Departamento de Matemática da Universidade de Coimbra

Métodos de Programação II	Teste Teórico-Prático 3	30/11/2007

Exercício 1

1.a) Descreva a propriedade principal que define uma estrutura de dados *Fila* e descreva, independentemente de qualquer representação ou linguagem de programação escolhida, quais os módulos absolutamente indispensáveis para implementar coerentemente uma tal estrutura, explicando o modo de operação de cada um.

1.b) Usando uma representação de **lista ligada** que lhe pareça mais adequada, implemente um dos módulos acima descritos (não se esqueça de apresentar as declarações necessárias para implementar a representação que escolheu).

Exercício 2

De acordo com as declarações dadas, explique o que está mal no módulo seguinte e proceda às correcções que achar necessárias.

```
typedef struct node{
         int jobNum;  node* seguinte;
        } NO;
typedef struct Qnode{
         node* frente;   node* fim;
        } QNO;

      QNO Q;

      void dequeue (QNO &Q, int &nextJob)
      // Remove o primeiro job num da fila
      { node* temp;

        if( vazia(Q) )
            printf("Fila Vazia!");
        else{
            nextJob = Q.frente -> jobNum;
            temp = Q.fim
            Q.front = frente -> seguinte;
            free(temp);
            if (Q.frente)
               fim = NULL;
         }
      }
```

Exercício 3

3.a) Descreva o algoritmo e descreva as estruturas de dados necessárias para uma função booleana que devolve TRUE se um parâmetro do tipo `string` contém parêntesis e chavetas emparelhados- ( ) [ ] ; Caso contrário, devolve FALSE.

Por exemplo:

- "1 [N](ov)(emb)ro"devolve true
- "3 {Oc[t}ubro]"devolve false

3.b) Implemente essa função com o cabeçalho:

```
Boolean EquilibrioParent(char *str)
{ ... }
```