

Programming Challenge Description:

Bob is developing a new strategy to get rich in the stock market. He wishes to invest his portfolio for 1 or more days, then sell it at the right time to maximize his earnings. Bob has painstakingly tracked how much his portfolio would have gained or lost for each of the last N days. Now he has hired you to figure out what would have been the largest total gain his portfolio could have achieved.

Example: Bob kept track of the last 10 days in the stock market. On each day, the gains/losses are as follows: 7 -3 -10 4 2 8 -2 4 -5 -2. If Bob entered the stock market on day 4 and exited on day 8, his gains would have been 16 (4 + 2 + 8 + -2 + 4).

Input:

The input consists of integers on a line separated by spaces. The input contains N , the number of days ($0 < N < 10000$), followed by N integers D ($-10000 < D < 10000$) indicating the gain or loss on that day.

Output:

For each test case, print a line containing the maximum possible gain over the period. If no gain is possible, print 0.

Test 1**Test Input** 

```
10 7 -3 -10 4 2 8 -2 4 -5 -6
```

Expected Output 

```
16
```

Bob has just won a shopping spree at his favorite store, Acme Electronics. Acme has provided Bob with a shopping cart that can hold L pounds of items. Bob wants to maximize the total value of items he can place into his shopping cart, without exceeding the weight limit. He can take no more than one of each available item.

For example, Bob has a shopping cart that can hold 10 pounds of items, and there are 4 items in the store:

```
5 4
3 2
10 8
4 8
```

Bob can maximize the value of his cart by selecting one item of value 10 (weight 8) and one of value 3 (weight 2) for a total of 13.

Input:

The input consists of a line containing two integers L and N, separated by a space. L is the maximum weight of items that he may place into his shopping cart, and N is the number of types of items in the store. Then N lines follow, each containing two integers P and W separated by a space. P contains the price of the item, and W contains the weight of the item. $1 \leq L, N, P, W \leq 1000$.

Output:

Print a line containing the maximum total price of all items that Bob can fit into his cart.

Test 1

Test Input 

```
10 4
```

Question 1

☰ Challenge

>_ Test Case Output

Programming Challenge Description:

Assume that someone dictates you a sequence of numbers and you need to write it down. For brevity, he dictates it as follows: first says the number of consecutive identical numbers and then says the number itself. E.g. The sequence 1 1 3 3 3 2 2 2 2 14 14 14 11 11 11 2 will be dictated as "Two ones, three threes, four twos, three fourteens, three elevens, one two", so you will write down the sequence 2 1 3 3 4 2 3 14 3 11 1 2. The challenge is to write the program which compresses the given sequence using this approach.

Input:

Your program should read lines from standard input. Each line is a sequence of L integers, where each integer is N, separated by a whitespace. N is in range [0, 99]. L is in range [1, 400].

Output:

For each test case, produce a single line of output containing a compressed sequence of numbers separated by a single space char.

Test 1

Test Input

```
40 40 40 40 29 29 29 29 29 29 29 57 57 92 92 92 92
92 86 86 86 86 86 86 86 86 86
```

Expected Output

```
4 40 8 29 2 57 5 92 10 86
```

Test 2

Test Input

```
7
```

☰ Challenge

>_ Test Case Output

Programming Challenge Description:

Write a program to determine the largest sum of contiguous integers in a sequence. Contiguous means that the integers are adjacent to each other. Take -4, 2, -5, 0, 3 as an example. For this example the largest sum would be 3 and was obtained from the subsequence 0, 3.

Input:

Your program should read lines of text from standard input. Each line will contain a sequence of comma-separated integers.

Output:

For each line of input, print to standard output the sum of the largest contiguous subsequence in each sequence, one integer per line. In other words, of all the possible contiguous subsequences for a given set, find the one with the largest sum, and print that sum.

Test 1

Test Input

```
-10, 2, 3, -2, 0, 5, -15
```

Expected Output

```
8
```

Test 2

Test Input

```
2,3,-2,-1,10
```

Expected Output

```
12
```

▶ 运行测试案例

提交最终答案

☰ 挑战

>_ 测试案例输出

编程挑战说明:

Write a program that finds the greatest difference between integers in a list. The list can contain positive and negative integers.

