



III MARATONA DE PROGRAMAÇÃO IFG CÂMPUS JATAÍ

Categoria: Intermediários

FORMATO DA COMPETIÇÃO

Os times receberão uma prova com diversos problemas que devem ser resolvidos durante 3 horas de competição. Os problemas deverão ser resolvidos em alguma das linguagens de programação disponíveis como C, C++, ou Java. No caso do Java, a classe principal deve ter o mesmo nome do arquivo fonte, sendo que esse nome é indicado abaixo do título de cada problema. Seus programas devem realizar as leituras da entrada padrão e as impressões na saída padrão. Os times poderão resolver um problema da prova em uma linguagem de programação e um outro problema em outra linguagem, sem qualquer prejuízo. Para a implementação **os times terão à sua disposição um computador e todo o material escrito que desejarem. Entretanto, não poderão fazer uso de material armazenado em meio digital ou ter acesso à Internet durante a competição sob pena de serem eliminados.**

Quando um time julgar que tem um programa que resolve um problema, ele pode submetê-lo à correção dos juízes, que compilam e executam este programa para uma bateria de testes desconhecida dos times. Um problema é considerado resolvido se, para todos os testes da bateria, ele devolve o resultado esperado pelos juízes. Para cada submissão o time recebe uma resposta, que pode ser satisfatória (e o problema está resolvido pelo time) ou indica algum erro ocorrido, como: resposta errada, erro de execução, erro de compilação de acordo com a tabela a seguir:

Resposta	Significado
<i>Not answered yet</i>	Sua submissão ainda não foi avaliada. Aguarde alguns instantes e atualize a página.
YES	Sua submissão está correta.

24 a 28 de junho | 2019

XIV SECOMP

SEMANA DA COMPUTAÇÃO

www.secompif.com.br

<i>NO - Compilation error</i>	Submissão incorreta por falha de compilação.
<i>NO - Runtime error</i>	Submissão incorreta por falha de segmentação.
<i>NO - Time limit exceeded</i>	Submissão incorreta por ultrapassar o tempo limite.
<i>NO - Presentation error</i>	Submissão incorreta por não seguir o padrão de saída.
<i>NO - Wrong answer</i>	Saída incorreta.

O time vencedor é aquele que resolve a maior quantidade de problemas nas 3 horas de competição. Empates no número de problemas resolvidos são classificados pelo tempo corrigido. Ganha aquele que tem o menor tempo corrigido. O tempo corrigido do time é dado pela soma dos tempos corrigidos somente dos problemas corretamente resolvidos pelo time. O tempo corrigido de um problema é dado pelo número de minutos decorridos desde o início da competição até o momento da primeira submissão correta somado com uma penalidade de 10 minutos por submissão incorreta feita anteriormente neste problema. Em caso de empate, será considerado vencedor o time cuja última submissão correta tenha sido feita mais cedo. Persistindo o empate, passará à penúltima submissão correta e assim por diante. Se persistir o empate a organização fará um sorteio entre os times envolvidos.

A organização da competição é responsável pela decisão de qualquer caso não previsto. Os times inscritos permitem o uso e divulgação de suas imagens e dos programas submetidos pela organização da competição.

Uma boa plataforma para treinamento antes da Maratona é a URI JUDGE ONLINE (<https://www.urionlinejudge.com.br>)

Na próxima página você encontra um exemplo de um problema a ser solucionado e como seriam as soluções em cada uma das linguagens permitidas.

Comitê Organizador da II Maratona de Programação



Exemplo de Problema e Solução

PROBLEMA 1001

Leia 2 valores inteiros e armazene-os nas variáveis **A** e **B**. Efetue a soma de **A** e **B** atribuindo o seu resultado na variável **X**. Imprima **X** conforme exemplo apresentado abaixo. Não apresente mensagem alguma além daquilo que está sendo especificado e não esqueça de imprimir o fim de linha após o resultado, caso contrário, você receberá "Presentation Error".

Entrada

A entrada contém 2 valores inteiros.

Saída

Imprima a variável **X** conforme exemplo abaixo, com um espaço em branco antes e depois da igualdade. Obs: O **X** está em maiúsculo e deve ter um espaço antes e um espaço depois do sinal de igualdade.

Exemplos de Entrada	Exemplos de Saída
10 9	X = 19
-10 4	X = -6
15 -7	X = 8

```
EXEMPLO C (PROBLEMA 1001)
#include <stdio.h>
int main() {
    int A, B, X;

    scanf("%d", &A);
    scanf("%d", &B);

    X=A+B;
    printf("X = %d\n", X);
    return 0;
}
```

24 a 28 de junho | 2019

XIV SECOMP

SEMANA DA COMPUTAÇÃO

www.secompif.com.br

EXEMPLO C++ (PROBLEMA 1001)

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

int main() {
    int A, B, X;

    cin >> A >> B;
    X = A + B;
    cout << "X = " << X << endl;

    return 0;
}
```

EXEMPLO JAVA (PROBLEMA 1001)

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;

public class Main {

    public static void main(String[] args) throws IOException {

        InputStreamReader ir = new InputStreamReader(System.in);
        BufferedReader in = new BufferedReader(ir);

        int A, B, X;

        A = Integer.parseInt(in.readLine());
        B = Integer.parseInt(in.readLine());

        X = A + B;

        System.out.printf("X = %d\n", X);

    }
}
```