**Universidade de Trás-os-Montes e**

**Alto-Douro**

# Metodologias de Programação I

Exercícios de Apoio às Aulas Práticas

Versão Daniel Alvim

Cursos:

Engenharia Informática (1º Ciclo)

Tecnologias de Informação e Comunicação (1º Ciclo)

## Expressões básicas

1. Implemente um programa em C que permita somar dois quaisquer números, visualizando o resultado obtido.

#include <stdio.h>

int main() {

int x, y, z;

/\* inserção dos valores a somar \*/ printf(“Insira dois valores inteiros : “); scanf(“%d %d”, &x, &y); /\* cálculo do valor da soma \*/ z = x + y;

/\* apresentação do resultado \*/ printf(“%d + %d = %d\n”, x, y, z);

return 0;

}

#include <stdio.h>

#include <stdlib.h>

int main ()

{

int num1, num2, num3;

printf("Introduza um numero \n");

scanf ("%i",&num1);

printf("Introduza um numero \n");

scanf ("%i",&num2);

num3 = num1 + num2;

printf("A soma de %i com %i e %i \n",num1,num2,num3);

return 0;

}

1. Implemente um programa em C que permita calcular a área de um quadrado. (Área=Lado\*Lado)

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main ()

{

int lado1,lado2,area;

printf("Introduza o lado do quadrado \n");

scanf("%i",&lado1);

printf("Introduza o lado do quadrado \n");

scanf("%i",&lado2);

area = lado1 \* lado2;

printf (" A area do quadrado com os lados %i e %i e %i",lado1,lado2,area);

return 0;

}

1. Implemente um programa em C que permita calcular o perímetro de um quadrado.

#include <stdlib.h>

#include <stdio.h>

#include <math.h>

int main ()

{

int lado1, perimetro;

printf (" Introduza o lado do quadrado \n");

scanf("%i",&lado1);

perimetro = lado1 \* 4;

printf ("O perimetro do quadrado e de %i ",perimetro);

return 0;

}

1. Implemente um programa em C que calcule o volume (V) de um cilindro através dos valores da altura (h) do cilindro e do raio (r) da base.

O volume é calculado por V=h\*PI\*r^2.

Considere o valor de PI=3.141592654

#include<stdio.h>

int main()

{

float altura, raio, volume;

float PI = 3.141592654;

printf("Insira a Altura: ");

scanf("%f", &altura);

printf("Insira o Raio: ");

scanf("%f", &raio);

volume = (altura \* PI \* (raio\*raio));

printf("O Volume e: %f \n", volume);

return 0;

}

1. Implemente um programa em C para ler a base e a altura de um triângulo.

Em seguida, escreva a área do mesmo.

Obs: Área = ( Base \* Altura ) / 2

#include <stdlib.h>

#include <stdio.h>

int main ()

{

float base, altura,area;

printf("Intruduza a base \n");

scanf("%f",&base);

printf("Intruduza a altura \n");

scanf("%f",&altura);

area = ((base \* altura) / 2);

printf (" A area do triangulo com base %f e altura %f e %f",base,altura,area);

return 0;

1. O preço de um automóvel é calculado pela soma do preço de fábrica com o preço dos impostos (45% do preço de fábrica) e a percentagem do revendedor (28% do preço de fábrica). Implemente um programa em C que leia o nome do automóvel e o preço de fábrica e que imprima o nome do automóvel e o preço final.

#include <stdio.h>

#include <string.h>

int main ()

{

char carro[256];

float pfabrica,pimpostos,prevendedor,pfinal;

printf ("Introduza o nome do carro \n");

scanf("%s",&carro);

printf ("Introduza o preco de fabrica \n");

scanf ("%f",&pfabrica);

pimpostos = pfabrica \* 0.45;

prevendedor = pfabrica \* 0.28;

pfinal = pfabrica + pimpostos + prevendedor;

printf (" O preco final do carro %s e %f",carro,pfinal);

return 0;

}

1. Uma empresa tem para um determinado funcionário uma ficha contendo o nome, número de horas trabalhadas e o número de dependentes de um funcionário.

Considere que:

* + A empresa paga 12 euros por hora e 40 euros por cada dependente.
  + Sobre o salário são feitos descontos de 8,5% para o INSS e 5% para IRS.

Faça um algoritmo para ler o Nome, número de horas trabalhadas e número de dependentes de um funcionário. Após a leitura, escreva qual o Nome, salário bruto, os valores descontados para cada tipo de imposto e finalmente qual o salário líquido do funcionário.

#include <stdlib.h>

#include <string.h>

#include <stdio.h>

#include <math.h>

int main ()

{

float horas,dependentes,salariobruto=0,inns=0,irs=0,salarioliquido=0;

char nome[216];

printf (" Introduza o nome do funcionario \n");

scanf ("%s",&nome);

printf (" Numero de horas que trabalha \n");

scanf ("%f",&horas);

printf (" Numero de dependentes do funcionario \n");

scanf ("%f",&dependentes);

salariobruto= 12 \* horas + dependentes \* 40;

inns= salariobruto \* 0.085;

irs= salariobruto \* 0.05;

salarioliquido= ((salariobruto - inns) - irs);

printf ("O funcionario %s tem o salario bruto de %f com o INNS de %f e o IRS de %f fica com um salario liquido de %f",nome,salariobruto,inns,irs,salarioliquido);

return 0;

}

## Estruturas de decisão

8. Implemente um programa em C que peça ao utilizador uma nota de avaliação (numérica) e que indique se o aluno está APROVADO ou REPROVADO. (este exercício pode ser desenvolvido para outras situações como por exemplo ORAL)

#include <stdio.h>

int main() {

float nota;

/\* inserção da nota do aluno \*/ printf(“Insira a nota : “); scanf(“%f”, &nota);

/\* apresentação do resultado \*/ if( nota >= 9.5 ) {

printf(“Aluno Aprovado!\n”);

}

else {

printf(“Aluno Reprovado!”);

}

return 0;

}

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#include <string.h>

int main ()

{

float nota;

printf ("Introduza a nota do Aluno \n");

scanf ("%f",&nota);

if (nota < 9.5)

{

printf ("Reprovado \n");

}

else if (nota > 9.5)

{

printf ("Aprovado \n");

}

else

{

printf ("Oral \n");

}

return 0;

}

9. Elabore um algoritmo que permita calcular a nota Média de um aluno atendendo às notas obtidas nas duas Frequências e visualizar se foi aprovado ou reprovado, tendo em conta que um aluno aprova sempre que a média é superior ou igual a 9,5 valores.

O peso das frequências é de 40% para a 1ª e 60% para a 2ª. Para cada uma delas é obrigatória uma nota mínima de 8,5 valores, sem a qual o aluno está automaticamente reprovado.

