

## II Maratona de Programação da FATEC-Rubens Lara

*9 de novembro de 2013*

### **Caderno de Problemas**

Este caderno contém 5 problemas, as páginas estão numeradas de 1 a 6, não contando esta página de rosto.

### **Informações Gerais:**

#### **A) Sobre a entrada**

- 1) A entrada de seu programa deve ser lida da entrada padrão.
- 2) A entrada é composta por um único caso de teste, descrito em um número de linhas que depende do problema.
- 3) Quando uma linha da entrada contém vários valores, estes são separados por um único espaço em branco; a entrada não contém nenhum outro espaço em branco.
- 4) Cada linha, incluindo a última, contém o caractere final de linha.
- 5) O final da entrada coincide com o final do arquivo.

#### **B) Sobre a saída**

- 1) A saída de seu programa deve ser escrita na saída padrão.
- 2) Quando uma linha da saída contém vários valores, estes devem ser separados por um único espaço em branco; a saída não deve conter nenhum outro espaço em branco.
- 3) Cada linha, incluindo a última, deve conter o caractere final de linha.

### **Realização:**



# Problema A

## Brincando na lagoa

Nome do arquivo fonte: lagoa.[c | cpp | java]

Dois amigos, Monte e Carlo, estavam a passear num parque e avistaram uma área quadrada de 4 Km<sup>2</sup>, segundo informação em uma placa do parque. Dentro desta região havia uma lagoa circular de raio 1 Km centrada nesta região. Alguns passos antes dessa região toda havia pedras de modo a você poder jogá-las nesta região, essas pedras podiam ou não cair na lagoa.

Monte começou a jogar várias pedras na região e Carlo contava quantas pedras caíram na lagoa e quantas caíram fora, porém dentro da região quadrada. Carlo percebeu que se automatizassem o processo de contagem de pedras, ele poderia aproximar a área da lagoa sem sequer usar a fórmula que eles aprenderam no colégio. Carlo também percebeu que se jogassem mais e mais pedras o resultado melhoraria muito.

### Entrada

A primeira linha da entrada contém um número  $N$  de pedras a serem jogadas na lagoa. As  $N$  linhas seguintes contêm as coordenadas  $X$  e  $Y$  em ponto flutuante, separadas por um espaço, das pedras que caíram na região quadrada (note que o centro da lagoa fica na coordenada  $X = 0, Y = 0$ ). Para ver se uma pedra caiu ou não na lagoa, basta elevar ao quadrado cada componente da coordenada e verificar se a sua soma é menor ou igual ao raio da lagoa.

### Saída

Para cada pedra deverá ser impressa uma linha com a palavra ACERTOU, caso a pedra acerte na lagoa, ou a palavra ERROU, caso contrário. Ao final deve ser mostrada uma linha com a quantidade  $Q$  de pedras que acertaram na lagoa e na linha seguinte um número  $A$  em ponto flutuante com quatro casas de precisão após o ponto, representado a aproximação de Carlo para a área da lagoa que é dada por  $A = 4.0 \frac{Q}{N}$  em Km<sup>2</sup>.

### Restrições

- $10 \leq N \leq 10^6$
- $-1.0 \leq X, Y \leq 1.0$

### Exemplo

Entrada	Saída
10	ERROU
1.0 1.0	ACERTOU
0.0 0.0	ACERTOU
0.5 0.5	ACERTOU
0.4 -0.37	ACERTOU
0.0 0.12	ACERTOU
0.1 0.5	ERROU
-0.9 0.9	ACERTOU
0.04 0.9	ACERTOU
-0.45 0.45	ACERTOU
-0.2 -0.2	8
	3.2000

# Problema B

## Torneio de esgrima

Nome do arquivo fonte: *esgrima.[c | cpp | java]*

Num torneio de esgrima 3 esgrimistas,  $A$ ,  $B$  e  $C$ , vão competir entre si. Neste torneio fixa-se a primeira rodada que será sempre  $A \times B$ . O competidor que perder sai e dá entrada para o outro competidor. No caso do primeiro combate, o perdedor será substituído por  $C$ . Por exemplo, se  $A$  ganhar,  $B$  sai e entra  $C$  fazendo o combate  $A \times C$ . Entretanto, se Bruno vencer o primeiro combate, então  $A$  sai fazendo o combate  $C \times B$ , e assim por diante.

Para ganhar um combate o competidor precisa fazer 5 toques no oponente e não há limite de tempo. O torneio se encerra quando um competidor atingir 4 vitórias seguidas.

Sua tarefa é automatizar esta competição. Pode haver vários torneios, sempre começando com o combate Alberto x Bruno.

### Entrada

A entrada contém várias linhas. Cada linha contém dois inteiros  $P_1$  e  $P_2$ , separados por um espaço.  $P_1$  e  $P_2$  representam a pontuação dos competidores no combate. A entrada termina com o final de arquivo.

### Saída

Para cada torneio, deve-se exibir uma linha com uma mensagem informando seu vencedor. A mensagem deve ter o seguinte formato:

Vencedor do torneio  $I$ :  $X$

Aonde  $I$  é o número do torneio e  $X$  é o vencedor,  $A$ ,  $B$  ou  $C$ .