输入:

Your program should read lines from standard input. Each line contains a comma-separated list of integers.

输出:

For each input list, print to standard output the greatest difference between any two integers in the list. Print each difference on its own line.

Test 1

测试输入

```
1,2,10,0,3,9
```

预期输出

```
10
```

Test 2

测试输入

```
4,-9,-3,0,7,9
```

预期输出

```
18
```

挑战

>_ 测试案例输出

编程挑战说明:

You are given 3 coins of value 1, 3 and 5. You are also given a total which you have to arrive at. Find the minimum number of coins to arrive at it.

输入:

Your program should read lines from standard input. Each line contains a positive integer number which represents the total you have to arrive at.

输出:

Print out the minimum number of coins required to arrive at the total

Test 1

测试输入

11

预期输出

3

Test 2

测试输入

20

预期输出

4

Programming Challenge Description:

When John was a little kid he didn't have much to do. There was no internet, no Facebook, and no programs to hack on. So he did the only thing he could... he evaluated the beauty of strings in a quest to discover the most beautiful string in the world.

Given a string s , little Johnny defined the beauty of the string as the sum of the beauty of the letters in it. The beauty of each letter is an integer between 1 and 26, inclusive, and no two letters have the same beauty. Johnny doesn't care about whether letters are uppercase or lowercase, so that doesn't affect the beauty of a letter. (Uppercase 'F' is exactly as beautiful as lowercase 'f', for example.)



You're a student writing a report on the youth of this famous hacker. You found the string that Johnny considered most beautiful. What is the maximum possible beauty of this string?

Input:

Your program should read lines from standard input. Each line is a string.

Output:

Print out the maximum beauty for the string.

☰ Challenge

>_ Test Case Output

Programming Challenge Description:

Write a program to determine the Mth to last element of a sequence.

Input:

Your program should read lines of text from standard input. Each line will contain a series of space delimited integers. The last integer in the series is M.

Output:

Print to standard output, the Mth integer from the end of the sequence, one integer per line. If M is larger than the sequence size, print a blank line.

Test 1

Test Input

```
a b c d 4
```

Expected Output

```
a
```

Test 2

Programming Challenge Description:

When John was a little kid he didn't have much to do. There was no internet, no Facebook, and no programs to hack on. So he did the only thing he could... he evaluated the beauty of strings in a quest to discover the most beautiful string in the world.

Given a string s , little Johnny defined the beauty of the string as the sum of the beauty of the letters in it. The beauty of each letter is an integer between 1 and 26, inclusive, and no two letters have the same beauty. Johnny doesn't care about whether letters are uppercase or lowercase, so that doesn't affect the beauty of a letter. (Uppercase 'F' is exactly as beautiful as lowercase 'f', for example.)



You're a student writing a report on the youth of this famous hacker. You found the string that Johnny considered most beautiful. What is the maximum possible beauty of this string?

Input:

Your program should read lines from standard input. Each line is a string.

Output:

Print out the maximum beauty for the string.

编程挑战说明:

Credits: *Programming Challenges* by Steven S. Skiena and Miguel A. Revilla

The problem is as follows: choose a number, reverse its digits and add it to the original. If the sum contains odd digits, repeat this procedure. eg.

168 (initial number) + 681 (reverse of initial number) = 1029

1029 + 9201 = 10230

10230 + 03201 = 13431

13431 + 13431 = 26862 (no odd digits)

In this particular case the first result with no odd digits (26862) appeared after the 4th addition. This method leads to numbers without odd digits in a few steps for many integers.

输入:

Your program should read lines of text from standard input. Each line will contain an integer $n < 4,294,967,295$. Assume each line of input will always have an answer and that it is computable with less than 1000 iterations (additions).

输出:

For each line of input, generate a line of output which is the number of iterations (additions) to compute first result without odd digits and the result. They should be on one line and separated by a single space character.

Test 1

测试输入

168

预期输出

4 26862

Question 2 of 2

Choose a machine learning method that you have used. Give a brief explanation of how the method works and what made it suitable for the problem you were working on.