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#include <string.h>

int main ()

{

float nota1,nota2,media;

printf (" Introduza a nota da primeira frequencia \n");

scanf ("%f",&nota1);

printf (" Introduza a nota da segunda frequencia \n");

scanf ("%f",&nota2);

if ( nota1 < 8.5 || nota2 <8.5)

{

printf ("O Aluno nao teve a nota minima de 8,5 valores numa das frequencias");

return 0;

}

media = nota1 \* 0.4 + nota2 \* 0.6;

if (media < 9.5 )

{

printf (" Aluno Reprovado com %f \n",media);

}

else if (media > 9.5)

{

printf (" Aluno Aprovado com %f \n", media);

}

else

{

printf (" Vais a Oral que te fodes");

}

return 0;

}

10.Implemente um programa em C que converta um par de valores hora, minutos do formato 24 horas para o formato AM/PM.

Exemplo: 13h07 -> 1h07 PM

00h25 -> 12h25 AM

12h21 -> 12h21 PM

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main ()

{

int hora,minutos;

printf ("Introduza a hora no formato 24 horas \n");

scanf ("%d",&hora);

printf ("Introduza os minutos no formato 24 horas \n");

scanf ("%d",&minutos);

if (hora >12)

{

hora= hora - 12;

printf(" No formato AM/PM fica: %d : %d PM \n",hora,minutos);

}

else if (hora == 12)

{

printf(" No formato AM/PM fica: %d : %d PM \n",hora,minutos);

}

else if (hora == 0)

{

hora = hora +12;

printf(" No formato AM/PM fica: %d : %d AM \n",hora,minutos);

}

else

{

printf(" No formato AM/PM fica: %d : %d AM \n",hora,minutos);

}

return 0;

}

11.Implemente um programa em C que converta um valor em bytes para um formato legível (Kilo, Mega, Giga ou Tera bytes consoante o múltiplo que melhor se adapte a uma representação de fácil leitura do valor).

Considere 1Ki bytes = 1024 bytes

Exemplo: 16548973 bytes = 15,78 Mi bytes

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

//Nao tenho a certeza mas suponho que seja isto;

int main ()

{

float bytes,kilo,mega,giga,tera;

printf ("Introduza o numero de bytes que deseja inserir \n");

scanf("%f",&bytes);

kilo= bytes /1024;

mega= kilo /1024;

giga= mega /1024;

tera= giga /1024;

printf(" O numero de bytes %f são %f Kilo, %f Mega, %f Giga, %f tera", bytes,kilo,giga,tera);

return 0;

}

12.Implemente um programa em C que faça a classificação qualitativa de uma nota (valor inteiro) de um aluno segundo os seguintes níveis:

0 <= nota < 5 : péssimo;

5 <= nota < 8 : mau;

8 <= nota < 10 : insuficiente;

10 <= nota < 12 : suficiente;

12 <= nota < 16 : bom;

16 <= nota <= 20 : excelente;

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main ()

{

int nota;

printf ("Introduza a sua nota \n");

scanf("%d",&nota);

if (nota >=0 && nota < 5)

{

printf ("Péssimo");

}

else if (nota >= 5 && nota < 8 )

{

printf ("Mau");

}

else if (nota >= 8 && nota < 10 )

{

printf ("Insuficiente");

}

else if (nota >= 10 && nota < 12 )

{

printf ("Suficiente");

}

else if (nota >= 12 && nota < 16 )

{

printf ("Bom");

}

else

{

printf ("Excelente");

}

return 0;

}

13.Implemente um programa em C que indique se um dado número inteiro é PAR ou IMPAR.

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main ()

{

int num1;

printf("Introduza um numero inteiro \n");

scanf("%d",&num1);

if ( num1 % 2 == 0)

{

printf ("O numero %i e um numero par",num1);

}

else

{

printf (" O numero %i e um numero impar",num1);

}

return 0;

}

14.Implemente um programa em C que, de entre dois números fornecidos pelo utilizador, permita encontrar o menor deles.

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main ()

{

float num1,num2;

printf (" Introduza o primeiro numero");

scanf ("%f",&num1);

printf (" Introduza o segundo numero");

scanf ("%f",&num2);

if ( num1 > num2)

{

printf (" O numero %f e menor que o numero %f",num2,num1);

}

else

{

printf ("O numero %f e menor que o numero %f",num1,num2);

}

return 0;

}

15.Implemente um programa em C que, de entre três números fornecidos pelo utilizador, permita encontrar o maior deles.

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main ()

{

float num1,num2,num3;

printf (" Introduza o primeiro numero \n");

scanf ("%f",&num1);

printf (" Introduza o segundo numero \n");

scanf ("%f",&num2);

printf (" Introduza o terceiro numero \n");

scanf ("%f",&num3);

if (num1 > num2 && num1 > num3)

{

printf (" O numero %f e maior que %f e que %f",num1,num2,num3);

}

else if (num2 > num1 && num2 > num3)

{

printf (" O numero %f e maior que %f e que %f",num2,num1,num3);

}

else

{

printf (" O numero %f e maior que %f e que %f",num3,num1,num2);

}

return 0;

}

16.Implemente um programa em C que permita, após a inserção de três valores correspondentes a comprimentos de segmentos de recta, verificar se estes podem formar um triângulo.

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main ()

{

float a,b,c;

printf(" Introduza o primeiro lado do triangulo \n");

scanf ("%f",&a);

printf(" Introduza o segundo lado do triangulo \n");

scanf ("%f",&b);

printf(" Introduza o terceiro lado do triangulo \n");

scanf ("%f",&c);

if (b-c < a && a< b+c && a-c < b && b < a + c && a-b < c && c < a+b )

{

printf (" E possivel formar um triangulo");

}

else

{

printf (" Nao é possivel formar um triangulo");

}

return 0;

}

17.Implemente um programa em C que permita, através da inserção da dimensão dos lados de um triângulo, identificar de que tipo de triângulo se trata. Considere apenas as seguintes situações: isósceles – dois lados iguais e um diferente; equilátero – todos os lados iguais; escaleno – todos os lados diferentes.

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main ()

{

float lado1,lado2,lado3;

printf ("Introduza o primeiro lado do triangulo \n");

scanf ("%f",&lado1);

printf ("Introduza o segundo lado do triangulo \n");

scanf ("%f",&lado2);

printf ("Introduza o terceiro lado do triangulo \n");

scanf ("%f",&lado3);

if (lado1 == lado2 && lado1 == lado3)

{

printf("O triangulo e equilatero");

}

else if (lado1 != lado2 && lado1 != lado3 && lado2 != lado3)

{

printf("O triangulo e escaleno");

}

else

{

printf(" O triangulo e isosceles");

}

return 0;

}

18.Implemente um programa em C que permita, através da inserção da dimensão dos lados de um triângulo, verificar se se trata de um triângulo rectângulo.

Sugestão: verificar através do Teorema de Pitágoras efectuando os cálculos com uma aproximação de 0,01.

