

NÚMERO 1

Linguagem de Programação
C++ (g++ 10.2.1, c++20 partial)

ENVIAR SOLUÇÃO

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main()
6 {
7     int num[10];
8     for (int i = 0; i < 10; i++)
9     {
10         cin >> num[i];
11     }
12     for (int i = 0; i < 10; i++)
13     {
14         if (num[i] % 2 == 0)
15         {
16             cout << num[i] << " ";
17         }
18     }
19     cout << endl;
20     for (int i = 0; i < 10; i++)
21     {
22         if (num[i] % 2 != 0)
23         {
24             cout << num[i] << " ";
25         }
26     }
27     return 0;
28 }
```

Skateboarding Resultado final

POR PTB2 09:41 08/08/2024

Linguagem de Programação
C++ (g++ 10.2.1, c++20 partial)

ENVIAR SOLUÇÃO

Aceito

Seu código acertou todos os casos de teste. Parabéns!

Você recebeu 25 XP! 🎉

10 pelo exercício + 15 de bônus de consistência 🌟

Você está a 1 dias em ofensiva. Mantenha o ritmo e siga evoluindo!

41 42

CANCELAR PRÓXIMO 9

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main()
6 {
7     int num[10];
8     for (int i = 0; i < 10; i++)
9     {
10         cin >> num[i];
11     }
12     for (int i = 0; i < 10; i++)
13     {
14         if (num[i] % 2 == 0)
15         {
16             cout << num[i] << " ";
17         }
18     }
19     cout << endl;
20     for (int i = 0; i < 10; i++)
21     {
22         if (num[i] % 2 != 0)
23         {
24             cout << num[i] << " ";
25         }
26     }
27     return 0;
28 }
```

Skateboarding Resultado final

POR PTB2 09:41 08/08/2024

NÚMERO 2

Entrada

A primeira linha da entrada consiste de 10 inteiros separados por um espaço em branco. A segunda linha contém um inteiro X .

Saída

Se X aparece no vetor, a saída consiste de duas linhas, a primeira linha contém um inteiro representando a quantidade de vezes que X apareceu no vetor e a segunda linha contém os índices do vetor que contêm valor igual ao X . Se X não aparecer no vetor, imprima apenas a mensagem "Mia x".

Exemplos de Entrada	Exemplos de Saída
1 2 3 4 5 6 7 8 9 10 0	Mia x
3 2 3 4 3 6 3 8 3 10 3	5 0 2 4 6 8

Traduzido por Luis Paulo

```

1 #include <iostream>
2 using namespace std;
3 int main()
4 {
5     int num[10], x, qtd, aparicoes[10], j = 0;
6     qtd = 0;
7     j = 0;
8     for (int i = 0; i < 10; i++)
9     {
10         cin >> num[i];
11     }
12     cin >> x;
13     for (int i = 0; i < 10; i++)
14     {
15         if (num[i] == x)
16         {
17             qtd++;
18             aparicoes[j] = i;
19             j++;
20         }
21     }
22     if (qtd > 0)
23     {
24         cout << qtd << endl;
25         for (int i = 0; i < qtd; i++)
26         {
27             cout << aparicoes[i] << " ";
28         }
29     }
30     else
31     {
32         cout << "Mia x";
33     }
34     return 0;
35 }
```

Entrada

A primeira linha da entrada consiste de 10 inteiros separados por um espaço em branco. A segunda linha contém um inteiro X .

Saída

Se X aparece no vetor, a saída consiste de duas linhas, a primeira linha contém um inteiro representando a quantidade de vezes que X apareceu no vetor e a segunda linha contém os índices do vetor que contêm valor igual ao X . Se X não aparecer no vetor, imprima apenas a mensagem "Mia x". Lembre-se que a primeira posição de um vetor tem índice 0.

Traduzido por Luis Paulo

Escrevendo Solução

Busca Simples no Vetor 02

Faça um programa para ler 10 valores inteiros, armazená-los em um vetor, depois ler um inteiro X . Se X aparece no vetor, imprima quantas vezes ele aparece e depois quais os índices que ele aparece. Se X não aparecer no vetor, imprima apenas a mensagem "Mia x". Lembre-se que a primeira posição de um vetor tem índice 0.

Entrada

A primeira linha da entrada consiste de 10 inteiros separados por um espaço em branco. A segunda linha contém um inteiro X .

Saída

Se X aparece no vetor, a saída consiste de duas linhas, a primeira linha contém um inteiro representando a quantidade de vezes que X apareceu no vetor e a segunda linha contém os índices do vetor que contêm valor igual ao X . Se X não aparecer no vetor, imprima apenas a mensagem "Mia x". Lembre-se que a primeira posição de um vetor tem índice 0.

