2. Pretende-se fazer a gestão informática do parque automóvel de uma escola, de tal modo que os lugares livres do parque são representados numa lista ligada, e ficando o parque cheio, os automóveis são colocados em fila de espera, por ordem de chegada, embora os automóveis dos professores tenham prioridade sobre os automóveis dos alunos e funcionários, ou seja, são colocados por ordem de chegada, mas à frente destes.

   Considerando a classe **FichAuto** que guarda informação relativa a um automóvel: matrícula, nome e tipo (prof, aluno ou func):


```
class FichAuto
{
    private:
       string matric;
       string nome;
       string tipo;  //prof,aluno,func

    public:
       FichAuto();
       FichAuto (string m, string n, string t);
       FichAuto (const FichAuto& fa);
       ~FichAuto () ;

       string getMatric() const ;
       string getNome() const ;
       string getTipo() const ;

       void setMatric (string mat) ;
       void setNome (string nom) ;
       void setTipo (string tip) ;

       FichAuto& operator = (const FichAuto& fa) ;
       bool operator == (const FichAuto& fa) const ;
       void escreve (ostream& out) const ;
};
```

a) Defina a classe **Parque** para representar o parque automóvel desta escola, admita que o parque tem CAP lugares livres. Indique apenas os atributos e a assinatura dos métodos necessários ao bom funcionamento do parque acima descrito.

b) Implemente o método checkin o qual existindo um lugar livre no parque insere o automóvel na lista do parque, e caso não exista, coloca-o em fila de espera.

c) Faça um método recursivo para listar todos os automóveis, de um dado tipo, no parque da escola.

isep instituto superior de engenharia do porto

Departamento de Engenharia Informática

Estruturas de Informação

Exame Época Normal 1ª Parte

3 de Fevereiro 2011

Duração 60 min.

Resolva cada exercício em folhas separadas

1. Considere um sistema para gerir os salários dos funcionários de uma empresa. Todos os funcionários possuem salário-dia fixo, código, nome e dias de trabalho efectivos no mês.

   Os gestores têm um prémio de gestão que multiplica o seu vencimento mensal por um factor e os motoristas para além do salário-dia base, têm definido um valor por Km percorrido, e têm a si associado o número de Kms percorridos por mês para cálculo do seu salário final.

   a) Apresente o diagrama de classes.

   b) Defina a classe base, a classe motorista e uma classe para gerir de forma indistinta num vector dinâmico os salários dos funcionários desta empresa, indicando apenas os atributos e a assinatura dos métodos necessários ao bom funcionamento das classes.

   c) Escreva uma função que calcule para um determinado mês o total de salários a pagar aos funcionários motorista.

INSTITUTO POLITÉCNICO DO PORTO

Estruturas de Informação

Exame Época Normal 2ª Parte

3 de Fevereiro 2011

Duração 90 min.

Resolva cada exercício em folhas separadas

1. Dado um vector **ordenado** A, com dim elementos e um valor k, deseja-se encontrar todos os pares de elementos de A cuja soma seja exactamente k. Sabe-se que A tem x pares de elementos.

```
void soma_pares (int A[], int K, int dim)
{
   bool enc=false ;

   for (int i = 0; i < (dim-1); i++)
     for (int j = i+1; j < dim; j++)
        if (A[i]+A[j] == K)
        {
           enc = true ;
           cout << "pos " << i << " -> " << A[i] << ", pos " << j << " -> " << A[j];
        }

    if (!enc)
         cout << "Par de valores não existe !" << endl ;
}
```

a) Indique se o método é determinístico ou não e analise a sua complexidade temporal T(n). Justifique.

b) Proponha uma implementação mais eficiente.

2. Considere uma fábrica constituída por vários postos de trabalho servidos por um robot. Cada posto de trabalho é identificado por um código e tem associado o tempo que o robot demora a fazer o fornecimento de matéria-prima. Conhece-se também o tempo de percurso do robot entre os vários postos de trabalho.

a) Apresente a definição das classes necessárias à modelação deste problema (não é necessário indicar os métodos das classes).

b) Dado um posto inicial, apresente os vários postos de trabalho alcançáveis pelo robot e o tempo total, de fornecimento e de percurso, do robot pelos vários postos.

3. Considere um sistema de rega disposto em forma de árvore binária em que a quantidade de água disponibilizada no ponto inicial vai repartir-se por todos os canais de tal modo que, quando há uma bifurcação, a quantidade de água é dividida de igual forma por cada um dos canais. Adicione à classe template ArvBinPesq<T> um método que determine a quantidade de água que chega a um dado ponto do sistema.