

16ª Maratona de Programação

30 de abril de 2016

Caderno de Aquecimento

Regras: *ACM International Collegiate Programming Contest*

Brasil – **Desafio ALGAR TELECOM**

(Este caderno contém 2 problemas; as páginas estão numeradas de 1 a 4)

Instruções

Linhas de compilação:

Linguagem	Linha de comando de compilação
C	gcc -lm teste.c -oteste
C++	g++ -lm teste.cpp -oteste
Java	javac teste.java

Linhas de execução:

Linguagem	Linha de comando de execução
C	./teste < entrada.in
C++	./teste < entrada.in
Java	java teste < entrada.in

Exemplo de problema:

Problema Teste

Arquivo fonte: teste.c, teste.cpp ou teste.java.
Para um dado valor n , calcular a expressão $3n + 1$.

Entrada

A entrada contém várias linhas. Cada linha contém um inteiro n ($1 \leq n \leq 100$).

Saída

Para cada caso de teste imprima uma linha contendo o resultado da expressão.

Exemplo de entrada

8
3
12
2

Exemplo de saída

25
10
37
7

Exemplo de solução em C++ (teste.cpp):

```
#include <iostream>
using namespace std;
int main()
{
    int n;
    while(cin >> n)
    {
        cout << 3*n+1 << "\n";
    }
}
```

Exemplo de solução em C (teste.c):

```
#include <stdio.h>
int main()
{
    int n;
    while(scanf("%d", &n)==1)
    {
        printf("%d\n", 3*n+1);
    }
}
```

Exemplo de solução em Java (teste.java):

```
import java.util.Scanner;
public class teste{
    public static void main(String []
args){
        int n;
        Scanner s = new
Scanner(System.in);
        try{
            while(true){
                n = s.nextInt();
                System.out.println(3*n+1);
            }
        } catch(Exception e){}
    }
}
```

Exemplo do arquivo de entrada (entrada.in):

8
3
12
2

Obs.: para programas criados em Java, o nome da classe deve ser o mesmo que o nome do arquivo. No exemplo acima, como o arquivo fonte deve ser teste.java, a classe principal deve-se chamar teste.

Apoio: <http://crbonilha.com/pt/>

Problema A

Predizendo com KNN

arquivo: knn.c , knn.cpp, knn.java

Cleber é um analista que foi contratado pelo Banco Star para construir um sistema de predição de segmento de clientes. A ideia é que o sistema consiga prever se um cliente pertence à classe *bronze*, *prata*, *ouro* ou *diamante*. Como Cleber é um especialista em mineração de dados, ele pensou: “vou aplicar o algoritmo KNN (K-Nearest Neighbors – K-vizinhos mais próximos) e facilmente conseguirei construir meu sistema!”

O algoritmo funciona assim: dado um novo cliente C, para descobrir a qual segmento ele vai pertencer, deve-se calcular a distância de similaridade entre C e os demais clientes já cadastrados na base. Dentre os K clientes mais similares a C, pegue o segmento predominante e atribua a C. Lembrando que quanto mais similares dois clientes, menor a distância entre eles.

O Banco Star possui as seguintes informações de cada cliente: *idade*, *salário*, *escolaridade* e o *segmento*.

A similaridade entre dois clientes é dada pela distância Euclidiana entre eles. Sejam $c1 = (id1, sal1, escol1)$ e $c2 = (id2, sal2, escol2)$ dois clientes com seus respectivos dados. A distância $d(c1, c2)$ entre eles é:

$$d(c1, c2) = \sqrt{(id1 - id2)^2 + (sal1 - sal2)^2 + 1}, \text{ se } escol1 \neq escol2$$

ou

$$d(c1, c2) = \sqrt{(id1 - id2)^2 + (sal1 - sal2)^2}, \text{ se } escol1 = escol2$$

Sua tarefa é implementar o algoritmo KNN para ajudar o Cleber!

