

Nome:			Hora:		Data:	
Curso:			Sala:		N.º Mec:	
A	B	C	Total			

Cotações: A – 7,5 valores; B – 7,5 valores; C – 5 valores;

No seu computador existe a pasta EXAMEPROGI. Os nomes dos vários ficheiros criados ao longo da prova deverão ser do tipo 27250*.java, em que 27250 é o número mecanográfico do aluno e * corresponde à pergunta que está a ser resolvida. Por exemplo, o ficheiro correspondente à pergunta A deverá ter o nome 27250A.java.

A

O responsável por um Clube de Natação pretende construir um programa que o ajude a calcular o valor a pagar por um utente no acto da inscrição.

O programa deve pedir ao utilizador a introdução de:

- Qual a modalidade a frequentar (1 para Iniciação e 2 para Aperfeiçoamento).
- O número de horas semanais (de 1 a 10) de prática.
- Se o utente tem familiares a frequentar o clube (0 para indicar que não tem)
- Se é a primeira vez que se inscreve neste clube e o número de mensalidades que pretende pagar.

Para o cálculo do valor a pagar é preciso ter em conta que cada hora de Iniciação custa 4 Euros e cada hora de aperfeiçoamento custa 6 Euros. O valor da inscrição é de 50 Euros para quem vem frequentar a piscina pela primeira vez e de 30 Euros para quem já esteve inscrito. Com a inscrição é obrigatório o pagamento de pelo menos a primeira mensalidade. Havendo familiares a frequentar existe um desconto de 10% do valor da mensalidade para o caso de o utente ter um familiar e de 25% para o caso de dois ou mais familiares. Considere que todos os meses têm 4 semanas.

Após perguntar todos os dados, o programa deve apresentar os cálculos no seguinte formato:

```
Nova Inscrição para o Clube de Natação
Primeira Vez
3 horas de aperfeiçoamento
1 Familiar
```

Item	Quantidade	Valor
Inscrição	1	50,00
Mensalidade	1	64,80
Total		114,80

Guarde o ficheiro criado no seu directório, com o nome de acordo com o estabelecido (nºmecA.java).

B

Escreva um programa em JAVA que analise as notas dos exames práticos num dos laboratórios duma sessão de exames de Programação 1.

O programa deve começar por solicitar as notas dos alunos (entre 0 e 20) até um máximo de 13 alunos. Como critério de paragem de inserção de notas de alunos considera-se a nota 99. De seguida, devem ser impressas as notas pela ordem pela qual foram introduzidas, a menor nota, a maior nota, a média e as notas ordenadas de modo crescente (ver exemplo). No final, o programa deve dar a oportunidade ao utilizador de repetir o processo ou de abandonar o programa.

Para a resolução deste problema apoie-se no seguinte exemplo, respeitando todas as formatações utilizadas.

Exame Pratico de Programacao 1

```
Aluno 1: 2
Aluno 2: 21
```

Classificacao incorrecta!!!

Aluno 2: 12
Aluno 3: 9
Aluno 4: 15
Aluno 5: 10
Aluno 6: 8
Aluno 7: 16
Aluno 8: 11
Aluno 9: 9
Aluno 10: 9
Aluno 11: 12
Aluno 12: 99

Processamento estatístico

Notas introduzidas:

A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11
2	12	9	15	10	8	16	11	9	9	12

Media: 10.27

Nota Minima: 2

Nota Maxima: 16

Notas ordenadas:

A1	A2	A3	A4	A5	A6	A7	A8	A9	A10	A11
2	8	9	9	9	10	11	12	12	15	16

Nota:

Para a resolução deste problema deverá utilizar funções e/ou procedimentos de acordo com a sua percepção. A não utilização de funções e/ou procedimentos será descontado na qualidade de solução apresentada. A utilização de variáveis globais será também fortemente penalizador.

Guarde o ficheiro criado no seu directório, com o nome de acordo com o estabelecido (nºmecB.java).

C

Escreva um programa em JAVA que caracterize um conjunto de pontos em duas dimensões. No máximo podem ser inseridos 40 pontos. Todos os valores devem ser apresentados com 2 casas decimais. O programa deve ter uma interface com o utilizador baseada num menu interactivo semelhante ao seguinte:

Caracterização de pontos em duas dimensões

1 - Inserir pontos
2 - Apresentar os pontos por quadrantes
3 - Procurar ponto mais próximo
S - sair

Opção ->

Na resolução tem de :

1) Usar para representação dos pontos um registo definido da seguinte forma em JAVA:

```
class Ponto{
    double x,y;
}
```

2) Definir uma função que dado um ponto p, retorna o quadrante

Quadrante	X	Y
1	≥ 0	≥ 0
2	< 0	≥ 0
3	< 0	< 0
4	≥ 0	< 0

3) Definir uma função que dados dois pontos p1 e p2 retorna a distância entre os dois

A distância entre um ponto P1 (x1,y1) e P2 (x2,y2) : $\text{distancia}(P1,P2) = \sqrt{(x1-x2)^2 + (y1-y2)^2}$

Opção 2: Apresentar os pontos separados por quadrantes:

```
Distritos:
Pontos do 1 quadrante
-----
( 1.00, 1.00 )

Pontos do 2 quadrante
-----
não existem pontos
...
```

Opção 3: Dado um ponto P, encontrar o ponto mais próximo

O ponto mais próximo é (... , ...) a uma distância de

Guarde o ficheiro criado no seu directório, com o nome de acordo com o estabelecido (nºmecC.java).