

Data: 18 de abril de 2016

Horário limite: 15:40

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.
- 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 respostas.

(9 pontos) 1) Diferencie Lista, Fila e Pilha.

(11 pontos) 2) Mostre a situação da fila F. Inicialmente vazia, após a execução de cada uma das operações a seguir:

```
1 Enfila (&F, 'a');
2 Enfila (&F, 'a');
3 Enfila (&F, 'b');
4 Enfila (&F, 'c');
5 Enfila (&F, Espia (&F));
6 Enfila (&F, Desenfila (&F));
7 Desenfila (&F);
8 Enfila (&F, 'e');
9 Desenfila (&F);
10 Enfila (&F, Espia (&F));
11 Enfila (&F, Enfila (&F, 'g') + 'a');
```

- A função 'Espia' retorna o elemento que seria desenfilado, mas não desenfila o elemento;
- A função 'Desenfila' retorna o elemento desenfilado;
- A função 'Enfila' retorna 1 quando consegue enfileirar e 0 quando não consegue.
 - Você pode assumir que a fila, neste exercício, possui tamanho suficiente para enfileirar todos os elementos.

(30 pontos) 3) 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 função RECURSIVA em C que decida se o vetor está balanceado ou não. Esta função deverá retornar 1 quando estiver balanceado e 0 quando não estiver.

A função tem o seguinte protótipo:

```
int balanceada(int *vetor, int inicio, int fim, int soma)
```

Pode considerar que já na primeira chamada da função o parâmetro `soma` já possui o valor da soma do maior com o menor elemento do vetor.

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

Exemplo de vetor balanceado:

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

Exemplo de vetor não balanceado:

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

(50 pontos) 4) Mais um parser wikipédico

Douglas, um jovem aluno que vive às margens do Rio Tocantins, começou a implementar um novo sistema de Wiki, com a sua própria linguagem de marcação.

Nesta linguagem, quando o usuário desejar que parte do texto fique em negrito o texto deverá ser colocado entre asteriscos, `*`, para colocar em itálico o texto deverá ser colocado entre barras, `/`, e para ficar sublinhado deverá ser colocado entre underscore, `_`.

O mais interessante deste modelo é que um texto pode ficar tanto em negrito quanto itálico, ou sublinhado e itálico, ou qualquer composição com as três configurações. E o mais importante é que a ordem de fechamento da configuração do texto deverá ser o inverso da ordem da configuração aberta (sempre feche a configuração mais recentemente aberta), e não pode sobrar configuração sem fechar no fim do texto. Veja que a Primeira e Terceira cadeias abaixo estão corretas, já a segunda e quarta não estão.

Um texto que começa em negrito e possui um texto em itálico dentro deverá finalizar o itálico antes de fechar o negrito, ao contrário do que a Segunda cadeia faz, e por isso ela está errada.

Primeira cadeia - O Rio `*Tocantins*` é um belo `_rio_`.

Segunda cadeia - Isso `*aqui _nao pode estar*` certo_.

Terceira cadeia - No `_carnaval_` do /Rio de `*Janeiro*/` sempre torço para a `*_/Imperatriz/_*`

Quarta cadeia - As coisas começam a ficar `*/_confusas/_*`

Douglas está em uma fase delicada de seu projeto e precisa que o sistema valide o texto que foi adicionado por um usuário, parece que seus usuários não se atentam muito aos detalhes das configurações, gerando até entradas como a Segunda cadeia acima.

Suponha que uma frase qualquer que possua sequência de letras, pontos, espaços, asteriscos, underscores e barras, e está armazenada em uma cadeia de caracteres (string) `s`. Como é hábito em C, o último caractere da cadeia é o caractere nulo.

Escreva a função `int bem_formada (char *string)` que retorna 1 se a string está bem formada, se não retorna 0. Também implemente quaisquer funções auxiliares para que sua função precise para funcionar.