



Exercícios Práticos

Introdução ao C

1. Escrever uma mensagem simples no ecrã, usando a função `printf()`.
2. Declarar uma variável do tipo inteiro (`int`), atribuir-lhe um valor, e escrever uma mensagem no ecrã, que inclua o valor dessa variável.
3. A mesma coisa, mas pedindo o valor ao utilizador (teclado).
4. Pedir ao utilizador um valor em minutos, e escrevê-lo no ecrã, num formato com dias, horas e minutos.
5. Pedir dois números ao utilizador e apresentar o resultado da sua multiplicação no ecrã.
6. Dado (pelo utilizador) o raio da base e a altura, calcular a área da superfície de um cilindro.

Exercícios sobre Condicionais

1. Programa que decompõe um dado número real nas suas duas partes, inteira e fraccionária. Ex 12.345 → 12 e 0.345
2. Pedir um valor ao utilizador e indicar se é positivo ou negativo.
3. Ler dois n.os e apresentá-los por ordem crescente.
4. Calcular o imposto referente a um salário, fornecido pelo utilizador. Se o salário for não superior a zero, avisar; se for maior que 1000, paga 10%, caso contrário, paga 5%.
5. Simular o lançamento de uma moeda — escreve aleatoriamente “Cara” ou “Coroa”.
6. Ler três n.os e apresentá-los por ordem crescente
7. Simular o lançamento de um dado (procurar funções `rand()` e `srand()`).
 - usar `#include` de `<stdlib.h>` e de `<time.h>`;
 - a expressão pode ser `rand()%6+1`;
 - para gerar a semente, usar (p. ex.) `srand(time(0))`;
8. Solicitar salário e um valor de prestação. Se prestação for maior que 20% do salário, imprimir : Empréstimo não pode ser concedido. Senão imprimir Empréstimo pode ser concedido.
9. Escreva um programa que aplique uma taxa de imposto aos solteiros de 10% e 9% aos casados
10. Escreva um programa que mostra depois de pedir uma letra ao utilizador, as 4 letras seguintes do alfabeto
11. Escreva um programa que dado estado Civil do utilizador ('S', 'C','D') escreve o seu estado civil por extenso.
12. Escreva um programa que calcule o imposto pago por mulheres e por homens sabendo que os homens pagam 10% de imposto e que os homens pagam mais 5% que as mulheres.



13. Escreva um programa que verifique se um ano é bissexto ou não.

Exercícios sobre Ciclos

1. Escrever no ecrã "olá" dez vezes.
2. Escrever no ecrã, os 10 primeiros números inteiros, usando o ciclo "while()".
3. Escrever no ecrã, os 20 primeiros múltiplos de 3, usando o ciclo "while()".
4. Idem com o "para"
5. Idem com "repetir"
6. Escrever no ecrã, o somatório de 20 números inteiros, dados pelo utilizador.
7. Escrever no ecrã a tabuada de um n.º pedido ao utilizador.
8. Escrever no ecrã, os 10 primeiros números inteiros, usando o ciclo "for()".
9. Pedir 10 números ao utilizador e apresentar o seu somatório.
10. Pedir ao utilizador uma sequência n.os inteiros positivos. Contar os pares. A sequência termina com a introdução de um 0.
11. Pedir N números ao utilizador e apresentar o seu máximo. N também é pedido ao utilizador.
12. Pedir ao utilizador uma sequência n.os inteiros positivos. Contar os pares. A sequência termina com a introdução de um 0.
13. Apresente os números de Fibonacci inferiores a um dado n.º, pedido ao utilizador. $F_{n+2} = F_{n+1} + F_n$, $F_0=0$ e $F_1=1$ ($n > 0$).
14. Imprimir alfabeto entre duas letras dadas.
15. Pedir ao utilizador dois n.os inteiros; calcular e escrever no ecrã, o máximo divisor comum de dois números
16. Contar o número de dígitos de um número
17. Mínimo múltiplo comum
18. Pedir uma sequência de 30 números ao utilizador, e que no fim apresentar o somatório dos valores aceites. Se o utilizador introduzir dois n.ºs consecutivos iguais, o programa deve rejeitar o segundo e pedir outro n.º em vez desse
19. Implementar o jogo "hi-lo". A finalidade do jogo é que o utilizador acerte num número entre 1 e 100, escolhido aleatoriamente pelo programa. A cada palpite do utilizador, o programa deve responder indicando se este é inferior ou superior ao número escondido. Quando o utilizador acertar, o programa deve dar-lhe os parabéns e indicar em quantas jogadas acertou.