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

int main ()

{

float a,b,c;

printf ("Introduza a hipotenusa \n");

scanf("%f",&a);

printf ("Introduza o cateto \n");

scanf("%f",&b);

printf ("Introduza o cateto \n");

scanf("%f",&c);

if ( a \*a == b\*b + c\*c)

{

printf ("E um triangulo rectangulo");

}

else

{

printf(" Não e um triangulo rectangulo");

}

return 0;

}

19.Implemente um programa em C que permita converter Km em Milhas. Se o número de Km for superior a 5000 deve também visualizar “muito longe”. 1 milha = 1609 metros;

#include <stdio.h>

#include <stdlib.h>

int main ()

{

float km,milhas;

printf("Introduza a distancia em kilometros \n");

scanf("%f",&km);

milhas = km / 1.609;

if ( km > 5000)

{

printf (" Muito Longe");

}

else

{

printf(" O valor de %f km em milhas e %f",km,milhas);

}

return 0;

}

20.Implemente um programa em C que faça uma conversão entre Euros e Escudos ou vice-versa consoante a preferência do utilizador. O utilizador deve primeiro escolher o tipo de conversão e depois inserir o valor a converter.

#include <stdio.h>

#include <stdlib.h>

int main ()

{

char converter;

float euros,escudos;

printf (" Pretende converter de euros para escudos ou de escudos para euros $/# \n");

scanf("%s",&converter);

switch (converter)

{

case '$':

printf("Introduza o valor em euros \n");

scanf("%f",&euros);

escudos = euros \* 200.482;

printf ("%f euros sao %f escudos \n", euros,escudos);

break;

case '#' :

printf("Introduza o valor em escudos \n");

scanf("%f",&escudos);

euros = escudos / 200.482;

printf ("%f escudos sao %f euros \n", escudos,euros);

break;

}

return 0;

}

## Estruturas de repetição

21.Implemente um programa em C que permita aceitar e visualizar de seguida o nome de 20 pessoas.

|  |
| --- |
| #include <stdio.h>    #define MAX\_NOME 50    int main()  {  char nome[MAX\_NOME]; int i;  for(i=1; i<=20; i++) { /\* inserção de um nome \*/ printf(“Insira o %dº nome : “, i); scanf(“%s”, nome);  /\* visualização do nome inserido \*/  printf(“Foi inserido ‘%s’\n”, nome);  }  return 0;  } |
|  |

#include <stdio.h>

#define MAX\_NOME 50

int main () {

char nome [MAX\_NOME];

int i;

for(i=1;i<=20;i++){

printf("Insira o %d nome: ",i);

scanf("%[^\n]s",nome);

printf("Foi insirido %s \n",nome);

fflush(stdin);

}

return 0;}

22.Implemente um programa em C que permita aceitar um valor numérico que apenas pode ser positivo. Se não for positivo deve insistir na leitura.

#include <stdio.h>

#include <stdlib.h>

int main (void)

{

float numero;

do

{

printf("Introduza um numero \n");

scanf("%f",&numero);

} while (numero<0);

printf ("O numero introduzido e o numero %.2f ",numero);

}

23.Implemente um programa em C que permita encontrar o maior de 50 números inseridos pelo utilizador.

#include <stdio.h>

int main () {

float num,maior;

int i;

maior = -99999999999999999999999;

for (i=1;i<=50;i++){

printf("Insira um numero: ");

scanf("%f",&num);

if (num>maior){

maior = num;}

}

printf("O maior numero inserido foi: %.3f",maior);

return 0;}

24.Implemente um programa em C que leia 30 números inteiros positivos e visualize o menor deles.

#include <stdio.h>

int main () {

float num,menor;

int i;

menor = 99999999999999999999999;

for (i=1;i<=5;i++){

printf("Insira um numero: ");

scanf("%f",&num);

if (num<menor){

menor = num;}

}

printf("O menor numero inserido foi: %.3f",menor);

return 0;}

25. Implemente um programa em C que permita encontrar o maior e menor número de uma série de números positivos fornecidos.

#include <stdio.h>

int main () {

float num,maior,menor;

int i;

maior = -99999999999999999999999;

menor = 99999999999999999999999;

for (i=1;i<=10;i++){

printf("Insira um numero: ");

scanf("%f",&num);

if (num>maior){

maior = num;}

if(num<menor){

menor = num;}

}

printf("O maior numero inserido foi: %.3f\n",maior);

printf("O menor numero inserido foi: %.3f",menor);

return 0;}

26.Dada uma série de 20 valores reais, implemente um programa em C que calcule e escreva a média aritmética destes valores. Entretanto se a média obtida for maior que 20 deverá ser atribuído o valor 20 para a média.

int main(){

float num, media,soma;

int i;

num = 0;

for (i=1;i<=6;i++){

printf("Insira um numero: ");

scanf("%f",&num);

soma = soma + num;

}

media = soma / 6;

if (media >= 20){

media = 20;

}

printf("Media: %.2f",media);

return 0;}

27.Implemente um programa em C que permita efectuar a soma dos 180 primeiros números inteiros positivos.

#include <stdio.h>

int main(){

int i,soma;

soma = 0;

for (i=1;i<=180;i++){

soma = soma + i;

}

printf("%d",soma);

return 0;}

28.Implemente um programa em C que permita calcular a soma dos 20 primeiros números pares positivos.

#include <stdio.h>

#include <stdlib.h>

int main ()

{

int i;

int numero[5],soma=0;

for (i=1;i<=5;i++)

{

printf("Introduza o %d numero \n",i);

scanf("%d",&numero[i]);

if ( numero[i]>0 && numero[i] % 2 == 0)

{

soma = soma + numero[i];

}

}

printf("\n");

printf("Os numeros introduzidos foram: \n");

for (i=1;i<=5;i++)

{

printf("%d \n",numero[i]);

}

printf("\n");

printf(" A soma dos numeros pares positivos e %d",soma);

return 0;

}

29.Implemente um programa em C que repita a mensagem “Introduza a Letra ‘a’” até que se verifique a sua inserção.

#include <stdio.h>

int main(){

char l;

do{

fflush(stdin);

printf("Introduza a letra 'a': ");

scanf("%s",&l);

}while(l!='a');

}

30.Implemente um programa em C que permita calcular o factorial de um número inteiro, tendo em conta que: N! = N\*(N-1)!; 4! = 4\*3\*2\*1=24; 0! = 1

#include <stdio.h>

#include <stdlib.h>

int main ()

{

int i;

int num,fatorial;

printf("Introduza o valor que pretende saber o fatorial \n");

scanf("%d",&num);

fatorial = num;

for (i=1;i<num;i++)

{

fatorial = fatorial \* i;

}

printf("O fatorial de %d e %d",num,fatorial);

}

31.Implemente um programa em C que verifique se um número é ou não primo.

