

p-Fólio Recurso

U.C. (21093) – Licenciatura em Engenharia Informática

Programação por Objetos

29 de Setembro de 2019

-- INSTRUÇÕES --

- O estudante deverá responder à prova na folha de ponto e preencher o cabeçalho e todos os espaços reservados à sua identificação, com letra legível.
- No caso de provas com escolha múltipla, **sem grelha de resposta**, deverá indicar a resposta correcta na folha de ponto, indicando o número da pergunta e a resposta que considera correcta.
- No caso de provas com escolha múltipla, **com grelha de resposta, tabela e/ou espaços para preenchimento**, deverá efectuar as respostas no enunciado, pelo que o mesmo deverá ser entregue ao vigilante, juntamente com a folha de ponto, **não sendo permitido ao estudante levar o enunciado**.
- Verifique no momento da entrega da(s) folha(s) de ponto se todas as páginas estão rubricadas pelo vigilante. Caso necessite de mais do que uma folha de ponto, deverá numerá-las no canto superior direito.
- Em hipótese alguma serão aceites folhas de ponto dobradas ou danificadas.
- Exclui-se, para efeitos de classificação, toda e qualquer resposta apresentada em folhas de rascunho.
- Os telemóveis deverão ser desligados durante toda a prova e os objetos pessoais deixados em local próprio da sala de exame.
- A prova é constituída por **1** página e termina com a palavra **FIM**. Verifique o seu exemplar e, caso encontre alguma anomalia, dirija-se ao professor vigilante nos primeiros 15 minutos da mesma, pois qualquer reclamação sobre defeito(s) de formatação e/ou de impressão que dificultem a leitura não será aceite depois deste período.
- Utilize unicamente tinta azul ou preta.
- Responda às questões de forma clara, sucinta, e apresente todos os cálculos.
- Quando solicitado, apresente ainda uma representação gráfica do resultado final obtido na questão.
- A cotação de cada uma das questões é indicada junto do enunciado da mesma.
- A prova é **SEM CONSULTA**. Todos os elementos necessários à resolução são fornecidos no enunciado.

Duração: 90 minutos

QUESTÃO 1 (12 valores)

Carolina controla em Excel uma folha de cálculo com a sua lista de compras mensal (um para cada mês). Ela regista o nome do produto, a unidade de compra, a quantidade prevista para um mês, a quantidade que efetivamente será comprada e o preço estimado (atualizado todo mês). Ela também incluiu em sua folha de cálculo as colunas "preço máximo já comprado" e "preço máximo a pagar" no mês corrente, onde esta última coluna é calculada a partir da coluna anterior acrescida de 5%. O "preço máximo já comprado" é inserido na folha de cálculo, a partir das compras efetivamente realizadas. Estes dois campos servem para ajuda-la a evitar pagar valores absurdos por algum produto. Além disso, ela calcula o total estimado para aquele mês, no final da tabela.

Por fim, ela mantém uma segunda folha de cálculo para perceber melhor onde os preços são mais em conta (controla 3 supermercados). Nesta ela coloca o preço mais baixo que ela pagou num determinado mês, indicando a que supermercado pertence. Veja a seguir as duas tabelas mantidas nas folhas de cálculo:

Produto	Unidade de compra	Quantidade no mês prevista	Quantidade comprada	Preço estimado	Preço máximo já comprado	Preço máximo a pagar
ARROZ	KG	8	7	1,80	1,90	2,00
FEIJÃO	KG	6	6	2,10	2,00	2,10
CARNE	KG	6	2	8,00	7,50	7,90
AÇÚCAR	KG	3	7,5	1,50	2,00	2,10
...						

Total estimado				150,00
----------------	--	--	--	--------

Produto	Mês de compra	Valor da compra	Supermercado
ARROZ	julho	1,80	AUCHAN
ARROZ	agosto	1,75	PINGO DOCE
ARROZ	setembro	2,05	CONTINENTE
FEIJÃO	julho	2,10	AUCHAN
FEIJÃO	agosto	2,25	PUNGO DOCE

Esta questão será avaliada da seguinte forma:

- Identificar todas as classes que devem ser criadas (basta dar o nome e explicar para que serve, **resumidamente, não é necessário especificar**) (2 pontos)
- Declaração da classe (**apenas o ficheiro .h, detalhado**) para gerir a lista de compras (todos os atributos e métodos necessários, incluindo os *getters* e *setters*) – 3 pontos
- Declaração da classe (**apenas o ficheiros .h, detalhado**) para conter os produtos (todos os atributos e métodos necessários, incluindo os *getters* e *setters*) - 3 pontos
- Definição dos **métodos** necessários para (**ficheiro .cpp, indicar em qual ficheiro deve o método estar inserido e detalhar apenas o método**):
 - Registrar cada item comprado; (2 pontos)
 - Obter o total estimado; (2 pontos)

Não esqueça de incluir os *#includes* necessários nos ficheiros .h.

Escreva com letra legível e adicione comentários onde for necessário para maior clareza!

FIM

a) Classes: HistoricoCompras, ListaCompra, ItemCompra, Produto e Supermercado

b)

```
#include <iostream>
#include <string>
#include <vector>
#include "ItemCompra.h"

namespace std {

class ListaCompra
{
private:
    int mes;
    vector<ItemCompra> vItemCompra;
    vector<ItemCompra>::iterator itItemCompra;
    float totalEstimado;
public:
    ListaCompra();
    virtual ~ListaCompra();
    vector<ItemCompra>::iterator getItItemCompra() const;
    vector<ItemCompra> getItemCompra() const;
    int getMes() const;
    float getTotalEstimado() const;
    void setItItemCompra(vector<ItemCompra>::iterator itItemCompra);
    void setItemCompra(vector<ItemCompra> vItemCompra);
    void setMes(int mes);
    void setTotalEstimado(float totalEstimado);
    int obterTotalEstimado();
};

} /* namespace std */

#endif /* LISTACOMPRA_H_ */
```

c)

```
#include <iostream>
#include <string>

namespace std {

class Produto {
private:
    string nome;
    string descricao;
    string unidade;
    int qtdPrevistoMes;
    float precoMaximoAnterior;
public:
    Produto();
    virtual ~Produto();
    string getDescricao() const;
    string getNome() const;
    float getPrecoMaximoAnterior() const;
    int getQtdPrevistoMes() const;
    string getUnidade() const;
    void setDescricao(string descricao);
    void setNome(string nome);
    void setPrecoMaximoAnterior(float precoMaximoAnterior);
    void setQtdPrevistoMes(int qtdPrevistoMes);
};

}
```

```

        void setUnidade(string unidade);
};

} /* namespace std */
#endif /* PRODUTO_H_ */

d)
ItemCompra::ItemCompra() {
    cout << "Qual é o produto" << endl;
    cin >> this->produto;
    cout << "Qual é a quantidade comprada" << endl;
    cin >> this->qtdEfetivaComprada;
    cout << "Qual é o preço Máximo" << endl;
    cin >> this->precoMaximoMes;
}

int ListaCompra::obterTotalEstimado() {
    int total;

    for (itItemCompra = vItemCompra.begin(); itItemCompra !=
vItemCompra.end(); itItemCompra++)
        total = total + itItemCompra->getQtdEfetivaComprada();

    cout << "Total estimado: " << total << " €" << endl;

    return total;
}

```