Traduzido por Luis Paulo

ACEITO

Seu código acertou todos os casos de teste. Parabéns!

Exemplos de Entrada

Exemplos de Entrada	Exemplos de Saída
1 2 3 4 5 6 7 8 9 10 0	Mia x
3 2 3 4 3 6 3 8 3 10 3	5 0 2 4 6 8

Traduzido por Luis Paulo

```

1 #include <iostream>
2 using namespace std;
3 int main()
4 {
5     int num[10], x, qtd, aparicoes[10], j = 0;
6     qtd = 0;
7     j = 0;
8     for (int i = 0; i < 10; i++)
9     {
10         cin >> num[i];
11     }
12     cin >> x;
13     for (int i = 0; i < 10; i++)
14     {
15         if (num[i] == x)
16         {
17             qtd++;
18             aparicoes[j] = i;
19             j++;
20         }
21     }
22     if (qtd > 0)
23     {
24         cout << qtd << endl;
25         for (int i = 0; i < qtd; i++)
26         {
27             cout << aparicoes[i] << " ";
28         }
29     }
30     else
31     {
32         cout << "Mia x";
33     }
34     return 0;
35 }
```

NÚMERO 3

Imprina o valor do menor elemento lido. Depois imprima uma linha contendo todos os índices que o menor valor aparece no vetor. Depois substitua todas as ocorrências do menor valor no vetor por -1 e imprima o vetor resultante.

Entrada

A entrada consiste de dez linhas. Cada linha contém um inteiro.

Saída

A saída consiste de 3 linhas. A primeira linha contém a mensagem "Menor: Y", sendo Y o valor do menor elemento do vetor. A segunda linha contém a mensagem "Ocorrencias:" seguida dos índices que o menor valor aparece no vetor. A terceira linha contém o vetor após serem realizadas as substituições.

Restrições

Todos os valores fornecidos serão não negativos menores que 100.

Exemplos de Entrada	Exemplos de Saída
1 2 3 4 5 6 7 8 9 10	Menor: 1 Ocorrencias: 0 -1 2 3 4 5 6 7 8 9 10
1 1 1 1 1 1 2 2 2 2	Menor: 1 Ocorrencias: 0 1 2 3 4 -1 -1 -1 -1 2 2 2 2

```
1 #include <iostream>
2 #include <cmath>
3
4 using namespace std;
5
6 int main()
7 {
8     int num[10], menor = 101, ocorrencias[10], j = 0;
9     for (int i = 0; i < 10; i++)
10    {
11        cin >> num[i];
12        if (num[i] < menor)
13        {
14            menor = num[i];
15        }
16    }
17    for (int i = 0; i < 10; i++)
18    {
19        if (num[i] == menor)
20        {
21            ocorrencias[j] = i;
22            j++;
23            num[i] = -1;
24        }
25    }
26    cout << "Menor: " << menor << endl;
27    cout << "Ocorrencias: ";
28    for (int i = 0; i < j; i++)
29    {
30        cout << ocorrencias[i] << " ";
31    }
32    cout << endl;
33    for (int i = 0; i < 10; i++)
34    {
35        cout << num[i] << " ";
36    }
37    cout << endl;
38    return 0;
39 }
```

Imprina o valor do menor elemento lido. Depois imprima uma linha contendo todos os índices que o menor valor aparece no vetor. Depois substitua todas as ocorrências do menor valor no vetor por -1 e imprima o vetor resultante.

Entrada

A entrada consiste de dez linhas. Cada linha contém um inteiro.

Saída

A saída consiste de 3 linhas. A primeira linha contém a mensagem "Menor: Y", sendo Y o valor do menor elemento do vetor. A segunda linha contém a mensagem "Ocorrencias:" seguida dos índices que o menor valor aparece no vetor. A terceira linha contém o vetor após serem realizadas as substituições.

Restrições

Todos os valores fornecidos serão não negativos menores que 100.

