

Introdução à Programação



Professor da UC: Fábio Santos

Trabalho realizado por:

Diogo Silva, nº190204007

Miguel Francisco, nº190204018

Índice

1. Manual Técnico
 - a. Introdução
 - b. Bibliotecas
 - c. Estrutura do Menu
 - d. Funcionalidades do Menu
 - i. Operações com números primos, múltiplos e divisores
 - ii. Vetores e Matrizes
 - iii. Gestão de Contactos
 - iv. Voltar ao Menu Inicial

2. Manual de Utilização
 - a. Introdução
 - b. Funcionamento do menu
 - c. Operações com números primos, múltiplos e divisores
 - d. Vetores e Matrizes
 - e. Gestão de contactos
 - f. Voltar ao Menu Inicial
 - g. Exemplos de Implementação

1. Introdução

Com este documento pretendemos oferecer um guia que explica o funcionamento detalhado e implementação do nosso Projeto.

Separado do Documento do Relatório para facilitar a procura de informação e a organização do projeto no geral.

2. Bibliotecas

As bibliotecas utilizadas foram: `stdio.h`, `stdlib.h` e ainda criamos um `header.h` (cabecalho) para o programa conhecer as funções antes de serem mandadas executadas.

`stdio.h`, o seu nome vem de origem inglesa, de “Standard Input-Output Header”, que em português significa “Cabeçalho padrão de entrada e saída”. Com esta biblioteca é possível executar o programa em código C sem erros, usando funções muito conhecidas, como por exemplo, o `printf` e o `scanf`.

`Stlib.h` é uma biblioteca com funções mais avançadas, pois, é possível alocar memória ou controlar processos, entre outras. A linguagem C++ tem uma biblioteca semelhante chamada de `sctdlib.h`.

```
//Bibliotecas utilizadas no projeto
#include <stdio.h>
#include <stdlib.h>
#include "header.h"
//Usamos um ficheiro header.h para declarar todas as funcoes
```

Figura 1 - Bibliotecas utilizadas no Projeto

3. Estrutura de Menu

Para fazer o menu principal, usamos a condição lógica if (state) e while caso a opção não seja a correta.

```
27 int menu() {
28     char escolha; //Variavel para selecionar a opcao pretendida
29
30     printf("\t===== MENU =====\n");
31     printf("1. Operacoes com numeros primos, multiplos e divisores;");
32     printf("\n2. Vetores;");
33     printf("\n3. Gestao de Contactos;");
34     printf("\n4. Terminar o programa.\n");
35     printf("\n5. Criar um ficheiro de texto\n");
36
37     printf("\nEscolha um numero\n");
38     scanf(" %c", &escolha);
39
40     //Se o caracter escolhido pelo o utilizador não for 1 nem 2 nem 3 nem 4,
41     //o programa entrará num ciclo infinito até o utilizador escrever uma caracter válido
42     while (escolha != '1' && escolha != '2' && escolha != '3' && escolha != '4')
43     {
44         printf("Numero errado, escolhe outro\n");
45         scanf(" %c", &escolha);
46     }
47
48     //Se a opção escolhida for 1, o programa mostra ao utilizador o menu 1
49     if (escolha=='1')
50     {
51         printf("\nFoi escolhida a opcao 1\n\n");
52         menu1();
53     }
54
55     //Se a opção escolhida for 2, o programa mostra ao utilizador o menu 2
56     if (escolha=='2')
57     {
58         printf("\nFoi escolhida a opcao 2\n\n");
59         menu2();
60     }
61
62     //Se a opção escolhida for 3, o programa mostra ao utilizador o menu 3
63     if (escolha=='3')
64     {
65         printf("\nFoi escolhida a opcao 3\n\n");
66         menu3();
67     }
68
69     //Se a opção escolhida for 4, sai do programa
70     if (escolha=='4')
71     {
72         sairDoProg();
73     }
```

Figura 2 - Estrutura de Menu

Manual de Utilização do menu

Após iniciar o programa, o utilizador vai ser recebido com um Menu principal, mostrando várias opções e cabe ao utilizador selecionar a opção que deseja.

