



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação
Disciplina: Projeto e Desenvolvimento de Algoritmos
AD2 1º semestre de 2010.

Nome –

Assinatura –

1ª questão (5.0 pontos): Sistema de acompanhamento

Problema

A companhia do metropolitano do Rio de Janeiro pretende iniciar um projeto de acompanhamento com o objetivo de estudar seus indicadores e desenvolver soluções de melhoria.

Para isso, os projetistas do software desenvolveram uma estrutura de dados capaz de organizar as informações de uso do metrô na cidade. Essa estrutura está organizada na forma de uma série de vetores que armazenam os dados dos bilhetes vendidos, como mostrado a seguir:

```
estação[]           // estação onde o bilhete foi usado
dia[]               // dia em que o bilhete foi usado
mês[]               // mês em que o bilhete foi usado
ano[]               // ano em que o bilhete foi usado
hora[]              // hora em que o bilhete foi usado
minuto[]            // minuto em que o bilhete foi usado
segundo[]           // segundo em que o bilhete foi usado
preço[]             // preço de aquisição do bilhete
bilheteria[]        // estação onde o bilhete foi comprado
```

Os vetores são atualizados quando o usuário passa o bilhete pela roleta. Exemplo:

```
estação[85432] ← 'Botafogo'
dia[85432] ← 12
mês[85432] ← 8
ano[85432] ← 2009
hora[85432] ← 9
minuto[85432] ← 16
segundo[85432] ← 33
```

```
preço[85432] ← 2.60  
bilheteria[85432] ← 'Botafogo'
```

Suponha que o próximo bilhete é introduzido na estação Carioca:

```
estação[85433] ← 'Carioca'  
dia[85433] ← 12  
mês[85433] ← 8  
ano[85433] ← 2009  
hora[85433] ← 9  
minuto[85433] ← 16  
segundo[85433] ← 34  
preço[85433] ← 2.60  
bilheteria[85433] ← 'Botafogo'
```

Observações:

- O índice dos vetores começa em um e é incrementado sempre que um novo bilhete é utilizado em uma roleta.
- Suponha a existência de uma variável global **total** que contém o número total de bilhetes registrados no sistema.
- Os dados registrados são armazenados em um banco de dados e estarão disponíveis para o seu programa na forma de variáveis globais (seus algoritmos não precisarão preencher esses vetores, apenas lê-los).
- As estações existentes do metrô são Copacabana, Botafogo, Flamengo, Carioca e Tijuca.

Sua tarefa:

Escreva funções (uma função por item) que consultem a base de dados e exibam os seguintes resultados que serão usados em um relatório gerencial:

1. Qual bilheteria mais vendeu ingressos durante o ano de 2009?
2. Em qual parte do dia (manhã, tarde ou noite) mais pessoas usam o metrô? Considere que o metrô fecha de meia noite às seis horas da manhã.
3. Em que mês de 2009 ocorreu o pico de utilização do metrô?
4. De 1990 a 2009, qual foi o ano em que mais ingressos foram vendidos na bilheteria da estação de Copacabana?
5. Que estação tem o maior movimento no horário das 07 às 11 horas?

2ª questão (5.0 pontos): Números Amigos

Números amigos são duplas de números inteiros em que um é igual à soma dos divisores próprios do outro e vice-versa. Vamos considerar que um número ($dp > 0$) é divisor próprio de um número N quando o resto da divisão de N por dp é zero. Consideraremos também que N não é divisor próprio de si mesmo. Por exemplo, somando-se os divisores de 220 (1, 2, 4, 5, 10, 11, 20, 44, 55, 110) chega-se a 284. Somando-se os divisores de 284 (1, 2, 4, 71, 142) obtém-se 220. Por isso os dois números são amigos. Considere que dois números iguais não são classificados como amigos.

Tarefa

Escreva um programa que receba como entrada um número inteiro L e imprima todos os pares de números amigos entre 1 e L .

Observe no exemplo de entrada e saída abaixo que os pares de números amigos somente aparecem uma vez na resposta.

Exemplo de Entrada e Saída

Limite?

