

**Data:** 10 de novembro de 2015

**Horário limite:** 11:40

Resolva as questões abaixo identificando-as claramente na folha de respostas. Mantenha o silêncio na sala (mantendo desligado aparelhos eletrônicos). A interpretação das questões faz parte da prova.

## Perguntas comuns e suas respostas:

- P: Tenho uma dúvida na questão tal.  
R: A compreensão do enunciado faz parte da prova.
- P: O que será corrigido?  
R: A lógica, a criatividade, a sintaxe, o uso correto dos comandos e dos tipos, os nomes das variáveis, a indentação, uso equilibrado de comentários no código e, evidentemente, a clareza. Nesta prova, você deverá sobretudo escrever códigos modulares, usando corretamente funções e/ou procedimentos, conforme o caso, além de uso correto de variáveis locais ou globais e a passagem de parâmetros por referência ou por valor. Finalmente, sua solução deve fazer uso correto da estrutura de vetores.
- P: Posso fazer a lápis?  
R: Não. A prova deverá ser feita a caneta.
- P: Posso responder na folha de questões?  
R: Não. A prova deverá ser respondida na folha de resposta.

**(20 pontos)** 1) Escreva um programa em linguagem C que faz a leitura de 2 vetores de 10 valores inteiros entre 0 a 255 e determina se estes vetores são parecidos nos seus valores, dado um grau de tolerância. Por exemplo, se o valor de cada posição do segundo vetor lido não varia no máximo em 5, pra mais ou para menos, em relação aos valores das respectivas posições do primeiro vetor, então seu programa deve responder que são parecidos. Caso contrário, seu programa deve responder que são diferentes com x posições discrepantes, como mostram os exemplos de execução.

### Entrada

A entrada possui um único caso de teste. O caso de teste possui exatamente 3 linhas. A primeira linha possui o primeiro vetor de 10 números inteiros  $V_i$ , ( $0 \leq V_i \leq 255$ ), a segunda linha possui o segundo vetor de 10 números inteiros  $V_j$ , ( $0 \leq V_j \leq 255$ ), e a terceira linha possui o grau de discrepância máxima aceita.

### Saída

A saída do seu programa possui uma única linha contendo a palavra “PARECIDOS” caso os vetores não possuam uma diferença maior que a discrepância máxima aceita, ou “DIFERENTES com X discrepantes”, onde X é o número de posições do vetor 2 que possuem uma discrepância maior que a máxima permitida.

### Exemplo de Entrada

```
3 3 3 8 8 8 2 2 2 1
2 3 3 7 9 9 2 2 1 1
3
```

### Saída para o exemplo

```
PARECIDOS
```

### Exemplo de Entrada

```
3 3 3 8 8 8 2 2 2 1
2 3 3 5 9 9 2 2 6 1
1
```

### Saída para o exemplo

DIFERENTES com 2 discrepantes

**(20 pontos)** 2) Uma sequência de  $N$  elementos  $V_i$  ORDENADOS de forma não decrescente, com  $N$  par, é dita balanceada quando a soma do maior elemento com o menor elemento é igual a soma do segundo maior elemento com o segundo menor elemento, que por sua vez é igual a soma do terceiro maior elemento com o terceiro menor elemento, e assim por diante. Escreva uma programa modular em C para ler, a partir do teclado, uma quantidade arbitrária de números inteiros positivos. O número zero é o último número fornecido como dado de entrada e não deve fazer parte dos cálculos. Depois da leitura dos dados, o programa deve verificar e imprimir se os valores lidos formam ou não uma sequência balanceada.

- $2 \leq N \leq 1000$
- $-1000 \leq V_i \leq 1000$

### Exemplo de Entrada

```
1 3 7 11 15 17 0
```

### Saída para o exemplo

Sim, eh balanceada.

### Exemplo de Entrada

```
1 3 5 8 9 10 11 27 32 44 45 51 0
```

### Saída para o exemplo

Nao eh balanceada.

**(60 pontos)** 3) Fazer um programa em C para ler uma sequência (não necessariamente ordenada) com uma quantidade arbitrária de números inteiros positivos do teclado e armazená-los em um vetor  $v$ . O último número lido é o zero, o qual não deve fazer parte dos valores de  $v$ . Além desses, mais um único valor deve ser lido, o que representa o limite de soma. Após a leitura dos dados, o programa deve guardar no vetor  $w$  apenas os últimos números de cada subsequência de  $v$  cuja soma de seus valores ultrapassa minimamente o limite de soma. Uma vez ultrapassado esse limite, uma nova subsequência deve ser iniciada a partir do valor que segue o último da subsequência identificada no momento. Ao final do processamento completo de  $v$ , todos os valores de  $w$  devem ser impressos. Veja um exemplo de execução abaixo:

### Exemplo de Entrada

```
33 51 23 94 66 28 11 73 19 8 31 0
90
```

### Saída para o exemplo

```
23 94 28 19
```

No exemplo acima, o valor 23 é o último da subsequência 33 51 23, cuja soma é 107, a qual minimamente ultrapassa o limite de soma 90, sendo assim copiado para o vetor  $w$ . Com isso, logo depois do valor 23, tem início a verificação da soma dos valores de uma nova subsequência que começa com o número 94. De cara, só o 94 já ultrapassa o limite 90, sendo também copiado para o vetor  $w$ , e assim por diante até que todo o vetor  $v$  seja processado.