Exercícios sobre Ciclos – Parte 2

1. Apresente uma sequência de N números aleatórios, em que cada $n.^o$ não pode ser igual a nenhum dos dois imediatamente anteriores. N deve ser pedido ao utilizador.
2. Pedir um $n.^o$ inteiro ao utilizador. Calcular e escrever no ecrã, todos os números perfeitos inferiores a esse valor (um $n.^o$ perfeito é aquele que é igual à soma dos seus divisores, excluindo ele próprio).
3. Lançar sequências de 5 dados até que saia uma sequência de dados todos iguais. Contar o número de lançamentos
4. Gerar aleatoriamente seis números de totoloto. O programa deve gerar chaves até que o utilizador diga que já não quer mais. Os números que compõem uma chave não podem ser repetidos.
5. Para cada número inteiro apresentar os seus divisores. Terminar assim que o número total de divisores ultrapasse um dado N dado.
6. Elabore um Algoritmo que calcule o factorial de n .

Exercícios sobre Funções e Vectores

1. Introduzir e executar os seguintes exemplos, disponíveis na bibliografia:
 - a. Um programa que solicite dois números ao utilizador e apresente no ecrã, o resultado da sua soma, e o dobro de cada um deles.
2. Resolver os seguintes exercícios:
 - a. Uma função que receba como parâmetro um dado inteiro, x , e devolva: -1 , se x for negativo; 1 se x for positivo. Um programa que peça x ao utilizador, evoque a função em causa e escreva o resultado no ecrã.
 - b. Uma função que receba como parâmetro um dado inteiro, x , calcule e devolva o seu simétrico. Um programa que peça x ao utilizador, evoque a rotina de cálculo do simétrico e escreva o resultado no ecrã.
 - c. Um procedimento que escreva no ecrã, um carácter, seguido dos n caracteres seguintes a esse, na tabela ascii. Um programa que peça os dados ao utilizador, e os use na evocação da função.
 - d. Uma função que calcule o factorial de um dado inteiro, x ; Um programa que peça x ao utilizador, evoque a rotina de cálculo do factorial e escreva o resultado no ecrã.
 - e. Uma função que calcule o mínimo de cinco números; Uma função que calcule o máximo de cinco números; Uma função que, dados 5 números, calcule a média entre o mínimo e o máximo; Um programa que peça 5 números ao utilizador, e usando aquelas rotinas, apresente no ecrã, a média entre o mínimo e o máximo, dos 5 números introduzidos
3. Introduzir e executar o seguinte exemplo, disponível na bibliografia:
 - a. Pedir ao utilizador os salários mensais de um trabalhador ao longo do ano, armazenando os valores num vector. No fim, apresentar os valores mensais e o total anual.
4. Resolver os seguintes exercícios:



- a. Gerar aleatoriamente 6 n.os correspondentes a bolas de totoloto e armazená-las num vector. Não se preocupe com o facto de as bolas poderem aparecer repetidas.
- b. Dada uma sequência de números (pedida ao utilizador) calcular e apresentar no ecrã, a percentagem de números superiores à média.
- c. No exemplo anterior, acrescentar o cálculo do máximo n.º introduzido.
- d. No exemplo anterior, alterar o valor do máximo no vector para 90% do seu valor original.
- e. Bolas do totoloto sem repetidos
- f. Dado um vector preenchido com números inteiros, trocar os seus elementos, de modo que no fim, todos os valores ímpares, fiquem colocados à esquerda de todos os pares.

5. Vectores

- a. Função que ordene um vector
- b. Função que aceite um inteiro e devolva vector com os dígitos que o compõem.
- c. Algoritmo que permita ao utilizador jogar “Master Mind”; o programa deve gerar aleatoriamente a combinação a descobrir, e depois responder a cada palpite do utilizador, indicando o n.º de elementos certos na casa certa, e o n.º de elementos certos na casa errada. O número de elementos e de cores a usar, deve ser pedido ao utilizador no início do programa (os valores típicos são 4 e 6 respectivamente). O jogador deve ter a possibilidade de desistir! Quando o utilizador acertar na combinação gerada, o programa deve indicar o n.º de jogadas em que o fez. Codifique e corra o programa num computador. O cálculo do numero de “brancas” e de “pretas” devem ser funções independentes.
- d. Faça um programa que permita registar as notas de um conjunto de alunos. O programa deve gerir informação sobre o número e notas de 3 exames. A interacção com o utilizador é efectuada através de um menu com uma lista de opções em que cada opção desencadeia a evocação de um procedimento. As opções são as seguintes:
 - a. acrescentar um conjunto de alunos; b. acrescentar um aluno; c. consultar as notas de um aluno; d. alterar as notas de um aluno; e. listagem por ordem crescente de numero; f. listagem por ordem decrescente de média; g. listagem dos alunos aprovados (média superior a 9,5 e nenhuma inferior a 7,5)



