

Função rand()

- P5.1** (a) Utilizando a função `rand()` escrever uma função em C que implemente o jogo **Adivinha o número**. O computador escolhe um número entre 1 e 1000 que terá de ser adivinhado pelo jogador num número máximo de tentativas. Em cada jogada o computador pede ao jogador que introduza um número e depois de o comparar com o que tem guardado, responde indicando se o jogador acertou – caso em que o jogo termina –, se o número é **Maior** ou **Menor** que o número guardado. Enquanto o jogador não adivinha ou número de tentativas não for atingido o jogo continua. No fim, deverá ser indicado se o jogador ganhou, e em quantas tentativas, ou perdeu.
- (b) Implementar o jogo anterior mas sendo o computador a adivinhar o número. Neste caso o computador deverá ser ele a propor um número e o utilizador a indicar se o número em que pensou é: **Maior** (1), **Menor** (-1) ou **Igual** (0). Neste último caso, o computador descobriu o número e ganhou. Nos restantes, casos deve usar a informação dada pelo utilizador, para a próxima tentativa. Sugestão: A melhor estratégia não será o computador escolher um número à sorte, mas sim o valor médio do intervalo em que o número se encontra.
- (c) Escrever um programa em C que faça um interface ao utilizador dos jogos das alíneas anteriores. O programa deve começar por imprimir um menu que permita seleccionar o modo em que se pretende jogar (ser o computador ou o utilizador a adivinhar) ou se se quer terminar o programa. Consoante a tarefa deve ser chamada a função correspondente ou sair do programa.
- P5.2** Escrever um programa em C que ajude a aprendizagem da multiplicação para alunos do 1º ciclo. Usar a função `rand()` para produzir dois inteiros entre 0 e 9, que são mostrados ao aluno (**Quanto é 4 vezes 7?**). O aluno deverá introduzir o resultado, que será comparado com a solução. O programa deverá permitir várias tentativas e de cada vez indicar se o aluno acertou ou errou (e neste caso qual a resposta correcta).

Caracteres

- P5.3** Escreve uma função `char maiuscula(char c)` que tenha como argumento um caracter e que retorne, caso o caracter seja uma letra minúscula, a letra maiúscula correspondente senão o próprio caracter. Escreve uma função análoga à anterior mas que transforme uma letra maiúscula em minúscula.
- P5.4** Escreve um programa que leia do `stdin` uma sequência de caracteres terminada por EOF e, usando funções da biblioteca `ctype` determine a frequência relativa de letras e algarismos.
- P5.5** Escreve um programa que leia do `stdin` uma sequência de caracteres terminada por EOF e a escreva no `stdout` permitindo apenas que cada linha tenha 80 caracteres.
- P5.6** Os bilhetes de avião são identificados por um número inteiro decimal muito grande, por exemplo 47715497443. Para que seja um número válido, o último dígito tem de ser igual ao resto da divisão inteira por 7 do número que resulta excluindo esse dígito. Escreve um programa em C que determine se um dado número é válido. Sugestão: Dado que o número pode ser demasiado grande, não uses o `scanf()`, mas sim lê cada caracter com o `getchar()` e depois transforma no dígito correspondente (p.e '0' corresponde 0, '1' a 1, etc).
- P5.7** Um ficheiro em HTML é constituído por texto anotado com **tags** que são palavras entre os caracteres `< e >`. Escreve um programa que leia do `stdin` e escreva no `stdout`, retirando todas as **tags** de HTML. Por exemplo, o texto seguinte ficaria apenas **Teste Isto é um teste**:
- ```
<html>
<body><h1>Teste</h1><p>Isto é um teste</p></body>
<html>
```