3000

220 amigo de 284

1184 amigo de 1210

2620 amigo de 2924



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

Curso de Tecnologia em Sistemas de Computação
Disciplina: Projeto e Desenvolvimento de Algoritmos
AD2 1º semestre de 2010.
Gabarito

1ª questão (5.0 pontos): Sistema de acompanhamento

função funcao1()

início

para i ← 1 **até** 5 **faça**

 aux[i] ← 0

próximo i

para i ← 1 **até** total **faça**

se ano[i] = 2009 **então**

se bilheteria[i] = 'Copacabana' **então**

 aux[1] ← aux[1] + 1

senão

se bilheteria[i] = 'Botafogo' **então**

 aux[2] ← aux[2] + 1

senão

se bilheteria[i] = 'Flamengo' **então**

 aux[3] ← aux[3] + 1

senão

se bilheteria[i] = 'Carioca' **então**

 aux[4] ← aux[4] + 1

senão

 aux[5] ← aux[5] + 1

fim se

fim se

fim se

fim se

fim se

próximo i

 maior ← 0

para i ← 1 **até** 5 **faça**

se aux[i] > maior **então**

 maior ← aux[i]

 j ← i

fim se

próximo i

 estação[1] ← 'Copacabana'

 estação[2] ← 'Botafogo'

 estação[3] ← 'Flamengo'

 estação[4] ← 'Carioca'

 estação[5] ← 'Tijuca'

resultado ← estação[j]

fim

```

função funcao2()
início
    para i ← 1 até 3 faça
        aux[i] ← 0
    próximo i

    para i ← 1 até total faça
        se hora[i] >= 6 E hora[i] <= 11 então
            aux[1] ← aux[1] + 1
        senão
            se hora[i] >= 12 E hora[i] <= 17 então
                aux[2] ← aux[2] + 1
            senão
                se hora[i] >= 18 então
                    aux[3] ← aux[3] + 1
                fim se
            fim se
        fim se
    próximo i

    maior ← 0
    para i ← 1 até 3 faça
        se aux[i] > maior então
            maior ← aux[i]
            j ← i
        fim se
    próximo i

    período[1] ← 'manhã'
    período[2] ← 'tarde'
    período[3] ← 'noite'

    resultado ← período[j]
fim

```

```

função funcao3()
início
    para i ← 1 até 12 faça
        aux[i] ← 0
    próximo i

    para i ← 1 até total faça
        se ano[i] = 2009 então
            aux[mês[i]] ← aux[mês[i]] + 1
        fim se
    próximo i

    maior ← 0
    para i ← 1 até 12 faça
        se aux[i] > maior então
            maior ← aux[i]
            j ← i
        fim se
    próximo i

    mês[1] ← 'Janeiro'
    mês[2] ← 'Fevereiro'
    mês[2] ← 'Março'
    mês[2] ← 'Abril'
    mês[2] ← 'Maio'
    mês[2] ← 'Junho'
    mês[2] ← 'Julho'
    mês[2] ← 'Agosto'
    mês[2] ← 'Setembro'
    mês[2] ← 'Outubro'
    mês[2] ← 'Novembro'
    mês[2] ← 'Dezembro'

    resultado ← mês[j]
fim

```

```

função funcao4()
início
    para i ← 1 até 20 faça
        aux[i] ← 0
    próximo i

    para i ← 1 até total faça
        se ano[i] >= 1990 E ano[i] <= 2009 então
            aux[ano[i]-1989] ← aux[ano[i]-1989] + 1
        fim se
    próximo i

    maior ← 0
    para i ← 1 até 20 faça
        se aux[i] > maior então
            maior ← aux[i]
            j ← i
        fim se
    próximo i

    resultado ← 1989 + j
fim

```

```

função funcao5()
início
    para i ← 1 até 20 faça
        aux[i] ← 0
    próximo i

    para i ← 1 até total faça
        se hora[i] >= 7 E hora[i] <= 10 então
            se estacao[i] = 'Copacabana' então
                aux[1] ← aux[1] + 1
            senão
                se estacao[i] = 'Botafogo' então
                    aux[2] ← aux[2] + 1
                senão
                    se estacao[i] = 'Flamengo' então
                        aux[3] ← aux[3] + 1
                    senão
                        se estacao[i] = 'Carioca' então
                            aux[4] ← aux[4] + 1
                        senão
                            aux[5] ← aux[5] + 1
                        fim se
                    fim se
                fim se
            fim se
        próximo i

    maior ← 0
    para i ← 1 até 5 faça
        se aux[i] > maior então
            maior ← aux[i]
            j ← i
        fim se
    próximo i

    estação[1] ← 'Copacabana'
    estação[2] ← 'Botafogo'
    estação[3] ← 'Flamengo'
    estação[4] ← 'Carioca'
    estação[5] ← 'Tijuca'

    resultado ← estação[j]
fim

```


2ª questão (5.0 pontos): Números Amigos

função somaDiv (**entradas:** n)
início

```
soma ← 0
para i ← 1 até n/2 faça
    se n mod i = 0 então
        soma ← soma + i
    fim se
próximo i
resultado ← soma
```

fim

início

```
imprima 'Limite? '
leia L
para i ← 1 até L faça
    para j ← i+1 até L faça
        se (somaDiv(i) = j) E (somaDiv(j) = i)
            E (i <> j) então
                imprima i, ' amigo de ', j
            fim se
        próximo j
    próximo i
```

fim