



Maratona de Programação da SBC 2017

Sub-Regional Brasil do ACM ICPC

9 de Setembro de 2017

Aquecimento

Informações Gerais

Este caderno contém 3 problemas; as páginas estão numeradas de 1 a 4, não contando esta página de rosto. Verifique se o caderno está completo.

A) Sobre os nomes dos programas

1) Sua solução deve ser chamada `codigo_de_problema.c`, `codigo_de_problema.cpp`, `codigo_de_problema.pas`, `codigo_de_problema.java` ou `codigo_de_problema.py`, onde `codigo_de_problema` é a letra maiúscula que identifica o problema. Lembre que em Java o nome da classe principal deve ser igual ao nome do arquivo.

B) Sobre a entrada

- 1) A entrada de seu programa deve ser lida da *entrada padrão*.
- 2) A entrada é composta de um único caso de teste, descrito em um número de linhas que depende do problema.
- 3) Quando uma linha da entrada contém vários valores, estes são separados por um único espaço em branco; a entrada não contém nenhum outro espaço em branco.
- 4) Cada linha, incluindo a última, contém exatamente um caractere final-de-linha.
- 5) O final da entrada coincide com o final do arquivo.

C) Sobre a saída

- 1) A saída de seu programa deve ser escrita na *saída padrão*.
- 2) Quando uma linha da saída contém vários valores, estes devem ser separados por um único espaço em branco; a saída não deve conter nenhum outro espaço em branco.
- 3) Cada linha, incluindo a última, deve conter exatamente um caractere final-de-linha.

Promoção:



Sociedade Brasileira de Computação

Problema A

Fatorial

O *fatorial* de um número inteiro positivo N , denotado por $N!$, é definido como o produto dos inteiros positivos menores do que ou iguais a N . Por exemplo $4! = 4 \times 3 \times 2 \times 1 = 24$.

Dado um inteiro positivo N , você deve escrever um programa para determinar o menor número k tal que $N = a_1! + a_2! + \dots + a_k!$, onde cada a_i , para $1 \leq i \leq k$, é um número inteiro positivo.

Por exemplo, para $N = 10$ a resposta é 3, pois é possível escrever N como a soma de três números fatoriais: $10 = 3! + 2! + 2!$. Para $N = 25$ a resposta é 2, pois é possível escrever N como a soma de dois números fatoriais: $25 = 4! + 1!$.

Entrada

A entrada consiste de uma única linha que contém um inteiro N ($1 \leq N \leq 10^5$).

Saída

Seu programa deve produzir uma única linha com um inteiro representando a menor quantidade de números fatoriais cuja soma é igual ao valor de N .

Exemplo de entrada 1 10	Exemplo de saída 1 3
Exemplo de entrada 2 25	Exemplo de saída 2 2

Problema B

Jogo de Estratégia

Um jogo de estratégia, com J jogadores, é jogado em volta de uma mesa. O primeiro a jogar é o jogador 1, o segundo a jogar é o jogador 2 e assim por diante. Uma vez completada uma rodada, novamente o jogador 1 faz sua jogada e a ordem dos jogadores se repete novamente. A cada jogada, um jogador garante uma certa quantidade de Pontos de Vitória. A pontuação de cada jogador consiste na soma dos Pontos de Vitória de cada uma das suas jogadas.

Dado o número de jogadores, o número de rodadas e uma lista representando os Pontos de Vitória na ordem em que foram obtidos, você deve determinar qual é o jogador vencedor. Caso mais de um jogador obtenha a pontuação máxima, o jogador com pontuação máxima que tiver jogado por último é o vencedor.

Entrada

A entrada consiste de duas linhas. A primeira linha contém dois inteiros J e R , o número de jogadores e de rodadas respectivamente ($1 \leq J, R \leq 500$). A segunda linha contém $J \times R$ inteiros, correspondentes aos Pontos de Vitória em cada uma das jogadas feitas, na ordem em que aconteceram. Os Pontos de Vitória obtidos em cada jogada serão sempre inteiros entre 0 e 100, inclusive.

Saída

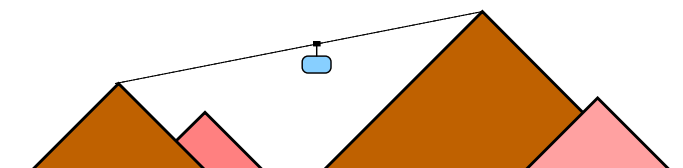
Seu programa deve produzir uma única linha, contendo o inteiro correspondente ao jogador vencedor.

Exemplo de entrada 1 3 3 1 1 1 1 2 2 2 3 3	Exemplo de saída 1 3
Exemplo de entrada 2 2 3 0 0 1 0 2 0	Exemplo de saída 2 1

Problema C

Teleférico

A turma da faculdade vai fazer uma excursão na serra e todos os alunos e monitores vão tomar um teleférico para subir até o pico de uma montanha. A cabine do teleférico pode levar C pessoas no máximo, contando alunos e monitores, durante uma viagem até o pico. Por questão de segurança, deve haver pelo menos um monitor dentro da cabine junto com os alunos. Por exemplo, se cabem $C = 10$ pessoas na cabine e a turma tem $A = 20$ alunos, os alunos poderiam fazer três viagens: a primeira com 8 alunos e um monitor; a segunda com 6 alunos e um monitor; e a terceira com 6 alunos e um monitor.



Dados como entrada a capacidade C da cabine e o número total A de alunos, você deve escrever um programa para calcular o número mínimo de viagens do teleférico.

Se você estiver com muita preguiça hoje, não se preocupe: virando a página você encontra soluções para este problema.

Entrada

A primeira linha da entrada contém um inteiro C , representando a capacidade da cabine ($2 \leq C \leq 100$). A segunda linha da entrada contém um inteiro A , representando o número total de alunos na turma ($1 \leq A \leq 1000$).

Saída

Seu programa deve imprimir uma linha contendo um número inteiro representando o número mínimo de viagens do teleférico para levar todos os alunos até o pico da montanha.

Exemplo de entrada 1 10 20	Exemplo de saída 1 3
Exemplo de entrada 2 12 55	Exemplo de saída 2 5
Exemplo de entrada 3 100 87	Exemplo de saída 3 1

Solução em C++

```
#include <iostream>
using namespace std;

int main(void){
    int C,A;

    cin >> C >> A; // le a entrada

    int quociente = A/(C-1);
    int resto = A%(C-1);
    int resposta = quociente;
    if ( resto > 0 ) resposta++;

    cout << resposta << endl; // escreve a resposta

    return 0;
}
```

Solução em Python

```
C = int(input()) # le a entrada
A = int(input())

quociente, resto = A/(C-1), A%(C-1)
resposta = quociente
if resto>0:
    resposta += 1

print(resposta) # escreve a resposta
```

Solução em Java

```
import java.util.Scanner;

public class C {
    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);
        int quociente, resto, resposta;
        int C,A;

        C=in.nextInt(); // le a entrada
        A=in.nextInt();

        quociente=A/(C-1);
        resto=A%(C-1);
        resposta=quociente;
        if (resto>0)
            resposta=resposta+1;

        System.out.println(resposta); // imprime a resposta
    }
}
```