#include <stdio.h>

int main(){

int num, x;

printf("Insira um numero: ");

scanf("%d",&num);

x= num/2;

while (num % x!= 0){

x--;}

if (x==1){

printf("O numero e primo");}

else{

printf("O numero %d nao e primo",num);}

return 0;}

32.Implemente um programa em C que, dado um número natural N, apresente no ecrã os N primeiros números da sequência de *fibonacci*.

|  |  |
| --- | --- |
|  1 ,*se*  *f* (*n*)  1 ,*se*  *f* (*n* 1)  *f* (*n*  2),*se*  | *n* 1  *n*  2 *n*  2 |

#include <stdio.h>

#include <stdlib.h>

int main ()

{

int num,i,fibonacci[1000];

printf("Introduza ate que numero quer a sequencia de fibonacci \n");

scanf("%d",&num);

fibonacci[1] =1;

fibonacci[2] =1;

for(i=3;i<=num;i++)

{

fibonacci[i]= fibonacci[i-1] + fibonacci[i-2];

}

printf(" A sequencia de fibonacci ate ao valor %d e: \n");

printf("1 º numero = %d \n",fibonacci[1]);

printf("2 º numero = %d \n",fibonacci[2]);

for(i=3;i<=num;i++)

{

printf("%d º numero = %d \n",i,fibonacci[i]);

}

return 0;

}

33.Implemente um programa em C para ler base e altura de 50 triângulos e

imprimir a sua área.

#include <stdio.h>

int main(){

float b,h,area;

int i;

for(i=1;i<=50;i++){

printf("Insira a altura: ");

scanf("%f",&h);

printf("Insira a base: ");

scanf("%f",&b);

area = h \* b;

printf("O triangulo %d tem de area: %.2f\n\n",i,area);

}

return 0;}

34.Implemente um programa em C que leia as notas de uma turma de 60 alunos numa disciplina e calcule e apresente a média das notas. O professor pode apenas inserir as notas dos alunos que entender, não sendo imperativo que haja a inserção das 60 notas visto que certos alunos podem faltar ao Exame.

#include <stdio.h>

int main(){

int i,j;

float media,soma,nota;

soma = 0;

printf("Ola Professor!\nQuantas notas prentende inserir?:");

scanf("%d",&i);

for (j=1;j<=i;j++){

printf("Insira uma nota: ");

scanf("%f",&nota);

soma = soma + nota;

}

system("cls");

printf("Inseriu %d notas\n",i);

Media = soma / i;

printf("Media: %.2f",media);

return 0;}

35.Implemente um programa em C que leia os valores anuais de Precipitação de 12 Localidades. Para além disso pretende-se que apresente o valor máximo, o valor mínimo e a média dos valores lidos.

#include <stdio.h>

#include <stdlib.h>

int main ()

{

struct localidade

{

char nome[100];

};

struct localidade nome[12];

int i,j,k;

float precipitacao[12],menor=999,maior=0,soma,media;

for(i=1;i<=12;i++)

{

printf("Introduza o nome da localidade \n");

scanf("%s",&nome[i]);

printf("Valor Anual de precipitacao \n");

scanf("%f",&precipitacao[i]);

soma= soma + precipitacao[i];

if(precipitacao[i] < menor)

{

menor = precipitacao[i];

j=i;

}

if(precipitacao[i] > maior)

{

maior = precipitacao[i];

k=i;

}

}

printf("\n");

for(i=1;i<=12;i++)

{

printf("No mes de %s o valor de precipitacao foi de %.2f \n",nome[i],precipitacao[i]);

}

media= soma /12;

printf("\n");

printf("O mes com menor precipitacao foi %s com %.2f \n", nome[j],menor);

printf("O mes com maior precipitacao foi %s com %.2f \n", nome[k],maior);

printf("A media de todos os meses obtidos e de %.2f \n",media);

return 0;

}

36.Implemente um programa em C que permita contar todos os números pares e ímpares inseridos pelo utilizador aleatoriamente.

#include <stdio.h>

#include <stdlib.h>

int main ()

{

int num,i,j=0,k=0;

int num1;

printf("Quantos numeros deseja introduzir. \n");

scanf("%d",&num);

for(i=1;i<=num;i++)

{

printf("Introduza um numero \n");

scanf("%d",&num1);

if(num1 % 2 == 0)

{

printf("Numero Par \n");

j++;

}

if (num1 % 2 != 0)

{

printf("Numero Impar \n");

k++;

}

}

printf("\n");

printf(" O numero de numeros pares encontrados e %d \n",j);

printf(" O numero de numeros impares encontrados e %d \n",k);

}

37.Implemente um programa em C que calcule a média de uma sequência, de valores reais, terminada por -1. O programa deve apresentar igualmente o número de valores introduzidos.

#include <stdio.h>

int main(){

int num,soma,contagem;

float media;

soma = 0;

contagem = 0;

printf("Insira uma sequencia de numeros para terminar insira -1.\n");

do{

printf("Insira um numero: ");

scanf("%d",&num);

if (num> -1){

soma = soma + num;

contagem = contagem +1;}

}while(num != -1);

media = soma / contagem ;

printf("Media: %.2f",media);

return 0;}

38.Implemente um programa em C que determine o valor máximo de uma sequência de números inteiros positivos (cuja marca de fim é -1) introduzidos sequencialmente através do teclado. A sequência pode ter 0 (só existe a marca de fim) ou mais elementos.

#include <stdio.h>

#include <stdlib.h>

int main ()

{

float num,maior;

do

{

printf("Introduza um numero \n");

scanf("%f",&num);

if (num > maior)

{

maior = num;

}

} while( num!= -1);

printf("O maior numero introduzido foi %.2f",maior);

return 0;

}

39.Implemente um programa em C que mostre dados estatísticos tendo como base um universo de 100 entrevistas realizadas com o objectivo de saber quantas pessoas começaram a trabalhar:

* com menos de 18 anos;
* com mais de 18 anos;
* com menos de 18 anos do sexo masculino;
* com mais de 18 anos do sexo masculino;
* com menos de 18 anos do sexo feminino;
* com mais de 18 anos do sexo feminino;

#include<stdio.h>

#include<stdlib.h>

int main()

{

int i,a=0,b=0,c=0,d=0,e=0,f=0;

int idade[i];

char sexo = 'a';

for(i=1;i<=3;i++)

{

printf("Introduza a idade do entrevistado \n");

scanf("%d",&idade[i]);

fflush(stdin);

printf("Introduza o sexo do entrevistado m/f \n");

scanf("%c",&sexo);

if ( sexo == 'm')

{

if (idade[i] < 18)

{

a++;

}

else

{

b++;

}

}

if ( sexo == 'f')

{

if (idade[i] < 18)

{

c++;

}

else

{

d++;

}

}

}

e=a+c;

f=b+d;

printf("\nTrabalhadores <18: %d", e);

printf("\nTrabalhadores >18: %d", f);

printf("\nTrabalhadores <18 sexo masculino: %d", a);

printf("\nTrabalhadores >18 sexo masculino: %d", b);

printf("\nTrabalhadores <18 sexo feminino: %d", c);

printf("\nTrabalhadores >18 sexo feminino: %d", d);

return 0;

}

40.Implemente um programa em C determine o máximo divisor comum entre dois números inteiros.

