

NÚMERO 1 DA APOSTILA

```
#include <iostream>

using namespace std;

int main()
{
    string nomes[6] = {"Tereza", "Jonas", "Beatriz", "Jamal", "Alice",
"Renzo"}; // criei um vetor do tipo string com 6 posicoes e guardei os
nomes nelas

    for (int i = 0; i < 6; i++) // fiz um for para poupar linhas de
codigo ao imprimir os nomes
    {
        cout << nomes[i] << endl; // imprimo o nome contido na posicao
i do meu vetor
    }
    system("pause");
    return 0;
}
```

```
#include <iostream>

using namespace std;

int main()
{
    // criei um vetor do tipo int com 9 posicoes e atribui valores as
suas posicoes
    int idades[9] = {14, 16, 14, 12, 8, 20, 17, 16, 14};
    // criei um for para mostrar os valores na posicao i do meu vetor
    for (int i = 0; i < 9; i++)
    {
        cout << idades[i] << endl; // mostra o valor guardado na
posicao i do meu vetor
    }
    system("pause");
    return 0;
}
```

```
#include <iostream>
```

```

using namespace std;

int main()
{
    // criei um vetor do tipo double sem especificar o seu tamanho, mas
    // ao declarar os valores que serao atribuidos a ele, o compilador vai
    // atribuir esse valor
    double notas[] = {20.5, 23.6, 28.7, 30.9, 20.0};
    // criei um for para mostrar os valores na posicao i do meu vetor
    for (int i = 0; i < 9; i++)
    {
        cout << notas[i] << endl; // mostra o valor guardado na posicao
        // i do meu vetor
    }
    system("pause");
    return 0;
}

```

```

#include <iostream>

using namespace std;

int main()
{
    int *sequencia; // cria um vetor sem determinar seu tamanho
    int tamanho;
    cout << "Quantos : ";
    cin >> tamanho;
    sequencia = new int[tamanho]; // meu vetor vai receber o valor da
    // variavel tamanho
    // fiz um for para pedir os valores do vetor ao usuario
    for (int i = 0; i < tamanho; i = i + 1)
    {
        cin >> sequencia[i]; // pede o valor que vai ser atribuido ao
        // vetor na posicao i
    }
    // fiz outro for para imprimir os valores do vetor
    for (int i = 0; i < tamanho; i = i + 1)
    {
        cout << sequencia[i] << endl; // imprime o valor do meu vetor
        // na posicao i
    }
    system("pause");
}

```

```
return 0;
}
```

NÚMERO 2

alunos. O vestibular consiste de uma ou mais provas sobre as matérias do Ensino Médio, visando avaliar os conhecimentos dos candidatos.

Um formato popular de prova de vestibular é a prova objetiva. Neste formato de prova, cada candidato deve escolher uma das cinco alternativas apresentadas pela questão como sendo a correta. Durante a correção dos cartões, cada questão onde a alternativa escolhida pelo candidato é a mesma do gabarito, ele ganha um ponto. Alguns dos vestibulares mais concorridos do Brasil são disputados por dezenas de milhares de candidatos, e, por isso, geralmente usa-se uma folha de leitura óptica e um programa de computador para corrigir as provas de todos os candidatos e gerar a lista com suas pontuações.

Você trabalha no comitê responsável pelo vestibular em uma faculdade e deve escrever um programa que, dado o gabarito e as respostas de um dos candidatos, determina o número de acertos daquele candidato.

Entrada

A entrada contém um único conjunto de testes, que deve ser lido do dispositivo de entrada padrão. A primeira linha da entrada contém um único inteiro N , indicando o número de questões da prova. A segunda linha da entrada contém uma cadeia de N caracteres, indicando o gabarito da prova. A terceira linha da entrada contém outra cadeia de N caracteres, indicando as opções marcadas pelo candidato. Ambas as cadeias contêm apenas os caracteres 'A', 'B', 'C', 'D' e 'E' (sempre em letra maiúscula).

Saída

Seu programa deve imprimir na saída padrão uma única linha contendo um único inteiro, indicando o número de acertos do candidato.

Restrições

- $1 \leq N \leq 80$

Exemplos de Entrada	Exemplos de Saída
7 AEDBCCE ADDCBCE	4
5 ABCDE	5

C++ (g++ 10.2.1; c++20 partial)