Caso o número esteja incorreto, o programa entrará em Loop até ser introduzido um número válido.

```
===== MENU =====
1. Operacoes com numeros primos, multiplos e divisores;
2. Vetores;
3. Gestao de Contactos;
4. Terminar o programa.

Escolha um numero
```

Figura 3 - Output do Menu Principal

Escolhendo as opções 1, 2 ou 3, o utilizador será enviado para outro menu com aplicações do tópico selecionado anteriormente.

Escolhendo a opção 4, o programa termina de imediato.

Exemplo de aplicação: Operações com números primos, múltiplos e divisores -> Opção 3 Mostrar o Somatório de números Primos num intervalo fechado.

```
===== MENU =====
1. Operacoes com numeros primos, multiplos e divisores;
2. Vetores;
3. Gestao de Contactos;
4. Terminar o programa.

Escolha um numero
1

Foi escolhida a opcao 1

1. Operacoes com numeros primos, multiplos e divisores
  -1. Verificar se um numero e primo;
  -2. Mostrar e contar os numeros primos existentes num intervalo fechado;
  -3. Mostrar o somatorio dos numeros primos existentes num intervalo fechado;
  -4. Mostrar o primeiro numero primo que seja maior ou igual a um valor inserido;
  -5. Mostrar o numero primo mais proximo de um valor inserido pelo utilizador;
  -6. Decompor um numero nos seus fatores primos;
  -7. Calcular o maximo divisor comum de dois numeros;
  -8. Calcular o menor multiplo comum de dois numeros;
  -0.Voltar ao menu anterior.

Escolha outra vez outro numero:3

Opcao Escolhida:      3. Mostrar o somatorio dos numeros primos existentes num intervalo fechado;
Introduza o numero minimo do intervalo:
```

Figura 4 - Exemplo de aplicação do menu

Operações com números primos, múltiplos e divisores

Escolhendo esta opção, é mostrado ao utilizador as seguintes opções:

```
1. Operacoes com numeros primos, multiplos e divisores
  -1. Verificar se um numero e primo;
  -2. Mostrar e contar os numeros primos existentes num intervalo fechado;
  -3. Mostrar o somatorio dos numeros primos existentes num intervalo fechado;
  -4. Mostrar o primeiro numero primo que seja maior ou igual a um valor inserido;
  -5. Mostrar o numero primo mais proximo de um valor inserido pelo utilizador;
  -6. Decompor um numero nos seus fatores primos;
  -7. Calcular o maximo divisor comum de dois numeros;
  -8. Calcular o menor multiplo comum de dois numeros;
  -0.Voltar ao menu anterior.
```

Figura 5 - Opções do menu 1

```
78 //Menu 1
79 void menu1()
80 {
81     printf("\n1. Operacoes com numeros primos, multiplos e divisores");
82     printf("\n\t-1. Verificar se um numero e primo;");
83     printf("\n\t-2. Mostrar e contar os numeros primos existentes num intervalo fechado;");
84     printf("\n\t-3. Mostrar o somatorio dos numeros primos existentes num intervalo fechado;");
85     printf("\n\t-4. Mostrar o primeiro numero primo que seja maior ou igual a um valor inserido;");
86     printf("\n\t-5. Mostrar o numero primo mais proximo de um valor inserido pelo utilizador;");
87     printf("\n\t-6. Decompor um numero nos seus fatores primos;");
88     printf("\n\t-7. Calcular o maximo divisor comum de dois numeros;");
89     printf("\n\t-8. Calcular o menor multiplo comum de dois numeros;");
90     printf("\n\t-0.Voltar ao menu anterior.");
91     printf("\nEscolha outra vez outro numero:");
92
93     //Para o menu, utilizamos a função Switch() para selecionar a opção que pretendemos
94     char escolha;
95     scanf(" %c", &escolha);
96     switch (escolha)
97     {
98     case '1':
99     {
100         printf("\nOpcao Escolhida:\t1. Verificar se um numero eh primo;");
101         TesteNumerosPrimos();
102
103         break;
104     }
105     case '2':
106     {
107         printf("\nOpcao Escolhida:\t2. Mostrar e contar os numeros primos existentes num intervalo fechado;");
108         MostrarEContarNumerosPrimosAte();
109         break;
110     }
111
112     case '3':
113     {
114         printf("\nOpcao Escolhida:\t3. Mostrar o somatorio dos numeros primos existentes num intervalo fechado;");
115         SomatorioDeNumerosPrimosNoIntervalo();
116         break;
117     }
118     case '4':
119     {
120         printf("\nOpcao Escolhida:\t4. Mostrar o primeiro numero primo que seja maior ou igual a um valor inserido;");
121         NumeroPrimoIgualAIntroduzido();
122         break;
123     }
```