Exercícios sobre Matrizes

1. Somar duas matrizes.
2. Resolver os seguintes exercícios:
 - a. Função que leia uma matriz do teclado
 - b. Função que calcule o mínimo de uma dada linha
 - c. Função que calcule a média de uma dada coluna
 - d. Função que calcule a soma dos elementos da diagonal de uma matriz
 - e. Idem da outra diagonal
 - f. Função que conte o número de ocorrências de um elemento numa matriz
 - g. Função booleana que indique se uma matriz é simétrica
 - h. Função que calcule a transposta de uma matriz
 - i. Função que transponha uma matriz
 - j. Função que calcule a soma de uma matriz quadrada com a sua transposta
 - k. Calcular e apresentar o resultado da multiplicação de duas matrizes dadas pelo utilizador.
 - l. Função que preencha um vector com os somatórios das linhas de uma matriz dada
 - m. Função que preencha dois vectores com os somatórios das linhas e colunas de uma dada matriz
 - n. Função que devolva uma matriz com a lista de coordenadas onde se encontra o valor máximo de uma dada matriz. (+ idem apenas com apenas uma passagem pela matriz)
 - o. Função que “rode” 90º o conteúdo de uma matriz quadrada

Exercícios sobre Apontadores

1. Implementar função "troca" (entre duas variáveis, trocar valor de a e b)
2. Implemente uma função que receba um array de inteiros e que mostre os seus elementos 1 por 1 usando aritmética de apontadores.
3. Dado um vector de inteiros, a função deve calcular o mínimo e o máximo.
4. Implemente um programa usando apontadores para ordenar 5 números e mostra-los no ecrã.
5. Escreva uma função `char * reverseArray(char *array)` que receba um vector de caracteres e possa inverter seu próprio conteúdo. Ex: `v1={'s', 'o', 'l', '\0'}` retornaria `v1= {'l', 'o', 's', '\0'}`.



Exercícios sobre Strings

1. Ler uma String e colocar a primeira letra de cada palavra em maiúsculas
2. Contar o número de palavras de uma frase. Qualquer caracter diferente de letra ou espaço é considerado separador
3. Procurar uma sequência de dois caracteres numa dada *string* sem recorrer às funções de "string.h".
4. Dada o nome de uma pessoa, converter todas as letras para minúsculas, e em maiúsculas todas as primeiras letras de cada palavra
5. Dado o nome de uma pessoa, eliminar todos os "de" "do" "da" e "e" desse nome.
6. Considerando que um *tab* corresponde a tabulações de oito caracteres, substituir todos os espaços que for possível por *tabs*.
7. Função que devolva um string com as palavras de uma dada string ordenadas por ordem alfabética
8. Ler um nome e escrevê-lo na forma "último-nome, restantes-nomes"
9. Construir funções em C para:
 - `size_t strlen(const char *s);`
 - `char *strcpy(char *dest, const char *src);`
 - `char *strncpy(char *dest, const char *src, size_t maxlen);`
 - `char *strcat(char *dest, const char *src);`
 - `char *strlwr(char *s);`
 - `int strcmp(const char *s1, const char *s2);`
 - `int stricmp(const char *s1, const char *s2);`
 - `int strncmp(const char *s1, const char *s2, size_t maxlen);`
 - `int strnicmp(const char *s1, const char *s2, size_t maxlen);`
 - `char *strchr(const char *s, int c);`
 - `char *strstr(const char *string, const char *substring);`

Exercícios sobre memória dinâmica

1. Peça o numero de alunos de uma turma:
 1. Lê as Notas desses alunos (números reais)
 2. Imprimir a média aritmética.
 3. Liberte a memória utilizada no final.
2. Escreva um programa que leia um número inteiro positivo *n* seguido de *n* números inteiros e imprima esses *n* números em ordem invertida. Por exemplo, ao receber 5 222 333 444 555 666 o seu programa deve imprimir 666 555 444 333 222 O seu programa não deve impor limitações sobre o valor de *n*.
3. Escreva um programa que leia uma string contendo o nome do utilizador. Use memoria dinâmica. No final apresente a string lida.



Exercícios sobre recursividade

1. Os números de Fibonacci são definidos da seguinte forma. O primeiro número é 1. O segundo também é 1. O n -ésimo número é definido como sendo a soma dos dois números anteriores.
$$\text{fib}(1) = 1$$
$$\text{fib}(2) = 1$$
$$\text{fib}(n) = \text{fib}(n-1) + \text{fib}(n-2), \text{ para } n > 2$$
 - a. Construa um programa para calcular a sequência de fibonnaci
 - i. Solução Iterativa
 - ii. Solução com recursividade
2. Implementar `int atoi(char *str)` – explicar e implementar
3. Dado um vector de inteiros, a função deve calcular o mínimo e o máximo.
4. Ordene um vector utilizando recursividade.
5. Implementar o `strlen()` recursivamente e sem notação de vectores.
6. Implementar `char *strchr(char *str, char ch)` recursivamente e sem notação de vectores