



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação**  
**Disciplina: Projeto e Desenvolvimento de Algoritmos**  
**AD2 2º semestre de 2010.**

**Nome –**

**Assinatura –**

**1ª questão (5.0 pontos): Quem ganhou na Fracasena?**

Em Pindorama, um país que fica ao norte do sul, o jogo é proibido. Os pindoramianos só podem jogar na Fracasena, Ultrasena, Hiperquina, Duplaalgumacoisa, Loteria mais que esportiva e outros quinhentos e trinta e um jogos, que em Pindorama são organizados pelo governo. Naturalmente, o governo prende quem joga nos jogos dos outros. Na Fracasena o jogador deve acertar seis números, entre 1 e 66, sorteados semanalmente e cada apostador somente pode escolher seis números. A sua tarefa é escrever um programa que descubra os jogadores que ganharam na Fracasena.

O formato da entrada e saída dos dados foi definida pelo contratante e deve ser seguida ao pé da letra. Os detalhes deste formato estão definidos a seguir.

**Entrada:**

A entrada dos dados é composta de várias linhas. Na primeira linha temos 6 números, em ordem crescente, entre 1 e 66, que correspondem aos números sorteados, isto é, o resultado da Fracasena. Em seguida temos um número inteiro  $N > 0$  indicando a quantidade de cartões que foram apostados durante a semana. Em seguida o programa deve ler  $N$  linhas com 6 números em ordem crescente, que são as apostas de cada jogador. Considere que todos os números fornecidos ao programa estarão dentro do padrão. Por exemplo, nunca haverá um número  $N$  menor ou igual a 0.

**Saída:**

A saída informa quem foram os apostadores ganhadores. A identificação do jogador é o número correspondente à sua ordem na lista. Portanto, cada jogador da lista é identificado por um número que começa com 1 e vai até  $N$ .

**Exemplo de uma possível entrada e saída de dados:**

Entre com os números sorteados:

1 3 10 20 30 45

Entre com a quantidade de cartões apostados:

5

Entre com os números do jogador 1:

1 3 10 20 30 45

```
Jogador 1 acertou.  
Entre com os números do jogador 2:  
1 3 10 20 30 44  
Entre com os números do jogador 3:  
2 4 6 8 10 12  
Entre com os números do jogador 4:  
1 3 10 20 30 45  
Jogador 4 acertou.  
Entre com os números do jogador 5:  
20 30 40 50 60 66
```

## 2ª questão (5.0 pontos): Um planejador de vôos

Sua tarefa nesse exercício é escrever um programa que permita ao usuário planejar uma rota de ida e volta passando por várias cidades.

A Figura 1 mostra o programa em execução em uma possível implementação do mesmo.

Em sua solução, considere a existência de uma função **getVoos(cidade)** que retorna um vetor contendo todos os destinos a partir de uma cidade origem. A lista de cidades destino termina com a string **'NoNo'**, para indicar o fim da lista. Por exemplo, para imprimir todos os vôos existentes a partir da cidade de São Paulo, faríamos:

```
destinos ← getVoos('São Paulo')  
i ← 1  
enquanto destinos[i] <> 'NoNo' faça  
    imprima destinos[i]  
    i ← i + 1  
fim enquanto
```

Quando chamado com uma string vazia quando argumento, o método **getVoos()** retorna a lista de cidades cobertas pela companhia aérea. Exemplo:

```
imprima 'Lista de cidades atendidas pela companhia:'  
cidades ← getVoos('')  
i ← 1  
enquanto cidades[i] <> 'NoNo' faça  
    imprima cidades[i]  
    i ← i + 1  
fim enquanto
```

Seu programa deve:

- Mostrar a relação de todas as cidades atendidas pela companhia aérea.
- Permitir ao usuário selecionar a cidade de origem da viagem.
- Dentro de um laço, imprimir todas as cidades para onde o usuário pode voar a partir da cidade atual e pedir a ele que escolha a próxima cidade. Considere que ao escolher a próxima cidade, o usuário sempre escolhe uma das cidades da lista apresentada.

