



# CLUBE DE PROGRAMAÇÃO

Atividade 1  
30 de abril de 2013

Ciência da Computação  
Prof. Leandro M. Zatesko

1	Nível 1	1
2	Nível 2	2
3	Nível 3	2
4	Nível 4	3
5	Nível 5	3

- Todos os programas devem ser escritos em C e compiláveis com `gcc -Wall -lm`.
- Os dados devem ser lidos da entrada padrão e impressos na saída padrão.

## 1 Nível 1

**EXERCÍCIO 1 (Subcadeias).** Escreva um programa que lê duas cadeias de caracteres, separadas por quebra-de-linha simples, e conta quantas vezes a primeira ocorre na segunda. Assuma que as cadeias não possuem mais de 100 caracteres nem quebras-de-linha.

Exemplos de entrada	Saídas correspondentes
ANA ANA E MARIANA GOSTAM DE BANANA	4
0 0 LIGO O RÁDIO OCULTO E OUÇO O OVO DO OCASO	5

**EXERCÍCIO 2 (A lista dos primos).** Escreva um programa que lê uma sequência de números inteiros não-nulos no intervalo  $[-2^{31}..2^{31} - 1]$  seguida por 0 e imprime todos os números primos da sequência, separados por quebras-de-linha simples.

Exemplos de entrada	Saídas correspondentes
-7	-7
9	2147483647
2147483647	2
2	
0	
0	
1	
4	
8	
16	
32	
64	
0	

**EXERCÍCIO 3** (O algoritmo mais antigo da Humanidade). Escreva um programa que lê dois números inteiros no intervalo  $[-2^{31}..2^{31} - 1]$ , não sendo ambos nulos, e calcula o máximo divisor comum entre eles.

Exemplos de entrada	Saídas correspondentes
-54 24	6
0 100	100
490 1287	1

## 2 Nível 2

**EXERCÍCIO 4** (Ordenação rápida). Escreva um programa que lê uma sequência de números inteiros no intervalo  $[-2^{31}..2^{31} - 1]$ , finalizada por 0, e imprime a sequência lida ordenada. Considere o número 0 como integrante da sequência e considere que o limite para o tamanho da sequência é a memória disponível. Para a ordenação, utilize o *QuickSort*.

Exemplos de entrada	Saídas correspondentes
2	-13
17	-7
8	-3
-3	0
5	2
4	4
11	5
-13	8
16	11
-7	16
0	17

**EXERCÍCIO 5** (Integral logarítmica). Escreva um programa que lê três números inteiros positivos  $a$ ,  $b$  e  $c$  e encontra o menor inteiro  $x$  no intervalo  $[a..b]$  tal que

$$\frac{x}{\ln x} \sum_{k=0}^{10} \frac{k!}{(\ln x)^k} > c.$$

Assuma que  $c \leq a \leq b \leq 100.000$  e utilize busca binária para encontrar  $x$ . Caso não encontre  $x$  que satisfaça a propriedade esperada, seu programa não deverá imprimir valor algum.

Exemplos de entrada	Saídas correspondentes
2000 8000 600	4265
10000 100000 1000	

## 3 Nível 3

**EXERCÍCIO 6** (Ordenação hippie). Escreva um programa que lê uma sequência de números inteiros no intervalo  $[-2^{31}..2^{31} - 1]$ , finalizada por 0, e imprime a sequência lida ordenada. Considere o número 0 como integrante da sequência e considere que o limite para o tamanho da sequência é a memória disponível. Para a ordenação, utilize o *HeapSort*.

Exemplos de entrada	Saídas correspondentes
2	-13
17	-7
8	-3
-3	0
5	2
4	4
11	5
-13	8
16	11
-7	16
0	17

## 4 Nível 4

**EXERCÍCIO 7 (Hidrocarbonetos anônimos).** Escreva um programa que lê a representação em ASCII de um hidrocarboneto, seguida por um #, e imprime o comprimento da maior subcadeia de carbono. A representação em ASCII de um hidrocarbonato é uma matriz com no máximo 10.000 linhas e no máximo 10.000 colunas composta apenas por letras C e espaços em branco.

Exemplos de entrada	Saídas correspondentes
CCC C C #	5
C CCC C C CCC C C #	9
CCCCC C CCCCCC CCCCC C C CCCC C #	16
C #	1

## 5 Nível 5

**EXERCÍCIO 8 (Harry Potter e os polinômios mágicos).** Na componente curricular de Técnicas Avançadas de Bruxaria, Harry Potter, no seu último ano em Hogwarts, desenvolveu um projeto de pesquisa e descobriu que cada feitiço tem um grau de impacto  $d$ , que é um inteiro positivo. O maior grau de impacto possível que um feitiço pode atingir é 19, e apenas dois bruxos conseguiram proferir um feitiço tão poderoso: Voldemort e o próprio Harry. O acadêmico de bruxaria conseguiu ainda modelar feitiços de grau de impacto  $d$  através de polinômios de grau  $d$ . Mas não é todo polinômio que modela um feitiço. Os polinômios que modelam feitiços são chamados de *polinômios mágicos*. Todo feitiço é modelado por um e só um polinômio mágico e todo polinômio mágico modela um e só um feitiço. Em suas anotações, Harry escreveu:

*Há apenas um polinômio mágico de grau 1, que é  $p(x) \equiv x$ . Um polinômio  $p(x)$  de grau  $d > 1$  é um polinômio mágico se  $p(x) \equiv x^d$  ou se  $p(x) \equiv -x^d + q(x)$ , sendo  $q$  um polinômio mágico de grau  $d - 1$ .*

Seja  $k$  um inteiro positivo, uma  $k$ -feitiçaria é uma sequência de  $k$  feitiços distintos de mesmo grau de impacto. Embora Potter tenha conseguido facilmente calcular quantos feitiços com grau de impacto  $d$  existem, ele está tendo dificuldades em computar quantas  $k$ -feitiçarias existem que utilizam apenas feitiços de grau de impacto no máximo  $d$ . Por exemplo, há 8 2-feitiçarias com grau de impacto no máximo 3: são as 2-feitiçarias definidas por:

1.  $x^2, -x^2 + x$ ;
2.  $-x^2 + x, x^2$ ;
3.  $x^3, -x^3 + x^2$ ;
4.  $x^3, -x^3 - x^2 + x$ ;
5.  $-x^3 + x^2, x^3$ ;
6.  $-x^3 + x^2, -x^3 - x^2 + x$ ;
7.  $-x^3 - x^2 + x, x^3$ ;
8.  $-x^3 - x^2 + x, -x^3 + x^2$ .

Para ajudar Harry Potter, escreva um programa que receba como entrada um inteiro  $d$  e um inteiro  $k$ ,  $0 \leq k \leq d \leq 19$ , e determina quantas  $k$ -feitiçarias existem que utilizam apenas feitiços de grau de impacto no máximo  $d$ .

Exemplos de entrada	Saídas correspondentes
3 2	8
15 7	64864800