There are no blank lines between test case output.

The first linked list has **5** elements: **16** → **12** → **4** → **2** → **5**. Printing this in reverse order produces:

5
2
4
12
16

The second linked list has **3** elements: **7** → **3** → **9** → **NULL**. Printing this in reverse order produces:

9
3
7

The third linked list has **5** elements: **5** → **1** → **18** → **3** → **13** → **NULL**. Printing this in reverse order produces:

13
3
18
1
5

For example:

Input	Result
3	5
5	2
16	4
12	12
4	16
2	9
5	3
3	7
7	13
3	3
9	18
5	1
5	5
1	
18	
3	
13	
3	17
3	1
11	11
1	15
17	11
3	12
12	14
11	15
15	7
4	5
5	
7	
15	
14	

Answer: (penalty regime: 0 %)

Reset answer

```
1 #include <bits/stdc++.h>
2
3 using namespace std;
4
5 class SinglyLinkedListNode {
6     public:
7         int data;
8         SinglyLinkedListNode *next;
9
10     SinglyLinkedListNode(int node_data) {
11         this->data = node_data;
12         this->next = nullptr;
13     }
14 };
15
16 class SinglyLinkedList {
17     public:
18         SinglyLinkedListNode *head;
19         SinglyLinkedListNode *tail;
20
21     SinglyLinkedList() {
```

```
21  singlyLinkedList() {
22      this->head = nullptr;
23      this->tail = nullptr;
24  }
25
26  void insert_node(int node_data) {
27      SinglyLinkedListNode* node = new SinglyLinkedListNode(node_data);
28
29      if (!this->head) {
30          this->head = node;
31      } else {
32          this->tail->next = node;
33      }
34
35      this->tail = node;
36  }
37  };
38
39  void print_singly_linked_list(SinglyLinkedListNode* node, string sep) {
40      while (node) {
41          cout << node->data;
42
43          node = node->next;
44
45          if (node) {
46              cout << sep;
47          }
48      }
49  }
50
51  void free_singly_linked_list(SinglyLinkedListNode* node) {
52      while (node) {
```

	Input	Expected	Got	
✓	3	5	5	✓
	5	2	2	
	16	4	4	
	12	12	12	
	4	16	16	
	2	9	9	
	5	3	3	
	3	7	7	
	7	13	13	
	3	3	3	
	9	18	18	
	5	1	1	
	5	5	5	
	1			
	18			
	3			
	13			
✓	3	17	17	✓
	3	1	1	
	11	11	11	
	1	15	15	
	17	11	11	
	3	12	12	
	12	14	14	
	11	15	15	
	15	7	7	
	4	5	5	
	5			
	7			
	15			
	14			

Passed all tests! ✓

Correct

Marks for this submission: 10.00/10.00.

Question 2

Correct

Mark 10.00 out of 10.00

Alexa has two stacks of non-negative integers, stack $a[n]$ and stack $b[m]$ where index 0 denotes the top of the stack. Alexa challenges Nick to play the following game:

- In each move, Nick can remove one integer from the top of either stack a or stack b .
- Nick keeps a running sum of the integers he removes from the two stacks.
- Nick is disqualified from the game if, at any point, his running sum becomes greater than some integer $maxSum$ given at the beginning of the game.
- Nick's *final* score is the total number of integers he has removed from the two stacks.

Given a , b , and $maxSum$ for g games, find the maximum possible score Nick can achieve.

Example
 $a = [1, 2, 3, 4, 5]$
 $b = [6, 7, 8, 9]$

The maximum number of values Nick can remove is **4**. There are two sets of choices with this result.

1. Remove **1, 2, 3, 4** from a with a sum of **10**.
2. Remove **1, 2, 3** from a and **6** from b with a sum of **12**.

Function Description
Complete the `twoStacks` function in the editor below.