#include <stdio.h>

void main(){

int resto,a,b,num1,num2;

printf("Insira um numero: ");

scanf("%d",&num1);

printf("Insira um numero: ");

scanf("%d",&num2);

if (num1>=num2){

a = num1;

b = num2;

}

else{

a = num2;

b = num1;

}

do{

resto = a % b;

a = b;

b = resto;

}while(resto!=0);

system("cls");

printf("O Maximo divisor comum: %d",a);

}

41.Implemente um programa em C determine o mínimo múltiplo comum entre dois números inteiros.

#include <stdio.h>

void main(){

int resto,a,b,num1,num2,mmc,i;

printf("Insira um numero: ");

scanf("%d",&num1);

printf("Insira um numero: ");

scanf("%d",&num2);

for(i=0;i<100;i++){

if (num1>=num2){

a = num1;

b = num2;

break;

}

else{

a = num2;

b = num1;

}

do{

resto = a % b;

a = b;

b = resto;

}while(resto!=0);}

mmc = (num1 \* num2)/a;

system("cls");

printf("Minimo multiplo comum: %d ",mmc);}

42.Implemente um programa em C que converta um número de uma base qualquer (entre 1 e 10) para base 10. Ex: 11 na base 2 é igual a 3 na base 10;

#include <stdio.h>

#include <stdlib.h>

int main()

{

int num=0,base=0,i=0,j=0,final=0;

printf("Introduza o numero que quer converter \n");

scanf("%d",&num);

printf("Introduza a base do numero que quer converter \n");

scanf("%d",&base);

for (i=base;i<10;i++)

{

j++;

}

final= num -j;

printf("O numero %d na base %d e %d na base 10 \n",num,base,final);

return 0;

}

## Funções

43.Implemente um programa que calcule o maior de 3 números reais inseridos pelo utilizador. Implemente para o efeito uma função que calcule/retorne o maior de dois valores reais.

|  |
| --- |
| #include <stdio.h>  float maior(float a, float b)  {  return( a>b?a:b );  }  int main()  { float max, valor; printf(“1º valor = ”); scanf(“%f”, &max); printf(“2º valor = ”); scanf(“%f”, &valor); max = maior(max, valor); printf(“3º valor = ”); scanf(“%f”, &valor); max = maior(max, valor); printf(“O maior valor inserido foi %f\n”, max); return 0;  } |

#include <stdio.h>

#include <stdlib.h>

int Maior(int x, int y){

/\*int maior = 0;

if (x>y){

maior = x;

}else{

maior = y;

}

return maior;

}\*/

if (x>y){

return x;

}else{

return y;

}

}

void main1(void)

{

int a = 0, b = 0, c = 0, maior = 0;

printf("\nInsira 3 numeros: ");

scanf("%d %d %d", &a, &b, &c);

maior = Maior(a, b); //chamada da funçao Maior

maior = Maior(Maior(a, b), c); //chamada da funçao maior

/\*if (maior > c){

maior = maior;

}else{

maior = c;

} //maior = maior(maior,c)\*/

printf("O maior foi: %d", maior);

printf("\n\n"); system("pause");

}44.Implemente um programa em C que permita inserir uma sequência de 100 elementos do tipo inteiro com valores entre 0 e 20.

Utilize uma função que permita ao utilizador inserir um valor inteiro apenas pode ter valor entre 0 e 20. Se tal não acontecer deve insistir na leitura.

#include <stdio.h>

#include <stdlib.h>

int maior (int a)

{

if (a<20 && a>0)

{

printf("O numero introduzido e valido");

}

else

{

printf("O numero introduzido nao e valido");

}

return 0;

}

int main ()

{

int num;

printf("Introduza um numero \n");

scanf("%d",&num);

maior(num);

return 0;

}

45.Implemente um programa em C que permita efectuar somas entre dois *x*

números fraccionários. O resultado deverá ser apresentado no formato: . *y*

Utilize funções para o máximo divisor comum e para o mínimo múltiplo comum.

#include <stdio.h>

#include <stdlib.h>

int mmc(int num2, int num4) {

int mmc, aux, i;

for (i = 2; i <= num4; i++) {

aux = num2 \* i;

if ((aux % num4) == 0) {

mmc = aux;

break;

}

}

return mmc;

}

int main ()

{

int num1,num2,num3,num4,num5,num6,num7,mmc1;

printf ("Introduza o numerador do primeiro numero fracionario \n");

scanf("%d",&num1);

printf ("Introduza o denominador do primeiro numero fracionario \n");

scanf("%d",&num2);

printf ("Introduza o numerador do primeiro numero fracionario \n");

scanf("%d",&num3);

printf ("Introduza o denominador do primeiro numero fracionario \n");

scanf("%d",&num4);

mmc1=mmc(num2,num4);

if (num2==num4)

{

num7= num1 +num3;

}

else

{

num5=num1\*mmc1;

num6=num3\*mmc1;

num7= num5 + num6;

}

printf("A soma de %d / %d com %d / %d e %d / %d",num1,num2,num3,num4,num7,mmc1);

return 0;

}

46.Implemente um programa em C de uma máquina de calcular. Esta deverá ter as seguintes funcionalidades:

1. Um menu de atendimento com todas as funcionalidades da máquina de calcular;
2. As operações Soma, Subtracção, Multiplicação e Divisão;
3. Cálculo do factorial de um número;
4. Verificar se um determinado número é primo;
5. Cálculo da potência de um número;
6. Seno, Co-seno;
7. Calcular o n.º de dígitos de um número;
8. Raiz quadrada e raízes de polinómios de segundo grau;

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#define PI 3.14159265

float primeiro(float a,float b)

{

float c;

char op1;

printf("Introduza a operacao que pretende realizar (+,-,\*,/)) \n");

scanf("%s",&op1);

switch (op1)

{

case '+':

{

c= a + b;

printf (" A soma de %.2f com %.2f e %.2f \n",a,b,c);

break;

}

case '-':

{

c= a - b;

printf (" A subtracao de %.2f com %.2f e %.2f \n",a,b,c);

break;

}

case '\*':

{

c= a \* b;

printf (" A multiplicacao de %.2f com %.2f e %.2f \n",a,b,c);

break;

}

case '/':

{

c= a / b;

printf (" A divisao de %.2f com %.2f e %.2f \n",a,b,c);

break;

}

}

}

float segundo (float a)

{

int i;

float fatorial;

fatorial = a;

for(i=1;i<a;i++)

{

fatorial=fatorial\*i;

}

printf(" O fatorial e %.2f \n",fatorial);