ENVIAR SOLUÇÃO

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main(){
6     int n, cont=0;
7     cin >> n;
8     char respostas[n], gabarito[n];
9     for(int i=0; i < n; i++){
10         cin >> gabarito[i];
11     }
12     for(int i=0; i < n; i++) {
13         cin >> respostas[i];
14     }
15     for(int i=0;i<n;i++){
16         if(respostas[i] == gabarito[i]){
17             cont++;
18         }
19     }
20     cout << cont;
21     return 0;
22 }
23
```

alunos. O vestibular consiste de uma ou mais provas sobre as matérias do Ensino Médio, visando avaliar os conhecimentos dos candidatos.

Um formato popular de prova de vestibular é a prova objetiva. Neste formato de prova, cada candidato deve escolher uma das cinco alternativas apresentadas pela questão como sendo a correta. Durante a correção dos cartões, cada questão onde a alternativa escolhida pelo candidato é a mesma do gabarito, ele ganha um ponto. Alguns dos vestibulares mais concorridos do Brasil são disputados por dezenas de milhares de candidatos, e, por isso, geralmente usa-se uma folha de leitura óptica e um programa de computador para corrigir as provas de todos os candidatos e gerar a lista com suas pontuações.

Você trabalha no comitê responsável pelo vestibular em uma faculdade e deve escrever um programa que, dado o gabarito e as respostas de um dos candidatos, determina o número de acertos daquele candidato.

Entrada

A entrada contém um único conjunto de testes, que deve ser lido do dispositivo de entrada padrão. A primeira linha da entrada contém um único inteiro N , indicando o número de questões da prova. A segunda linha da entrada contém uma cadeia de N caracteres, indicando o gabarito da prova. A terceira linha da entrada contém outra cadeia de N caracteres, indicando as opções marcadas pelo candidato. Ambas as cadeias contêm apenas os caracteres 'A', 'B', 'C', 'D' e 'E' (sempre em letra maiúscula).

Saída

Seu programa deve imprimir na saída padrão uma única linha contendo um único inteiro, indicando o número de acertos do candidato.

Restrições


- $1 \leq N \leq 80$

Exemplos de Entrada	Exemplos de Saída
7 AEDBCCE ADDCBCE	4
5 ABCDE	5

C++ (g++ 10.2.1; c++20 partial)

ENVIAR SOLUÇÃO

```
1 #include <iostream>
2
3 using namespace std;
4
5 int main(){
6     int n, cont=0;
7     cin >> n;
8     char respostas[n], gabarito[n];
9     for(int i=0; i < n; i++){
10         cin >> gabarito[i];
11     }
12     for(int i=0; i < n; i++) {
13         cin >> respostas[i];
14     }
15     for(int i=0;i<n;i++){
16         if(respostas[i] == gabarito[i]){
17             cont++;
18         }
19     }
20     cout << cont;
21     return 0;
22 }
23
```



Aceito

Seu código acertou todos os casos de teste. Parabéns!

CANCELARPRÓXIMO 9

NÚMERO 3

ROGIBH-IAP 2024 - Google Di...EXERCÍCIO 10 - Documentos G...IAP-RSF_permanente: Atividade...Busca Simples no Vetor 01 | N...+

←→↻neps.academy/br/exercicio/395

Escrevendo Solução

Busca Simples no Vetor 01

Faça um programa para ler 10 valores inteiros, armazená-los em um vetor, depois ler um inteiro X . Imprima "SIM" se X aparece no vetor, ou "NAO" caso contrário.

Entrada

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

Saída

Imprima "SIM" se X aparece no vetor, ou "NAO" caso contrário.

Restrições

Todos os valores fornecidos não serão negativos nem maiores que 100.

Exemplos de Entrada

Exemplos de Saída

1 1 3 3 4 4 6 6 5 5	SIM
5	
1 1 3 3 4 4 6 6 5 5	NAO
2	

Traduzido por **Daniel Hosomi**

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 inteiros[10], num, cont=0;
8     for(int i=0; i<10; i++){
9         cin >> inteiros[i];
10    }
11    cin >> num;
12    for(int i=0; i<10; i++){
13        if(inteiros[i] == num){
14            cout << "SIM" << endl;
15            cont++;
16            break;
17        }
18    }
19    if(cont == 0){
20        cout << "NAO" << endl;
21    }
22    return 0;
23 }
24
```

23°C

Ensolado

Pesquisar

FOR PTB2

10:30

06/08/2024

ROGIBH-IAP 2024 - Google Di...EXERCÍCIO 10 - Documentos G...IAP-RSF_permanente: Atividade...Busca Simples no Vetor 01 | N...+

←→↻neps.academy/br/exercicio/395

Escrevendo Solução

Busca Simples no Vetor 01

Faça um programa para ler 10 valores inteiros, armazená-los em um vetor, depois ler um inteiro X . Imprima "SIM" se X aparece no vetor, ou "NAO" caso contrário.

Entrada

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

Saída

Imprima "SIM" se X aparece no vetor, ou "NAO" caso contrário.

Restrições

Todos os valores fornecidos não serão negativos nem maiores que 100.

Exemplos de Entrada

Exemplos de Saída

1 1 3 3 4 4 6 6 5 5	SIM
5	
1 1 3 3 4 4 6 6 5 5	NAO
2	

Traduzido por **Daniel Hosomi**

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 inteiros[10], num, cont=0;
8     for(int i=0; i<10; i++){
9         cin >> inteiros[i];
10    }
11    cin >> num;
12    for(int i=0; i<10; i++){
13        if(inteiros[i] == num){
14            cout << "SIM" << endl;
15            cont++;
16            break;
17        }
18    }
19    if(cont == 0){
20        cout << "NAO" << endl;
21    }
22    return 0;
23 }
24
```

23°C

Ensolado

Pesquisar

FOR PTB2

10:30

06/08/2024

Aceito

Seu código acertou todos os casos de teste. Parabéns!

CANCELARPRÓXIMO 8