

MAC 115 – Introdução à Ciência da Computação

INSTITUTO DE FÍSICA - PRIMEIRO SEMESTRE DE 2004

Primeiro Exercício-Programa (EP1)

Entregue até 26/4/2004, no pelo **Panda**.**O PROBLEMA DE COLLATZ ($3x + 1$)**

Considere o seguinte algoritmo que gera uma seqüência de inteiros. Comece com um inteiro positivo n . Se n é par, divida por 2. Se n é ímpar, multiplique por 3 e some 1. Repita este processo com o novo valor de n , terminando quando atingimos $n = 1$. Por exemplo, a seguinte seqüência de números será gerada para o valor inicial $n = 22$:

22 11 34 17 52 26 13 40 20 10 5 16 8 4 2 1

É uma conjectura bem conhecida que este algoritmo de fato termina para todo n positivo. Sabe-se que isto é verdade para valores de n até 1.000.000, mas não se conhece uma prova para valores arbitrários de n . (Veja, por exemplo, <http://mathworld.wolfram.com/CollatzProblem.html>.)

Para cada n , chamemos de *vida de n* o número de números na seqüência até atingirmos 1, incluindo o 1 final. Por exemplo, a vida de $n = 22$ é 16 (veja acima). Escreva um programa que recebe uma lista de pares de inteiros positivos i e j e determina a vida ‘mais longa’ entre os inteiros de i a j , incluindo ambos i e j .

Entrada

A entrada será uma lista dos pares i e j , com um par por linha. Todos os inteiros serão entre 0 e 1.000.000.

Saída

Para cada par de inteiros i e j , a saída deve ser o par i e j , seguido na mesma linha pelo máximo da vida dos inteiros n com $i \leq n \leq j$.

Exemplo**Entrada**

```
1 10
100 200
201 210
900 1000
```

Saída

```
1 10 20
100 200 125
201 210 89
900 1000 174
```

OBSERVAÇÕES IMPORTANTES SOBRE OS EXERCÍCIOS-PROGRAMAS

Todos os exercícios-programas devem ter o seguinte cabeçalho:

```
/*
/*****
/* Fulano de Tal */
/* Exercício-Programa xx */
/* Curso yy - Turma zz -- Professor: Ciclano de Tal */
/* Computador: ..... */
/* Compilador: ..... */
*****/
```

1. O exercício-programa é **estritamente individual**. Exercícios copiados (com ou sem eventuais disfarces) receberão nota ZERO.
2. **Exercícios atrasados não serão aceitos.**
3. Exercícios com erros de sintaxe (ou seja, erros de compilação) receberão nota ZERO.
4. É **muito importante** que seu programa tenha comentários e esteja bem indentado, ou seja, digitado de maneira a ressaltar a estrutura de subordinação dos comandos do programa (conforme visto em aula). A avaliação dos exercícios-programas levará isto em conta.
5. Cada programa deve ser executado tantas vezes quantas forem necessárias para testar todos os casos possíveis para as entradas. Você pode dizer no seu programa (em comentários), quais dados você usou para testar seu programa. *A escolha desses dados também influirá na sua nota.*
6. Guarde com você até o final do semestre, em seus disquetes, os arquivos de seus programas, pois eles podem ser solicitados pelo professor caso surja alguma dúvida.
7. Entregue o seu programa pelo sistema **Panda**.