}

int terceiro (int a)

{

int i,j=0;

for (i=2;i<a;i++)

{

if ( a % i == 0)

{

printf("O numero nao e primo \n");

j++;

break;

}

}

if (j==0)

{

printf("O numero e primo \n");

}

return 0;

}

float quarto(float a,float b)

{

int i;

float c;

c=a;

for (i=1;i<b;i++)

{

a=a\*c;

}

printf("O numero %.2f elevado a %.2f e %.2f \n",c,b,a);

}

double quinto (double a)

{

char p;

double result;

printf("Pretende saber o Seno ou Co-seno do numero %f (s/c) \n",a);

scanf("%s",&p);

if(p == 's')

{

result = sin (a\*PI/180);

printf("O seno de %.2f e %.2f",a,result);

}

}

int sexto(int a)

{

if ( a - 99999999 >= 0)

{

printf("O numero tem 9 digitos ");

}

else if ( a - 9999999 >= 0)

{

printf("O numero tem 8 digitos ");

}

else if ( a - 999999 >= 0)

{

printf("O numero tem 7 digitos ");

}

else if ( a - 99999 >= 0)

{

printf("O numero tem 6 digitos ");

}

else if ( a - 9999 >= 0)

{

printf("O numero tem 5 digitos ");

}

else if ( a - 999 >= 0)

{

printf("O numero tem 4 digitos ");

}

else if ( a - 99 >= 0)

{

printf("O numero tem 3 digitos ");

}

else if ( a - 9 >= 0)

{

printf("O numero tem 2 digitos ");

}

else

{

printf("O numero tem 1 digitos ");

}

}

float setimo (float a,float b)

{

float c;

c= pow(a,b);

printf(" A raiz %f de %f e %f \n",a,b,c);

}

int main ()

{

float num1,num2,z;

int num3;

int op;

double num4;

printf("########################## \n");

printf("# # \n");

printf("# Bem Vindo # \n");

printf("# # \n");

printf("# # \n");

printf("########################## \n");

printf("\n");

printf("\n");

printf("Introduza qual a funcionalidade que pretende usar: \n");

printf("\n");

printf("##################################################################################################### \n");

printf("# 1) As operacoes Soma, Subtraccao, Multiplicacao e Divisao. # \n");

printf("# 2) Calculo do factorial de um numero. # \n");

printf("# 3) Verificar se um determinado numero e primo. # \n");

printf("# 4) Calculo da potencia de um numero. # \n");

printf("# 5) Seno, Co-seno. # \n");

printf("# 6) Calcular o numero de digitos de um numero. # \n");

printf("# 7) Raiz quadrada e raizes de polinomios de segundo grau. # \n");

printf("##################################################################################################### \n");

printf("\n");

scanf("%d",&op);

while (op > 1 || op < 7)

{

switch (op)

{

case 1:

{

printf("Introduza o primeiro numero \n");

scanf("%f",&num1);

printf("Introduza o segundo numero \n");

scanf("%f",&num2);

primeiro(num1,num2);

break;

}

case 2:

{

printf("Introduza o numero \n");

scanf("%f",&num1);

segundo(num1);

break;

}

case 3:

{

printf("Introduza o numero \n");

scanf("%d",&num3);

terceiro(num3);

break;

}

case 4:

{

printf("Introduza o numero \n");

scanf("%f",&num1);

printf("Introduza a potencia que quer elevar o numero \n");

scanf("%f",&num2);

quarto(num1,num2);

break;

}

case 5:

{

printf("Introduza o numero \n");

scanf("%lf",&num4);

quinto(num4);

break;

}

case 6:

{

printf("Introduza o numero \n");

scanf("%d",&num3);

sexto(num3);

break;

}

case 7:

{

printf("Introduza o numero \n");

scanf("%f",&num1);

printf("Introduza o grau da raiz \n");

scanf("%f",&num2);

setimo(num1,num2);

break;

}

default :

{

printf("Opcao Invalida, volte a inserir uma opcao \n");

scanf("%d",&op);

}

}

}

return 0;

}

## Vectores

47.Considere um vector V com Tamanho N preenchido com números inteiros. Implemente em código C um algoritmo que apresente no ecrã todos os elementos do vector.

#include <stdio.h>

#define N 10

int main() {

int i, vector[N]={12, 6, 7, 25, 106, 71, 9, 10, 37, 11}; printf(“Elementos\n”); for(i=0; i<N; i++) { printf(“vector[%d]=%d\n“, i, vector[i]);

}

return 0;

}

#include <stdio.h>

#include <stdlib.h>

int main ()

{

int i,x,num[100];

printf("Introduza quantos numeros pretende introduzir no vetor \n");

scanf("%d",&x);

for(i=1;i<=x;i++)

{

printf("Introduza o %d numero \n",i);

scanf("%d",&num[i]);

}

printf("O vector introduzido foi \n");

for(i=1;i<=x;i++)

{

printf("%d º = %d \n",i,num[i]);

}

return 0;

}

48.Considere um vector V com Tamanho N preenchido com números inteiros. Implemente em código C um algoritmo que calcule e apresente no ecrã a soma de todos os elementos do vector com valor ímpar.

#include <stdio.h>

#include <stdlib.h>

int main()

{

int x,i,num[200],soma=0

;

printf("Introduza o numero de elementos que o vetor tenha \n");

scanf("%d",&x);

for(i=1;i<=x;i++)

{

printf("Introduza o %d numero \n",i);

scanf("%d",&num[i]);

if (num[i] % 2 != 0)

{

soma = soma + num[i];

}

}

printf(" A soma dos valores introduzidos impares e %d \n",soma);

return 0;

}

49.Considere um vector V com Tamanho N preenchido com números inteiros. Implemente em código C um algoritmo que calcule e apresente no ecrã a soma de todos os elementos do vector que estejam numa posição de índice ímpar.

#include <stdio.h>

#include <stdlib.h>

int main ()

{

int x,i,num[256],soma=0;

printf("Introduza o numero de elementos que o vetor tem \n");

scanf("%d",&x);

for(i=1;i<=x;i++)

{

printf("Introduza o %d numero \n",i);

scanf("%d",&num[i]);

if (i % 2 != 0)

{

soma = soma + num[i];

}

}

printf(" A soma de todos os elementos do vector que estejam numa posicao de indice impar e %d \n",soma);

return 0;

}

50.Crie um programa em código C que preencha um vector V de N elementos do tipo inteiro. O programa deverá ter pelo menos os seguintes subprogramas:

a) Ler\_Vector

b) Visualizar\_Vector

#include <stdio.h>

#include <stdlib.h>

funcao (int x)

{

int i,num[20

];

for(i=1;i<=x;i++)

{

printf("Introduza o numero \n");

scanf("%d",&num[i]);

}

printf(" O vector e \n");

for(i=1;i<=x;i++)

{

printf("%d \n",num[i]);

}

}

int main ()

{

int x;

printf(" Introduza o numero de elementos que pretende introduzir \n");

scanf("%d",&x);

funcao(x);

}

51.Crie um programa em código C que armazene num vector 10 nomes completos.