Figura 6 - Estrutura do código do menu

```

122         break;
123     }
124     case '5':
125     {
126         printf("\nOpcao Escolhida:\t5. Mostrar o numero primo mais próximo de um valor inserido pelo utilizador;");
127         NumeroPrimoMaisPertoDoIntroduzido();
128         break;
129     }
130     case '6':
131     {
132         printf("\nOpcao Escolhida:\t6. Decompor um numero nos seus fatores primos;");
133         Decomposicao();
134         break;
135     }
136     case '7':
137     {
138         printf("\nOpcao Escolhida:\t7. Calcular o maximo divisor comum de dois numeros;");
139         CalcularMDC();
140         break;
141     }
142     case '8':
143     {
144         printf("\nOpcao Escolhida:\t8. Calcular o menor multiplo comum de dois numeros;");
145         CalcularMMC();
146         break;
147     }
148     }
149     case '0':
150     {
151         printf("\nOpcao Escolhida:\t0.Voltar ao menu anterior.\n");
152         break;
153     }
154     }
155 }

```

Figura 7 - Estrutura do código do menu (continuação)

Menu 1, opção 1: Verificar se um número é primo

```
Opcao Escolhida:      1. Verificar se um numero eh primo;
Escolhe um numero maior que 2:
```

Figura 8 - Output do menu 1, opção 1

Este programa tem como objetivo testar se o número introduzido é primo e tem de ser maior do que 2.

É pedido ao utilizador um número para a verificação.

```
156 void TesteNumerosPrimos()
157 {
158     printf("\nEscolhe um numero maior que 2: ");
159     int numero, contador, resultado = 0;
160     scanf("%d", &numero);
161
162     //Verificação se numero introduzido é maior que 2, se não for, o programa pede de novo um número
163     while (numero < 2)
164     {
165         printf("Erro\nIntroduz outro numero: ");
166         scanf("%d", &numero);
167     }
168
169     //Fazer o teste se o numero em que se encontra na variavel "numero" é primo
170     for (contador = 2; contador <= numero / 2; contador++)
171     {
172         if (numero % contador == 0)
173         {
174             resultado++;
175         }
176     }
177
178     //Se o numero for primo
179     if (resultado == 0)
180     {
181         printf("O numero escolhido, que foi o %d, eh numero Primo", numero);
182     }
183
184     //Se o numero não for primo
185     else
186     {
187         printf("O numero escolhido, que foi o %d, nao eh numero Primo", numero);
188     }
189     voltar_menu_inicial();
190 }
191
```

Figura 9 - Código para verificar se um número é primo

Exemplo de Aplicação: Supondo que é inserido o número 9, o Output será:

```
Opcao Escolhida:      1. Verificar se um numero eh primo;  
Escolhe um numero maior que 2: 9  
O numero escolhido, que foi o 9, nao eh numero Primo  
  
      Introduza qualquer caracter para voltar ao menu principal:
```

Figura 10 - Output para verificar se um número é primo

Menu 1, opção 2: Mostrar e contar os números primos existentes num intervalo fechado

```
Opcao Escolhida:      2. Mostrar e contar os numeros primos existentes num intervalo fechado;
Introduza o valor do primeiro numero
```

Figura 11 - Output do menu 1, opção 2

Este programa tem como objetivo mostrar e contar apenas os números primos dentro de um intervalo fechado.