### Restrições

- $0 \leq P_1, P_2 \leq 5$
- $1 \leq I \leq 100$

### Exemplo

Entrada	Saída
5 3 3 5 4 5 1 5 2 5 4 5 4 5 5 2 5 1 5 0 5 1	Vencedor do torneio 1: C Vencedor do torneio 2: A

# Problema C

## Teste de raciocínio

Nome do arquivo fonte: teste.[c | cpp | java]

Uma empresa de RH está elaborando testes de inteligência para selecionar candidatos da área de Tecnologia da Informação para uma grande empresa do ramo.

Será entregue uma folha de papel do teste para que o candidato descreva a lógica que está ocorrendo. Esta folha conterá algumas linhas com 4 números inteiros,  $P$ ,  $Q$ ,  $R$  e  $S$ . O número  $S$  deverá ser a concatenação entre  $|P - Q|$ ,  $Q * R$  e  $P + R$ .

Sua tarefa é automatizar o processo de impressão dessas folhas para a realização do teste.

### Entrada

A primeira linha da entrada contém um número inteiro  $N$ , representando o número de linhas da folha de teste. Cada uma das  $N$  linhas seguintes possui três inteiros,  $P$ ,  $Q$  e  $R$ , separados por um espaço.

### Saída

Para cada linha da folha de teste seu programa deve produzir uma linha contendo o inteiro  $S$ , conforme descrito anteriormente.

### Restrições

- $1 \leq N \leq 100$
- $0 \leq P, Q, R < 10$

### Exemplos

Entrada	Saída
4	1169
1 2 8	22011
7 5 4	265
4 6 1	3012
3 0 9	

Entrada	Saída
2	0158
5 5 3	5811
9 4 2	

# Problema D

## Títulos de CD's

*Nome do arquivo fonte: titulos. [c | cpp | java]*

Júlia é uma adolescente que adora baixar músicas e vídeos da internet em seu tablet. Como era de se esperar, o disco rígido de seu tablet rapidamente ficou cheio, então ela decidiu esvaziá-lo descarregando as músicas e vídeos em CD's. Cada CD tem sua própria caixa de plástico e Júlia escreveu na parte da frente da caixa o conteúdo de cada CD. Como o número de CD's aumentou muito, Júlia teve que colocá-los na vertical para que eles coubessem na prateleira. Desta forma, os títulos dos CD's não ficaram mais visíveis, então ela escreveu os títulos de todos os seus CD's em um arquivo texto, mas ela precisa que esses títulos apareçam verticalmente para seja possível colá-los na borda dos CD's, permitindo assim a leitura de seus títulos sem a necessidade de removê-los da prateleira.

Júlia lhe pediu para escrever um programa que deve imprimir os títulos dos CD's na vertical. Ela quer que sejam impressas linhas entre cada título, de tal forma que quando ela imprimi-los, seja possível recortá-los facilmente.

Júlia descobriu que cabem apenas 24 caracteres na borda de cada caixa de CD, portanto, os títulos dos CD's devem possuir 24 caracteres, preenchidos com espaços em branco quando necessário. Se o título de um CD possuir mais que 24 caracteres, apenas os 24 primeiros devem ser impressos.

### Entrada

A entrada é composta de no máximo 50 títulos, um por linha. Cada título contém de 1 a 100 caracteres. Um único caractere '#' em uma linha indica o final da entrada.

### Saída

Seu programa deve produzir os mesmos títulos da entrada impressos verticalmente, ordenados da esquerda para a direita na mesma ordem da entrada. Deve haver uma coluna de caracteres pipe ('|') nas extremidades e separando cada título, e se uma linha de caracteres de subtração ('-') (1 por coluna) no início e no final.

### Restrições

- A entrada é composta por no mínimo 1 e no máximo 50 títulos de CD
- Um título de CD possui no mínimo 1 e no máximo 100 caracteres

## Exemplo

Entrada	Saída
123456789012345678901234	-----
Supernatural, 1a. temporada	1 S O
One Direction	2 u n
#	3 p e
	4 e
	5 r D
	6 n i
	7 a r
	8 t e
	9 u c
	0 r t
	1 a i
	2 l o
	3 , n
	4
	5 l
	6 a
	7 .
	8
	9 t
	0 e
	1 m
	2 p
	3 o
	4 r
	-----

# Problema E

## Média

Nome do arquivo fonte: *media.[c | cpp | java]*

Um professor aplica três avaliações aos seus alunos para determinar a média final de cada um deles, mas utiliza apenas as duas maiores notas para o cálculo da média. Sendo assim, a média de um aluno na disciplina do professor é a média aritmética das duas maiores notas entre as três avaliações que o aluno fez.

Ajude o professor e escreva um programa que dadas as notas das três avaliações de um aluno, determine sua média final.

### Entrada

A entrada é composto por uma linha com três números reais,  $A_1$ ,  $A_2$  e  $A_3$ , separados por um espaço.  $A_1$  representa a nota da primeira avaliação do aluno,  $A_2$  a nota da segunda avaliação e  $A_3$  a nota da terceira avaliação.

### Saída

Seu programa deve produzir uma linha contendo um número real representando a média final do aluno com uma casa de precisão após o ponto.

### Restrições

- $0.0 \leq A_1, A_2, A_3 \leq 10.0$

### Exemplos

<b>Entrada</b> 5.5 7.0 6.0	<b>Saída</b> 6.5
<b>Entrada</b> 8.0 4.0 7.5	<b>Saída</b> 7.8
<b>Entrada</b> 5.0 4.0 4.0	<b>Saída</b> 4.5