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int main ()

{

int i,x,j;

char nome[30][30];

printf("Introduza quantos nomes pretende introduzir \n");

scanf("%d",&x);

for(i=1;i<=x;i++)

{

for(j=1;j<=x;j++)

{

printf("Introduza o %d nome \n",i);

fflush(stdin);

scanf("%[^\n]",&nome[i,j]);

}

}

printf("Os nomes apresentados sao: \n");

for(i=1;i<=x;i++)

{

for(j=1;j<=x;j++)

{

printf("%d nome = %s \n ",i,nome[i,j]);

}

}

return 0;

}

52.Implemente em linguagem C o algoritmo da ordenação por Selecção.

#include <stdio.h>

#include <stdlib.h>

int main()

{

int num[100];

int i,j,minimo,temp,tamanho;

printf("Introduza o numero de elementos do vetor \n",i);

scanf("%d",&tamanho);

for(i=0;i<tamanho;i++)

{

printf("Introduza o %d numero \n",i);

scanf("%d",&num[i]);

}

for(i=0;i<tamanho;i++)

{

printf("Vetor inicial: %d \n",num[i]);

}

printf("\n");

for(i=0;i<tamanho;i++)

{

minimo=i;

for (j=i+1;j<tamanho;j++)

{

if(num[j] < num[minimo])

{

minimo = j;

temp= num[i];

num[i] = num[minimo];

num[minimo] = temp;

}

}

}

for(i=0;i<tamanho;i++)

{

printf("Vetor ordenado: %d \n",num[i]);

}

return 0;

}

53.Implemente em linguagem C o algoritmo da ordenação por Bolha.

#include <stdio.h>

#include <stdlib.h>

int main()

{

int num[100];

int i,j,minimo,temp,tamanho;

printf("Introduza o numero de elementos do vetor \n",i);

scanf("%d",&tamanho);

for(i=0;i<tamanho;i++)

{

printf("Introduza o %d numero \n",i);

scanf("%d",&num[i]);

}

for(i=0;i<tamanho;i++)

{

printf("Vetor inicial: %d \n",num[i]);

}

printf("\n");

for(i=0;i<tamanho;i++)

{

for (j=tamanho;j<i+1;j--)

{

if(num[j] < num[j-1])

{

temp= num[j];

num[j] = num[j-1];

num[j-1] = temp;

}

}

}

for(i=0;i<tamanho;i++)

{

printf("Vetor ordenado: %d \n",num[i]);

}

return 0;

}

54.Implemente em linguagem C o algoritmo da pesquisa linear.

55.Implemente em linguagem C o algoritmo da pesquisa Binária.

56.Modifique o algoritmo de ordenação por bolha realizado anteriormente de forma a conseguir ordenar um *array* de nomes completos.

57.Escreva o código de uma função que aceite como parâmetro de entrada/saída um vector com números reais e a dimensão desse vector. A função deve preencher esse vector com o valor zero. void limpa\_vector(float \*, int tamanho);

#include <stdio.h>

int main(){

int n;

printf("Insira o tamanho do vetor: ");

scanf("%d",&n);

float vetor[n];

limpa\_vector( vetor, n);

return 0;}

void limpa\_vector(float \*vetor, int n){

int i;

for(i=0;i<n;i++){

vetor[i]= 0;

}

}

58.Desenhe um algoritmo e realize a sua implementação em código C que: armazene uma sequência de números inteiros positivos, num *array* dinâmico (cuja marca de fim é -1) introduzidos sequencialmente através do teclado.

#include <stdio.h>

#include <limits.h>

int main(){

int m = INT\_MAX;

int i;

int vet[m];

int \*vetor;

vetor = (int\*)malloc(m\*sizeof(int));

i = -1;

do{

i++;

printf("Insira numeros inteiros reais (-1 para terminar):");

scanf("%d",&vetor[i]);

}while(vetor[i]!=-1);

return 0;}

## Manipulação de *strings*

59.Escreva o código de um programa que peça ao utilizador para introduzir o nome e o sobrenome e que construa e apresente no ecrã uma *string* com o nome completo.

#include <stdio.h>

#include <string.h>

int main() {

char nome[20], sobrenome[20], completo[40]; printf(“Nome = ”); scanf(“%s”, nome); printf(“Sobrenome = ”); scanf(“%s”, sobrenome); strcpy(completo, nome); strcat(completo, “ “); strcat(completo, sobrenome); print(“Nome completo = %s\n”, completo);

return 0;

}

#include <stdio.h>

#include <stdlib.h>

#include <string.h>

int main ()

{

char nome[20];

char sobrenome[20];

char completo[20];

printf("Introduza o nome \n");

scanf("%s",nome);

printf("Introduza o sobrenome \n");

scanf("%s",sobrenome);

strcpy(completo,nome);

strcat(completo," ");

strcat(completo,sobrenome);

printf("Nome completo e: %s",completo);

return 0;

}

60.Escreva o código de um programa que peça ao utilizador para introduzir uma *string* e que apresente no ecrã a *string* escrita “ao contrário”.

Exemplo : “Universidade” deve ser escrito como “edadisrevinU”

#include <stdio.h>

#include <string.h>

int main(){

int n,i;

char nome[100];

printf("Insira o nome: ");

scanf("%[^\n]s",nome);

n = strlen(nome);

for(i=n-1;i>=0;i--){

printf("%c",nome[i]);

}

}

61.Escreva o código de um programa que permita a introdução de uma *string* e que escreva no ecrã a sigla representativa dessa *string*.

Exemplo : “World Wide Web”  “WWW”

#include <stdio.h>

#include <string.h>

int main(){

int n,i,j=0;

char nome[100], sigla[100];

for(i=0;i<100;i++){

sigla[i]='\0';

}

printf("Insira o nome (Maiusculas para a Sigla): ");

scanf("%[^\n]s",nome);

n = strlen (nome);

for(i=0;i<n;i++){

if((nome[i]>='A')&&(nome[i]<='Z')){

sigla[j]= nome[i];

j++;}

}

printf("%s",sigla);

return 0;}

62.Escreva o código de uma função que aceite como parâmetro de entrada uma *string* e que escreva no ecrã a sigla correspondente a essa *string*. A função deve ignorar as sequências “de”, “do”, “da”, “das”, “dos”, “e”.

