

Universidade Federal de Goiás – UFG
Instituto de Informática – INF
Bacharelados (Núcleo Básico Comum)
Algoritmos e Estruturas de Dados 1 – 2022/1
Lista de Exercícios nº 02 – Recursividade
Turmas: INF0063/INF0286
Prof. Ronaldo Lopes de Oliveira

Sumário

1	Sequência de Fibonacci recursiva	2
2	Reverso de um número natural	3
3	Conversão de decimal para binário	4
4	Fatorial duplo	5

1 Sequência de Fibonacci recursiva



(+)

Sem dúvida a chamada “Sequência de Fibonacci” (ou “Sucessão de Fibonacci”) é uma das mais famosas sequências numéricas da Matemática. Os dois primeiros termos desta sequência são: $f_0 = 0$ $f_1 = 1$. A partir do terceiro termo, cada termo é obtido somando-se os dois termos imediatamente anteriores a ele, ou seja: $f_n = f_{n-1} + f_{n-2}$, com $n \in \mathbb{N}$ e $n \geq 2$

Considere que seja dado um número n , $n \in \mathbb{N}^*$. Usando o conceito de recursividade, elabore um programa em C para imprimir até o n -ésimo termo da “Série de Fibonacci”.

Observação: Note que a contagem dos termos foi iniciada com o termo 0 (zero): $f_0 = 0$.

Entrada

A única linha da entrada contém um número natural n , indicando a ordem máxima dos termos desejados da “Série de Fibonacci”. Sabe-se que $1 \leq n \leq 1000$.

Saída

Seu programa deve imprimir uma única linha contendo até o n -ésimo termo da série, sempre separados por um único espaço em branco.

Exemplos

Entrada	Saída
0	0
1	0 1
8	0 1 1 2 3 5 8 13 21

Observação: Uma questão interessante é:

Será possível encontrar uma *fórmula* explícita que seja capaz de fornecer um determinado termo da “Sequência de Fibonacci” sem a necessidade de realizar a geração de todos os termos anteriores?

Se isto for possível, gerar utilizar a *fórmula* será mais *eficiente* que utilizar uma função geradora, seja ela recursiva ou iterativa?

Se não for possível, qual o motivo da *impossibilidade*?

2 Reverso de um número natural



(+)

Todo número natural estritamente positivo $n \in \mathbb{N}^*$ possui um *número reverso* correspondente. Por exemplo, considere que n seja escrito da seguinte maneira:

$$n = d_k d_{k-1} d_{k-2} \cdots d_2 d_1 d_0$$

onde $k \in \mathbb{N}^*$ corresponde ao número de dígitos significativos que formam n , ou seja, $d_k \in \{1, 2, 3, \dots, 9\}$ e $d_i \in \{0, 1, 2, \dots, 9\}$, com $0 \leq i < k$.

O *número reverso* de n é $n^r = d_l d_{l-1} d_{l-2} \cdots d_{k-2} d_{k-1} d_k$, sendo d_l o primeiro dígito não nulo, tomados nesta ordem, dentre $d_k d_{k-1} d_{k-2} \cdots d_2 d_1 d_0$ do número original n .

Escreva um programa em C usando recursividade que seja capaz de determinar o *número reverso* de um certo número natural estritamente positivo n fornecido como entrada.

Entrada

A única linha da entrada contém um único número natural estritamente positivo, n , $1 \leq n \leq 10^6$.

Saída

Seu programa deve imprimir uma única linha com o valor de n^r , o *número reverso* de n .

Exemplos

Entrada	Saída
411	114

Entrada	Saída
1230	321

Entrada	Saída
138000	831

3 Conversão de decimal para binário



(++)

Escreva um programa, em C, que receba um número natural $n \in \mathbb{N}$, representado utilizando a notação decimal, e o converta para sua notação binária. O programa deve utilizar uma “função recursiva” para realizar a conversão.

Entrada

A primeira linha conterá um número natural estritamente positivo k , $1 \leq k \leq 1000$, que representa o número de casos de teste que virão em seguida.

Cada uma das k linhas seguintes possuem, cada uma, um único número natural, $0 \leq n_i < 10^6$, com $1 \leq i \leq k$, representado utilizando a notação decimal, a ser convertido para sua correspondente representação binária.

Saída

Seu programa deve imprimir k linhas, cada uma com a correspondente representação binária de um número da entrada.

Exemplos

Entrada	Saída
5	1
1	10
2	11
3	100
4	101
5	

Entrada	Saída
3	101000001
321	1011110001
753	11111111
255	

Entrada	Saída
1	1011011001111100000
373728	

4 Fatorial duplo



(++)

Pode-se definir uma função $f''(n)$, *fatorial duplo* de n , com $n \in \mathbb{N}$, como sendo o produto de todos os números naturais ímpares de 1 até n , inclusive este, quando ele é ímpar. Assim, por exemplo, tem-se que:

$$f''(1) = 1$$

$$f''(2) = 1$$

$$f''(3) = 3$$

$$f''(5) = 15$$

Você deve escrever uma função recursiva, em C, que seja capaz de, recebendo n , imprimir o valor de $f''(n)$.

Entrada

A única linha de entrada contém o valor de n , com $1 \leq n \leq 100$.

Saída

Imprima uma única linha de saída, com o valor de $f''(n)$.

Exemplo

Entrada	Saída
1	1

Entrada	Saída
7	105

Entrada	Saída
10	945