Exemplos de Entrada	Exemplos de Saída
1 2 3 4 5 6 7 8 9 10	Menor: 1 Ocorrencias: 0 -1 2 3 4 5 6 7 8 9 10
1 1 1 1 1 1 2 2 2 2	Menor: 1 Ocorrencias: 0 1 2 3 4 -1 -1 -1 -1 2 2 2 2

Traduzido por Luis Paulo

```
1 #include <iostream>
2 #include <cmath>
3
4 using namespace std;
5
6 int main()
7 {
8     int num[10], menor = 101, ocorrencias[10], j = 0;
9     for (int i = 0; i < 10; i++)
10    {
11        cin >> num[i];
12        if (num[i] < menor)
13        {
14            menor = num[i];
15        }
16    }
17    for (int i = 0; i < 10; i++)
18    {
19        if (num[i] == menor)
20        {
21            ocorrencias[j] = i;
22            j++;
23            num[i] = -1;
24        }
25    }
26    cout << "Menor: " << menor << endl;
27    cout << "Ocorrencias: ";
28    for (int i = 0; i < j; i++)
29    {
30        cout << ocorrencias[i] << " ";
31    }
32    cout << endl;
33    for (int i = 0; i < 10; i++)
34    {
35        cout << num[i] << " ";
36    }
37    cout << endl;
38    return 0;
39 }
```

Escrevendo Solução

Substituição no Vetor

Faça um programa para ler 10 valores inteiros e armazenar em um vetor. Imprima o valor do menor elemento lido. Depois imprima uma linha contendo todos os índices que o menor valor aparece no vetor. Depois substitua todas as ocorrências do menor valor no vetor por -1 e imprima o vetor resultante.

Entrada
A entrada consiste de dez linhas. Cada linha é um inteiro.

Saída
A saída consiste de 3 linhas. A primeira é o menor valor lido. A segunda é a mensagem "Ocorrencias:" seguida dos índices que o menor valor aparece no vetor. A terceira linha contém o vetor após a substituição.

Restrições
Todos os valores fornecidos serão não-negativos.

Exemplos de Entrada

Exemplos de Entrada	Exemplos de Saída
1 2 3 4 5 6 7 8 9 10	Menor: 0 Ocorrencias: 0 -1 2 3 4 5 6 7 8 9 10
1 2 3 4 5 6 7 8 9 10	Menor: 1 Ocorrencias: 0 0 1 2 3 4 5 6 7 8 9

Linguagem de Programação: C++ (g++ 10.2.1, c++20 partial)

ENVIAR SOLUÇÃO

ACEITO

Seu código acertou todos os casos de teste. Parabéns!

CANCELAR **PRÓXIMO 9**

```
using namespace std;
int main()
{
    int menor = 10;
    int num[10];
    int ocorrencias[10];
    for (int i = 0; i < 10; i++)
    {
        cin >> num[i];
        if (num[i] < menor)
        {
            menor = num[i];
            ocorrencias[i] = 1;
        }
        else
            ocorrencias[i] = 0;
    }
    cout << "Menor: " << menor << endl;
    cout << "Ocorrencias: ";
    for (int i = 0; i < 10; i++)
    {
        cout << ocorrencias[i] << " ";
    }
}
```

NÚMERO 4

ENVIAR SOLUÇÃO

imprima "pertence" se o número buscado pertencer ao vetor, ou "nao_pertence" caso contrário.

Entrada
As entradas são compostas pelo tamanho N do vetor, os elementos do vetor $V[i]$ e o número buscado X .

Saída
A saída é composta por pertence se caso o número estiver no vetor ou nao_pertence caso contrário.

Restrições

- $2 \leq N \leq 1000$
- $1 < X \leq 10^8$

Exemplos de Entrada

Exemplos de Entrada	Exemplos de Saída
5 1 2 3 4 5 8	nao_pertence
5 1 2 3 4 5 3	pertence
7 1 2 3 4 5 6 7 7	pertence

Linguagem de Programação: C++ (g++ 10.2.1, c++20 partial)

```
#include <iostream>
using namespace std;
int main()
{
    int n, x, cont = 0;
    cin >> n;
    int v[n];
    for (int i = 0; i < n; i++)
    {
        cin >> v[i];
    }
    cin >> x;
    for (int i = 0; i < n; i++)
    {
        if (v[i] == x)
        {
            cont = 1;
        }
    }
    if (cont == 0)
    {
        cout << "nao_pertence" << endl;
    }
    else
    {
        cout << "pertence" << endl;
    }
    return 0;
}
```

The screenshot shows a browser window with several tabs open. The active tab is titled "Busca Simples no Vetor" from neps.academy. The page displays a success message: "Seu código acertou todos os casos de teste. Parabéns!" (Your code got all the test cases right. Congratulations!). It also shows a progress bar indicating 41 XP received and 42 XP available. Below the message, there are two buttons: "CANCELAR" (Cancel) and "PRÓXIMO" (Next). The background of the page shows parts of the exercise code and input/output sections.