Protótipo : void sigla(char string[]);

Exemplo : “Universidade de Trás-os-Montes e Alto Douro”  “UTAD”

#include <stdio.h>

#include <string.h>

int main(){

int i;

char nome[100],sigla[100];

for(i=0;i<100;i++){

sigla[i]='\0';

nome[i] = '\0';

}

printf("Insira o nome (Maiusculas para a Sigla): ");

scanf("%[^\n]s",nome);

letras(&nome,&sigla);

system("cls");

printf("%s",sigla);

return 0;}

void letras(char \*nome, char \*sigla){

int i,j=0,n;

n = strlen (nome);

for(i=0;i<n;i++){

if(((nome[i]>='A')&&(nome[i]<='Z')&&(nome[i-1]==' '))|| ( (nome[i]>='A')&&(nome[i]<='Z')&&(i==0) ) ){

sigla[j]= nome[i];

j++;}

}

}

## Manipulação de ficheiros

63.Escreva o código de um programa que escreva no ficheiro “mensagem.txt” o texto “Olá mundo!”.

#include <stdio.h>

int main()

{

FILE \*ficheiro; if((ficheiro=fopen(“mensagem.txt”,”w”))==NULL) {

printf(“Erro de abertura do ficheiro!\n”);

}

else {

fprintf(ficheiro,”Olá mundo!”); fclose(ficheiro);

printf(“Ficheiro produzido com Sucesso!\n”);

}

return 0;

}

#include <stdio.h>

#include <stdlib.h>

int main ()

{

FILE \*ficheiro;

if((ficheiro= fopen("mensagem.txt","w") == NULL)

{

printf("Erro na abertura do ficheiro \n");

}

else

{

fprintf(ficheiro,"Ola Mundo \n");

fclose(p);

printf("Ficheiro produzido com sucesso");

}

return 0;

}

65.Escreva o código de um programa que permita contabilizar o número de linhas de texto que estão armazenadas no ficheiro “dados.txt”.

#include <stdio.h>

#include <stdlib.h>

#define MAX 1000

/\* run this program using the console pauser or add your own getch, system("pause") or input loop \*/

int main() {

FILE \*ficheiro;

int n = 0;

char linha[MAX];

if((ficheiro = fopen("dados.txt","rt"))==NULL){

printf("ERRO!");

}

else{

while((fgets(linha,MAX,ficheiro))!=NULL){

n++;

}

}

printf("O numero de linhas e %d",n);

return 0;

}

66.Implemente um programa em código C que calcule todas as potências de base 2 até ao valor 32767, e armazene as mesmas num ficheiro (Pot2.txt).

67.Implemente um programa em código C converta todos os caracteres minúsculos, de um qualquer ficheiro de texto, em caracteres maiúsculos.

//feito pelo Hugo

#include<stdio.h>

#include<string.h>

#include<math.h>

main()

{

char string[100];

FILE \*ficheiro;

ficheiro=fopen("Pot2.txt", "r",”w”);

while(fgets(string, 100, ficheiro)!=NULL)

{

fprintf(ficheiro,"%s", strupr(string));

}

fclose(ficheiro);

}

68.Implemente um programa em código C que compare o conteúdo de dois ficheiros e que indique no ecrã se os ficheiros são ou não iguais.

//feito pelo Hugo

#include<stdio.h>

#include<string.h>

#include<math.h>

main()

{

char string[100];

char string1[100];

FILE \*ficheiro;

FILE \*ficheiro1;

ficheiro=fopen("Pot2.txt", "r"); //acho q aqui tínhamos q fazer um if !=null

while(fgets(string, 100, ficheiro))

{

printf("%s \n", string);

}

Fclose(ficheiro);

ficheiro1=fopen("daf.txt", "r");

while(fgets(string1, 100, ficheiro1))

{

printf("%s\n\n", string1);

}

Fclose(ficheiro1);

if(string[100] == string1[100]) // aqui podemos e devemos fazer com strcmp

{

printf("Sao iguais");

}

else

{

printf("Sao diferentes");

}

}

69.Implemente um programa que permita gerir as notas dos alunos referentes a uma avaliação de uma turma. Os alunos são identificados através do número mecanográfico e do nome. A nota deve ter um valor do tipo real.

A lista de alunos está armazenada num ficheiro “alunos.txt” com a informação número e nome separados pelo carácter ‘#’ (um registo por linha).

Exemplo:

12480#Laurindinha Ferreira

12576#Jagunço da Silva ...

EOF

O programa incluir a completa gestão da informação tendo um menu de opções com as seguintes tarefas:

1. Carregar a lista de alunos do ficheiro para a memória;
2. Guardar a lista de alunos (apenas os alunos) no ficheiro “alunos.txt”, respeitando o formato dos dados especificado para esse ficheiro;
3. Acrescentar um aluno à lista na memória;
4. Eliminar, à escolha do utilizador, um aluno da lista na memória;
5. Atribuir manualmente as notas aos alunos na memória;
6. Produzir uma pauta em ficheiro com os dados completos de cada aluno. Esse ficheiro, de nome “pauta.txt”, deve conter um registo por linha com os dados separados pelo carácter ‘\*’; Exemplo:

12480\*Laurindinha Ferreira\*14.5

12576\*Jagunço da Silva\*7.9 ...

EOF

1. Apresentar no ecrã a pauta da avaliação, ordenada por ordem alfabética de nome;
2. Sair do programa;

O programa deve usar alocação dinâmica das variáveis de grande dimensão.

## Listas ligadas

70.Desenvolva um programa em código C que implemente uma lista ligada (simples) de números inteiros. O programa deve conter funções de manipulação da lista para:

1. mostrar no ecrã os elementos que estão na lista;
2. inserir um novo elemento no início da lista;
3. inserir um novo elemento no fim da lista;
4. inserir um novo elemento de forma ordenado;
5. eliminar um elemento da lista;
6. destruir a lista;

71.Desenvolva um programa em código C que implemente uma lista de espera de um Centro de Saúde. A implementação da lista de espera deve ser feita recorrendo a uma lista ligada. Para cada elemento da lista deve ser guardado apenas o nome do utente e o número do cartão de utente.

Devem ser implementadas as seguintes funções:

1. mostrar no ecrã os utentes que estão à espera de consulta;
2. atender o próximo utente (primeiro da lista);
3. registar novo utente na lista de espera (fim da lista);

72.Implemente um programa, em linguagem de programação C, que permita ao utilizador gerir os salários mensais de uma empresa. Os dados que constituem a folha de pagamentos, respeitantes a cada funcionários, são o número mecanográfico, o nome, o salário por hora (real) e a quantidade de horas trabalhadas. O programa deve usar listas ligadas (simples) e conter funções para:

1. Preencher uma folha de pagamento com os dados dos funcionários (criar lista);
2. Mostrar a folha de pagamentos no ecrã, indicando o salário total de cada funcionário;
3. Criar um ficheiro “folha.txt” com os dados de todos os funcionários;
4. Efectuar uma pesquisa, na lista ligada, por um funcionário através do número mecanográfico ou através do nome;
5. Alterar (na lista ligada) o número de horas trabalhadas referente a um qualquer funcionário (pesquisado com a função da alínea anterior);