

Aula 0

Introdução ao C++ e à Maratona

Maratona de Programação

FEI

April 15, 2025

Programação Competitiva

- A Prova
 - ▶ Em times compostos por três estudantes, os competidores terão que resolver, durante cinco horas, o maior número de problemas possível.
- O que têm à disposição?
 - ▶ Apenas um computador
 - ▶ Material impresso

Programação Competitiva

- Formato dos Problemas
 - ▶ Enunciado
 - ▶ Entrada (Input)
 - ▶ Saída (Output)
- Como entregar a solução de um problema?
 - ▶ Para cada problema, o competidor deve submeter o código completo ao juiz virtual, onde ele será avaliado por meio de uma série de casos de teste predeterminados.

Programação Competitiva

beecrowd | 2172

Evento

Por Nivaldo Carvalho, UNIFE  Brazil

Timelimit: 2



Prog e Cackto começaram recentemente a jogar um jogo de RPG chamado Fortaleza. Neste, para o jogador evoluir de nível o mesmo precisa derrotar monstros, nos quais dá um valor de experiência (XP) para o jogador.

A produtora do jogo, Jogos Extremos, anunciou que na próxima semana irá realizar o primeiro evento XP no qual aumentará a experiência dos monstros em X vezes. Como Prog e Cackto estão em um nível muito alto no qual os monstros tem um valor muito alto de pontos de experiência, eles estão tendo dificuldades de calcular a quantidade de pontos de experiência que os monstros terão durante o evento. Você pode ajudá-los?

Entrada

Haverá diversos casos de teste. Cada caso de teste contém dois valores X ($0 < X \leq 3$) indicando o valor de aumento da EXP dos monstros e M ($10 \leq M \leq 2^{32}-1$) indicando o valor de EXP do monstro. A entrada termina com os valores $X = 0$ e $M = 0$, nos quais não devem ser processados.

Saída

Para cada caso, seu programa deverá mostrar um valor E , referente ao novo EXP do monstro.

Exemplo de Entrada	Exemplo de Saída
1 544768710	544768710
2 538533133	1077066266
3 38884958	116654874
0 0	

Tipos de Dados

Tipo	int	long long	double	bool	char
Descrição	Inteiro de 32 bits	Inteiro de 64 bits	Ponto flutuante de dupla precisão	Verdadeiro ou falso	Caractere de 8 bits
Tamanho	4 bytes	8 bytes	8 bytes	1 bit	1 byte
Intervalo	-2^{31} até $+2^{31} - 1$	-2^{63} até $+2^{63} - 1$	$-1.7E + 308$ até $+1.7E + 308$	0 ou 1	-2^7 até $+2^7 - 1$

Tipos de Dados

- Como declarar uma variável em C++?
 - ▶ Especificando o tipo da variável seguido de seu nome.

```
1 int num = 1234567891;  
2  
3 long long num = 1234567891234567891;  
4  
5 bool condicao = 0;  
6  
7 char letra = 'K';
```

Preparando o Ambiente

- `include<bits/stdc++.h>`
 - ▶ Inclui todas as bibliotecas padrão do C++.
- `using namespace std;`
 - ▶ Permite usar elementos do *std* (Standard namespace) sem escrever `std::`.
- `int main()`
 - ▶ Função principal do programa.
- `return 0;`
 - ▶ Indica que a execução foi bem-sucedida.

Preparando o Ambiente

- Código em C++

```
1 #include <bits/stdc++.h>
2
3 using namespace std;
4
5 int main(){
6
7     return 0;
8 }
```

Entrada (Input)

- Como receber uma entrada em C++?
 - ▶ Para receber o valor de uma variável já declarada, podemos utilizar o comando `cin`.

```
1 int num;  
2  
3 cin>>num;
```

Saída (Output)

- Como imprimir uma saída em C++?
 - ▶ Para imprimir o valor de uma variável, podemos utilizar o comando `cout` junto com `endl` para realizar a quebra de linha.

```
1 cout<<"Hello World !"<<endl;
```

Laços de Repetição

- O que é?
 - ▶ É uma estrutura de programação que permite executar um bloco de código diversas vezes.
- Laços mais comuns:
 - ▶ For
 - ▶ While
 - ▶ Do-While

Laço de Repetição - For

- A sintaxe do *for* em C++ deve ter:
 - ▶ Inicialização
 - ▶ Condição
 - ▶ Atualização
- A quantidade de repetições é preestabelecida.

```
1 for (inicializacao; condicao; atualizacao) {  
2     // logica  
3 }
```

Laço de Repetição - While

- A sintaxe do *while* em C++ deve ter:
 - ▶ Condição
- Ocorre enquanto a condição de parada não for atendida, sem garantir nenhuma repetição.

```
1 while(condicao) {  
2     // logica  
3 }
```

Laço de Repetição - Do-While

- A sintaxe do *do-while* em C++ deve ter:
 - ▶ Condição
- Ocorre enquanto a condição de parada não for atendida, garantindo pelo menos uma repetição.

```
1 do{  
2     // logica  
3 } while(condicao)
```

Arrays

- Um array é uma sequência de elementos do mesmo tipo armazenados de forma contígua na memória.
- Cada elemento pode ser acessado por um índice que representa sua posição no array.
- Para declarar um array, devemos especificar o tipo de variável a ser armazenada, seguido de seu nome e do tamanho máximo.

```
1 int arr[10];  
2  
3 for(int i = 0; i < 10; i++){  
4     cin>>arr[i];  
5 }
```

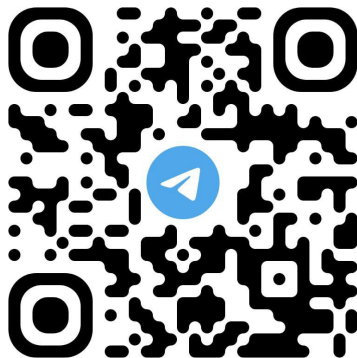
- É uma plataforma online que utilizamos para treinar com problemas de diferentes dificuldades.



Problemas de Programação Competitiva

- **Exercício 1:** Distância Entre Dois Pontos
- **Exercício 2:** Evento
- **Exercício 3:** Fibonacci Fácil
- **Exercício 4:** Troca em Vetor I
- **Exercício 5:** Preenchimento de Vetor IV

Grupo do Telegram



Maratona FEI

Referências

- [1] Usaco Guide. <https://usaco.guide/general/>.
- [2] Maratona SBC. <https://maratona.sbc.org.br>.
- [3] Beecrowd. <https://judge.beecrowd.com>.