Entrada

A entrada é composta por **T** casos de teste ($1 \leq T \leq 100$). Cada caso de teste inicia com uma linha descrevendo o novo cliente C do Banco Star do tipo: “**id,sal,escol,?**” (sem aspas) onde *id* é um inteiro representando a idade ($18 \leq id \leq 100$), *sal* é um número decimal representando o valor do salário em bitcoin ($1 \leq sal \leq 500$), *escol* $\in \{\text{segundograu}, \text{superior}, \text{posgraduacao}\}$ e o símbolo ‘?’ indica que o segmento do cliente deve ser previsto.

Na linha seguinte estão dois inteiros **K** e **M**, onde K ($1 \leq K \leq M$) é um parâmetro do algoritmo e M ($1 \leq M \leq 100$) é a quantidade de clientes já cadastrados no banco.

Em seguida estão M linhas, cada uma representando um cliente da base do tipo: “**id,sal,escol,seg**” (sem aspas), onde *seg* $\in \{\text{bronze}, \text{prata}, \text{ouro}, \text{diamante}\}$ e os demais atributos seguem as mesmas propriedades do novo cliente.

O número 0 (zero) indica o final do conjunto de casos de teste. Considere que sempre haverá um segmento predominante dentre os K clientes mais similares a C, não sendo necessário aplicar critério algum de desempate.

Saída

Imprima uma linha contendo o segmento *seg* predito pelo algoritmo.

Exemplo de Entrada

```
19,0.5,segundograu,?
3 5
60,50,superior,ouro
25,3,segundograu,bronze
40,100,posgraduacao,diamante
18,0.3,segundograu,bronze
24,0.9,posgraduacao,ouro
60,500,superior,?
5 5
60,50,superior,ouro
25,3,posgraduacao,diamante
18,100,posgraduacao,diamante
18,0.3,superior,ouro
24,0.9,posgraduacao,ouro
0
```

Exemplo de Saída

```
bronze
ouro
```

Problema B

Limite de Velocidade

Arquivo fonte: veloc.c, veloc.cpp ou veloc.java

Cesarino e Juvenil estavam viajando felizes da vida pelos Estados Unidos até que o odômetro de seu possante quebrou. Como eles tinham combinado de dividir as despesas, eles ficaram sem saber quantas milhas haviam sido percorridas. Para tentar resolver o problema, Juvenil resolveu registrar, usando um cronômetro de um relógio, a velocidade e o tempo gasto naquela velocidade. Como a forma de registrar ficou meio estranha, você precisa fazer um programa para ajudá-los. Por exemplo, se os registros fossem:

Velocidade em milhas por hora	Tempo Total gasto em horas
20	2
30	6
10	7

Significa que eles dirigiram 2 horas a 20 milhas por hora. Então $6-2=4$ horas a 30 milhas por hora, então $7-6=1$ hora a 10 milhas por hora. A distância percorrida é então $(2)(20) + (4)(30) + (1)(10) = 40 + 120 + 10 = 170$ milhas. Note que o tempo total percorrido é considerado sempre à partir do início da viagem, não dá linha anterior mostrada nos registros.

Entrada

A entrada consiste de um ou mais conjuntos de registros. Cada conjunto inicia com uma linha contendo um inteiro n , $1 \leq n \leq 10$, seguido por n pares de valores, um par por linha. O primeiro valor, s , é a velocidade em milhas por hora e o segundo valor, t , é o tempo total gasto. Ambos os valores, s e t são inteiros, $1 \leq s \leq 90$ e $1 \leq t \leq 12$. Os valores de t estão sempre em ordem crescente. Um valor de -1 para n finaliza a entrada.

Saída

Para cada conjunto de entrada, escreva a distância percorrida, seguida por um espaço em branco, seguida pela palavra "milhas" e uma quebra de linha.

Exemplo de Entrada

```
3
20 2
30 6
10 7
2
60 1
30 5
-1
```

Saída para o exemplo de entrada

```
170 milhas
180 milhas
```