`twoStacks` has the following parameters: - `int maxSum`: the maximum allowed sum
- `int a[n]`: the first stack
- `int b[m]`: the second stack

Returns
- `int`: the maximum number of selections Nick can make

Input Format
The first line contains an integer, g (the number of games). The $3 \cdot g$ subsequent lines describe each game in the following format:

1. The first line contains three space-separated integers describing the respective values of n (the number of integers in stack a), m (the number of integers in stack b), and $maxSum$ (the number that the sum of the integers removed from the two stacks cannot exceed).
2. The second line contains n space-separated integers, the respective values of $a[i]$.
3. The third line contains m space-separated integers, the respective values of $b[i]$.

Constraints

- $1 \leq g \leq 50$
- $1 \leq n, m \leq 10^5$
- $0 \leq a[i], b[i] \leq 10^6$
- $1 \leq maxSum \leq 10^9$

Subtasks

- $1 \leq n, m, \leq 100$ for 50% of the maximum score.

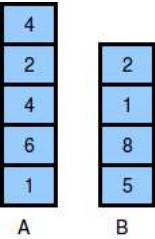
Sample Input 0

1
5 4 10
4 2 4 6 1
2 1 8 5

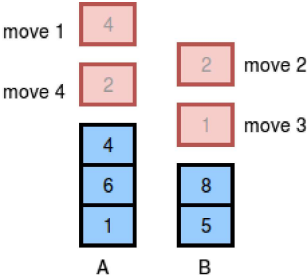
Sample Output 0

4

Explanation 0
The two stacks initially look like this:



The image below depicts the integers Nick should choose to remove from the stacks. We print **4** as our answer, because that is the maximum number of integers that can be removed from the two stacks without the sum exceeding $x = 10$.



(There can be multiple ways to remove the integers from the stack, the image shows just one of them.)

For example:

Input	Result
1 5 4 10 4 2 4 6 1 2 1 8 5	4
3 7 2 668 12 54 75 66 99 22 66 93 32 3 10 541 34 60 55 47 68 67 23 18 99 24 39 56 12 5 7 580 29 21 75 81 73 42 32 49 22 48 91 67	9 11 11

Answer: (penalty regime: 0 %)

Reset answer

```
1 #include <bits/stdc++.h>
2
3 using namespace std;
4
5 string ltrim(const string &);
6 string rtrim(const string &);
7 vector<string> split(const string &);
8
9 /*
10  * Complete the 'twoStacks' function below.
11  *
12  * The function is expected to return an INTEGER.
13  * The function accepts following parameters:
14  * 1. INTEGER maxSum
15  * 2. INTEGER_ARRAY a
16  * 3. INTEGER_ARRAY b
17  */
18
19 int twoStacks(int x, vector<int> a, vector<int> b) {
20     int sum = 0, i=0, j=0, ii=a.size(), jj=b.size() , cnt=0;
21     while(i<ii && sum+a[i]<=x)
22         sum+=a[i++];
23     cnt = i;
24     while(i>=0 && j<jj)
25     {
26         sum+=b[j++];
27         while(sum>x && i>0)
28             sum-=a[--i];
29         if(sum<=x && (i+j)>cnt)
30             cnt = i+j;
31     }
32     return cnt;
33 }
34
35 int main()
36 {
37
38     string g_temp;
39     getline(cin, g_temp);
40
41     int g = stoi(ltrim(rtrim(g_temp)));
42
43     for (int g_itr = 0; g_itr < g; g_itr++) {
```

```
44     string first_multiple_input_temp;
45     getline(cin, first_multiple_input_temp);
46
47     vector<string> first_multiple_input = split(rtrim(first_multiple_input_temp));
48
49     int n = stoi(first_multiple_input[0]);
50
51     int m = stoi(first_multiple_input[1]);
52
```

	Input	Expected	Got	
✓	1 5 4 10 4 2 4 6 1 2 1 8 5	4	4	✓
✓	3 7 2 668 12 54 75 66 99 22 66 93 32 3 10 541 34 60 55 47 68 67 23 18 99 24 39 56 12 5 7 580 29 21 75 81 73 42 32 49 22 48 91 67	9 11 11	9 11 11	✓

Passed all tests! ✓

Correct

Marks for this submission: 10.00/10.00.