

Atenção

- O exame é sem consulta.
- As resoluções das perguntas 2-a), 2-b) e 3 devem ser realizadas em três folhas independentes, devidamente identificadas (nome completo, número de aluno (do Inforestudiante), unidade curricular e a data). A primeira folha (com o canto descartável) é a folha de prova e as outras (duas) folhas são folhas de continuação.
- Deixe 10 linhas em branco no início de cada folha de continuação.
- Na folha de rascunho coloque o nome completo e o número de aluno (do Inforestudiante).
- Para a resolução de cada uma das perguntas deve apresentar apenas a implementação da solução em linguagem C.
- Nota: a correção das questões 2a) e 2b) será efetuada de forma independente. O facto de não responder à questão 2b) não inviabiliza a resposta à questão 2a). Da mesma forma, poderá resolver a questão 2b) sem resolver a questão 2a), considerando a *chamada da função*, de acordo com o *protótipo definido*.

2. a) [3 valores] Implemente uma função (**PontoNoRetangulo**) que verifica se um determinado ponto está dentro ou fora de um dado retângulo. A função devolve **1** se o ponto está dentro do retângulo. A função devolve **0** se o ponto está fora do retângulo. Considere que a função recebe três vetores de números reais (de dimensão dois), correspondentes às coordenadas dos três pontos. O primeiro vetor (**vInfEsq**) contém as coordenadas do vértice inferior esquerdo do retângulo. O segundo vetor (**vSupDir**) contém as coordenadas do vértice superior direito. O terceiro vetor (**ponto**) contém as coordenadas do ponto (que pode estar, ou não, dentro do retângulo).

```
int PontoNoRetangulo(float vInfEsq[], float vSupDir[], float ponto[])
```

2. b) [4 valores] No âmbito da unidade curricular de Empreendedorismo e Inovação, um grupo de alunos do DEIS decidiu organizar um evento desportivo aberto a toda a comunidade, cujo valor angariado reverterá para a sua viagem de finalistas. O evento consistirá num jogo de lançamento de dardos, cujo alvo será uma placa retangular, afixada numa parede, com o logótipo do ISEC (Figura 1). O preço da inscrição no jogo é de 10 € para os participantes, em geral, havendo um desconto de 50% para os estudantes. Essa inscrição dá direito a 15 lançamentos e a um lanche, oferecido pela organização. Sempre que um participante acertar no alvo (em qualquer ponto, dentro da placa), ganha um ponto. No final, o vencedor é o participante que tiver mais pontos, o qual tem direito a um prémio simbólico. Se houver um empate, o vencedor é o participante que jogou em último lugar.

Desenvolva um programa que, para cada participante, pergunte se é estudante (com um desconto de 50%) e, para cada um dos 15 lançamentos, solicite as coordenadas dos pontos (em metros) em que o dardo se fixou. Considere que as coordenadas dos pontos, em que o dardo se fixou, estão circunscritas à zona definida pela parede. Recorrendo à função definida na alínea a), o programa deve calcular o número de pontos obtidos pelo participante. A leitura dos dados termina quando o utilizador indicar que não há mais participantes.

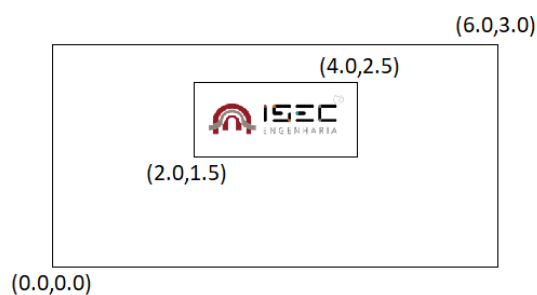


Figura 1 – Placa-alvo na parede.

Na implementação da solução, considere que as coordenadas do canto inferior esquerdo e do canto superior direito da placa retangular (alvo) são, respetivamente, (2.0, 1.5) e (4.0, 2.5), tendo como sistema de referência a parede com as dimensões 6m x 3m (Figura 1).

No final, o programa, deve apresentar o número total de participantes, o valor total angariado e o número (identificador) do participante vencedor.

Exemplo de execução do programa (e da função):

```
Inserir participante (S/N)? S
Participante 1:
Estudante (S/N)? S
Coordenadas do lançamento 1: 2.6 2.1
(...)
Coordenadas do lançamento 15: -1.4 0.5
Coordenadas do lançamento 15: 1.4 0.5
Conseguiu 3 pontos!
Inserir participante (S/N)? S
Participante 2:
Estudante (S/N)? N
Coordenadas do lançamento 1: 3.2 7.9
Coordenadas do lançamento 1: 3.2 2.9
(...)
Coordenadas do lançamento 15: 2.7 2.0
Conseguiu 5 pontos!
(...)
Inserir participante (S/N)? N
-----
Total de participantes = 270
Valor angariado (€) = 1920
Nº do participante vencedor = 68
```

3. [3 valores] Implemente uma função que encontre a maior palavra de uma frase (**frase**). A função deve devolver, por parâmetro, essa palavra (**palavra**) e mostrá-la ao utilizador. Assuma que, na frase, as palavras estão separadas por um espaço e que não existem espaços no início e no final da frase. No caso de existirem palavras com tamanhos iguais deve ser escolhida a primeira, dessas palavras, que aparece na frase. Assuma que a *string* **palavra** tem um comprimento suficiente para conter qualquer palavra.

void maiorPalavra (char frase[], char palavra[])

Exemplo da utilização da função:

```
void main()
{
    char palavra[MAX];
    char fr[] = "Era uma linda coisa anticonstitucionalissimamente pois sim";

    maiorPalavra(fr, palavra);
    printf("\nMaior palavra de: %s => ", fr);
    puts(palavra);
}
```

Exemplo da utilização do programa:

anticonstitucionalissimamente

Maior palavra de: Era uma linda coisa anticonstitucionalissimamente pois sim => anticonstitucionalissimamente