

Ensino Secundário
Curso Profissional de Técnico de Gestão e Programação de Sistemas Informáticos

Módulo6: Estruturas de Dados Dinâmica

10º PSI2

12 de Junho de 2024

Ficha de Avaliação Sumativa

Nome: Simão Rodrigues

Nº _____

(Vale 30%) Parte Teórico-Prática 72%

1 Distingue estrutura estática de estrutura dinâmica:

estrutura estática é quando temos um tamanho definido, estrutura dinâmica é quando não temos um tamanho definido ou seja só temos o (vetor/array) quando a utilizarmos quisermos

2 Indica uma situação para que na elaboração de um programa terias de utilizar a classe List em vez de ArrayList:

Por exemplo numa lista do Spotify a usuários e classe List Para armazenar músicas caso que usasse o ArrayList ou não esta armazenar músicas em vetores *Porquê??*

3. Preenche os espaços em branco:

Nas estruturas dinâmicas, utilizando a classe "List" a instrução add serve para se inserir um novo elemento no vetor. Paralelamente a instrução remove retira um elemento do vetor utilizando a classe "ArrayList".

Já na estrutura FIFO a instrução poll permite saber o conteúdo do 1º elemento da estrutura, sendo que a instrução push adiciona um elemento na pilha, enquanto a instrução pop remove um elemento na fila.

4. Sendo este o vetor, definido num programa:

```
static List<alugueres> vetor_carros=new List<alugueres>();
```

Preenche os espaços sublinhados, tendo presente que a função seguinte utilizaria este vetor:

```
static void insirir (List<alugueres a>){  
    alugueres a;  
    char resp;  
    int i=0;  
  
    do {  
        Console.WriteLine("Insira o nome do cliente {0} ",(i+1));  
        a.nome_cli=Console.ReadLine();  
        Console.WriteLine("Insira o ano de nascimento {0} ",(i+1));  
        a.ano=int.Parse(Console.ReadLine());  
        Console.WriteLine("Insira o custo do aluguer {0} ",(i+1));  
        a.custo=double.Parse(Console.ReadLine());  
        vetor_carros.Add(a);  
        Console.WriteLine("Deseja inserir um novo aluguer? ");  
        resp=char.Parse(Console.ReadLine());  
        i++;  
        Console.WriteLine();  
    } while(resp=="S" || resp=='s');  
}
```


Nome: Simão Rodrigues

Nº 10

1 - Elabore um programa (utilizando funções com os nomes que figuram nas alíneas) que permita ao utilizador (vendedor de seguros) realizar as seguintes tarefas:

- Private List?? Program?? O que é??*
- a) Registe um cliente novo (nome, a localidade, o telefone, nº da apólice, custo, data do contrato) (função `registar_novo`); *↳ Dados pedidos fora da função??*
- b) Apresentar o nome dos clientes que fizeram seguros numa certa data (dia, mês, ano) (função `seg_data`); *↳ Não funciona - esse registo é datatime*
- c) Apresentar a percentagem de clientes pertencentes a uma certa localidade, introduzida pelo utilizador (função `perc_loc`);
- d) Calcular o total dos seguros vendidos considerando todos os clientes (função `total_seg`);
- e) Listar todos os dados do nome do cliente, sendo introduzido o seu número da apólice (função `list_apolice`);
- f) Listar todos os dados de todos os registos por ordem alfabética do nome do cliente (função `list_todos`); *↳ Order by??*

2 - Pretende-se que desenvolva um programa, que simule uma **fila de espera** de uma impressora, utilizando uma fila de **strings**, em que cada string corresponderá ao nome do documento.

O programa deverá implementar as seguintes tarefas:

- 1 – Enviar documento para a fila de espera;
- 2 – Imprimir próximo documento (informa o utilizador que está a imprimir o doc com o respetivo nome);
- 3 – Saber o número de documento em espera;
- 4 – Sair.

O menu com as opções deve ser sempre apresentado até que o utilizador escolha a opção 4 (Sair).

Nota: Não é permitido o uso da internet, nem Guarda os 2 programas, no teu ambiente de trabalho e espera que o professor faça a sua recolha.