

Sétima aula prática

Fibonacci Fácil

Adaptado de URI Online Judge | 1151

A seguinte sequência de números 0 1 1 2 3 5 8 13 21... é conhecida como série de Fibonacci. Nessa sequência, cada número, depois dos 2 primeiros, é igual à soma dos 2 anteriores. Escreva um algoritmo que leia um inteiro N ($N < 46$) e mostre o N-ésimo número dessa série.

OBS: Resolva o problema utilizando uma função recursiva.

Entrada

Um valor inteiro N ($0 < N < 46$).

Saída

Exemplo de Entrada	Exemplo de Saída
5	3

Números de Armstrong

Um número é dito Número de Armstrong se atende a seguinte propriedade: a soma de cada dígito do número elevado a quantidade de dígitos deve resultar no próprio número. Difícil? Um exemplo: 153 é um Número de Armstrong pois $1^3 + 5^3 + 3^3 = 153$. Já 125 por exemplo não é um Número de Armstrong pois $1^3 + 2^3 + 5^3 = 134$.

Faça um programa utilizando quatro funções (além da função principal), pegaUnidade, pegaDezena, pegaCentena e pegaUnidadeMilhar.

Entrada

Um valor inteiro N ($0 < N < 10000$).

Saída

Exemplo de Entrada	Exemplo de Saída
153	S
125	N

Números Amigos

Dois números são ditos amigos quando a soma dos divisores próprios de um número é igual ao outro número, por exemplo: 284 e 220 são números amigos, pois os divisores próprios de 220 são 1, 2, 4, 5, 10, 11, 20, 22, 44, 55 e 110. Efetuando a soma destes números obtemos o resultado 284 e vice-versa.

Faça um programa que preenche um vetor com um número par de elementos e após isso passe ele como parâmetro para uma função. A função deve comparar de par em par se os dois números do par são amigos.

Entrada

N elementos num vetor, sendo N par.

Saída

Exemplo de Entrada	Exemplo de Saída
284 220 42 45 78 90	S N N

Exponenciação mais rápida

O algoritmo ingênuo de exponenciação pode ser melhorado se calcularmos x^n de uma maneira diferente. Se n é par, $x^n = x^{\lfloor n/2 \rfloor} \times x^{\lfloor n/2 \rfloor}$. Se n é ímpar, $x^n = x^{\lfloor n/2 \rfloor} \times x^{\lfloor n/2 \rfloor} \times x$.

Implemente o protótipo de função abaixo, de forma que a função calcule x^n da maneira indicada anteriormente:

```
long long power(int n, int x);
```

Observações:

- Proibido utilizar for na função power.
- Não esqueça que funções recursivas precisam de um caso base para terminar.

Entrada

As entradas são n e x , respectivamente.

Saída

Exemplo de Entrada	Exemplo de Saída
2 10	1024

Selection Sort

Faça um programa que leia um vetor de inteiros de tamanho N ($2 \leq N \leq 512$) e ordene seus elementos usando a função `selection_sort`. Esta função deve fazer o seguinte:

1. Encontrar o maior elemento do vetor e movê-lo para a última posição (com uma troca).
2. Chamar-se recursivamente para ordenar os $N-1$ elementos que restaram.

Segue o protótipo da função:

```
void selection_sort(int v[], int n);
```

Entrada

A entrada consiste do tamanho N , seguido por N inteiros.

Saída

Exemplo de Entrada	Exemplo de Saída
5 4 3 4 2 1	1 2 3 4 4

Substring

Considere a função:

```
int ondeEsta(char frase1[], char frase2[]);
```

Esta função retorna em que posição do vetor `frase1` se encontra o vetor `frase2`. Caso o vetor não apareça a função retorna o valor `-1`. Escreva a função e um programa que, usando esta função imprima se um vetor está contido no outro, e no caso positivo em que posição ele está.

Conversão de base

Escreva um programa que converta um número inteiro positivo da base 10 para a base 2. Para isto escreva uma função que tenha o seguinte protótipo:

```
void converter(int numeroBase10, int numeroBase2[32]);
```

A função recebe o número na variável `numeroBase10` e retorna todos os 32 bits no vetor `numeroBase2`. O número na base 2 deve ser armazenado no vetor da seguinte maneira: bit 31 na posição 31, bit 30 na posição 30 e assim sucessivamente. O seu programa deve imprimir o número na ordem correta com todos os 32 dígitos.

Terceira aula prática

Crie um programa que recebe números n , m e um caractere *fill*, e imprime n linhas, cada uma com m caracteres *fill*. O seu programa deve usar a função que tem o protótipo abaixo:

```
void desenha(int n, int m, char fill);
```

Esta função desenha a figura na tela como descrito.

Saída

Exemplo de Entrada	Exemplo de Saída
5 4 #	#### #### #### #### ####

Inversão de dígitos

Crie um programa que recebe um inteiro N e imprime seus dígitos ao contrário. Seu programa deverá usar a função que tem o protótipo abaixo:

```
int digitos_inversos(int n);
```

Esta função retorna o número que possui os dígitos de n , na ordem inversa.

Saída

Exemplo de Entrada	Exemplo de Saída
7631	1367