- Uma vez que o usuário tenha escolhido uma rota circular (isto é, o usuário escolheu uma cidade que o leva de volta à cidade de origem), o laço termina e o programa imprime a rota escolhida.

Você precisará, de alguma forma, armazenar as cidades escolhidas pelo usuário à medida que ele monta seu plano de voo.

```
Bem-vindo ao Planejador de Voos!
Aqui está a lista de cidades em nossa base de dados:
Belo Horizonte
Brasília
Curitiba
Fortaleza
Natal
Porto Alegre
Porto Seguro
Recife
Rio de Janeiro
São Paulo
Salvador
Vamos planejar uma viagem ida e volta!
Escolha a cidade de origem: Natal
De(o) Natal você tem voos diretos para:
Belo Horizonte
Brasília
Curitiba
Fortaleza
Porto Alegre
Rio de Janeiro
São Paulo
Para onde você quer ir? Curitiba
De(o) Curitiba você tem voos diretos para:
Belo Horizonte
Brasília
Fortaleza
Natal
Porto Alegre
Recife
Rio de Janeiro
São Paulo
Salvador
Para onde você quer ir? Belo Horizonte
De(o) Belo Horizonte você tem voos diretos para:
Brasília
Curitiba
Fortaleza
Natal
Recife
Rio de Janeiro
São Paulo
Salvador
Para onde você quer ir? Natal
A rota que você escolheu é:
Natal
Curitiba
Belo Horizonte
Natal
```

Figura 1. Um exemplo de execução do programa



Fundação CECIERJ - Vice Presidência de Educação Superior a Distância

**Curso de Tecnologia em Sistemas de Computação**  
**Disciplina: Projeto e Desenvolvimento de Algoritmos**  
**AD2 2º semestre de 2010.**  
**Gabarito**

**1ª questão (5.0 pontos): Quem ganhou na Fracasena?**

**início**

```
    imprima 'Entre com os números sorteados.'  
    para i ← 1 até 6 faça  
        leia sorteados[i]  
    próximo i  
    imprima 'Entre com a quantidade de cartões jogados.'  
    leia quantidade  
    para i ← 1 até quantidade faça  
        acertos ← 0  
        imprima 'Entre com os números do jogador ', i  
        para j ← 1 até 6 faça  
            leia jogou  
            se jogou = sorteados[j] então  
                acertos ← acertos + 1  
            fim se  
        próximo j  
        se acertos = 6 então  
            imprima 'Jogador ', i , ' acertou.'  
        fim se  
    próximo i  
fim
```

## 2ª questão (5.0 pontos): Um planejador de vôos

**início**

```
imprima 'Bem-vindo ao Planejador de Voos!'
imprima 'Aqui está a lista de cidades em nossa base de dados:'
cidades ← getVoos('')
i ← 1
enquanto cidades[i] <> 'NoNo' faça
    imprima ' ', cidades[i]
    i ← i + 1
fim enquanto
imprima 'Vamos planejar uma viagem ida e volta!'
imprima 'Escolha a cidade de origem: '
leia cidadeOrigem
numCidades ← 1
rota[numCidades] ← cidadeOrigem
cidadeAtual ← cidadeOrigem
acabou ← falso
enquanto não acabou faça
    destinos ← getVoos(cidadeAtual)
    imprima 'De(o) ', cidadeAtual, ' você tem vôos diretos para: '
    i ← 1
    enquanto destinos[i] <> 'NoNo' faça
        imprima ' ', destinos[i]
        i ← i + 1
    fim enquanto
    imprima 'Para onde você quer ir? '
    leia proximaCidade
    numCidades ← numCidades + 1
    rota[numCidades] ← proximaCidade
    se proximaCidade = cidadeOrigem então
        acabou ← verdadeiro
    fim se
    cidadeAtual ← proximaCidade
fim enquanto

imprima 'A rota que você escolheu é: '
para i ← 1 até numCidades faça
    imprima rota[i]
próximo i
```

**fim**