É pedido ao utilizador dois números, um para o valor mínimo e outro para o valor máximo do intervalo que deseja introduzir

```
192 void MostrarEContarNumerosPrimosAte(){
193     int numero1, numero2, teste1, teste2, SerPrimo, ManterNumero1, contador = 0;
194
195     printf("\nIntroduza o valor do primeiro numero\n");
196     scanf("%d", &numero1);
197     printf("Introduza o valor do segundo numero\n");
198     scanf("%d", &numero2);
199     puts("-----");
200
201     //Verificação se o segundo numero introduzido é menor do que dois
202     if(numero2 < 2)
203     {
204         printf("Erro: Nao existem numeros primos ate %d\n", numero2);
205         exit(0);
206     }
207
208     //Caso isso não se verifique
209     printf("\nOs numeros primos sao: \n");
210
211     ManterNumero1 = numero1;
212
213     //Verificação de números primos
214     if( numero1 % 2 == 0)
215     {
216         numero1++;
217     }
218
219     for(teste1 = numero1; teste1 <= numero2; teste1 = teste1 + 2)
220     {
221
222         SerPrimo = 0;
223
224         for(teste2 = 2; teste2 <= teste1 / 2; teste2++)
225         {
226
227             if ((teste1 % teste2) == 0)
228             {
229
230                 SerPrimo = 1;
231                 break;
232             }
```

Figura 12 - Código para Mostrar e contar os números primos existentes num intervalo fechado

```

232
233     }
234 }
235
236 //Cada vez que se verifique um numero primo no intervalo,
237 //é incrementado um valor à variavel contador
238 if(SerPrimo == 0)
239 {
240
241     printf("%d\n", teste1);
242     contador++;
243 }
244 }
245
246 printf("\nEntre %d e %d existem: %d numero/numeros primos\n", ManterNumero1, numero2, contador);
247 puts("-----");
248 voltar_menu_inicial();
249
250 }

```

Figura 13 - Código para Mostrar e contar os números primos existentes num intervalo fechado (continuação)

Exemplo de Aplicação: Supondo que o intervalo fechado é de 2 a 10, o Output será:

```

Opcao Escolhida:      2. Mostrar e contar os numeros primos existentes num intervalo fechado;
Introduza o valor do primeiro numero
2
Introduza o valor do segundo numero
10
-----

Os numeros primos sao:
3
5
7

Entre 2 e 10 existem: 3 numero/numeros primos
-----

Introduza qualquer caracter para voltar ao menu principal:

```

Figura 14 - Output para mostrar e contar os numeros primos existentes num intervalo

Menu 1, opção 3: Mostrar o somatório dos números primos existentes num intervalo fechado

```
Opcao Escolhida:      3. Mostrar o somatorio dos numeros primos existentes num intervalo fechado;
Introduza o numero minimo do intervalo:
```

Figura 15 - - Output do menu 1, opção 3

Este programa tem como objetivo mostrar o somatório de números primos dentro de um intervalo fechado.

É pedido ao utilizador dois números, um para o valor mínimo e outro para o valor máximo do intervalo que deseja introduzir.

```
252 void SomatorioDeNumerosPrimosNoIntervalo()
253 {
254     int NumeroMinimo, NumeroMaximo, Contador, ContadorTeste, SerPrimo, Somatorio=0;
255
256     //Pedir ao utilizador para indicar um numero limite minimo e o numero limite maximo
257     printf("\nIntroduza o numero minimo do intervalo:\n");
258     scanf("%d", &NumeroMinimo);
259     printf("Introduza o numero maximo do intervalo:\n");
260     scanf("%d", &NumeroMaximo);
261
262
263     // Encontrar todos os numeros primos entre o numero minimo e o numero maximo selecionado pelo utilizador
264
265     for(Contador=NumeroMinimo; Contador<=NumeroMaximo; Contador++)
266     {
267
268         //Verificar se o numero dentro da variavel Contador eh primo ou nao
269
270         SerPrimo = 1;
271
272         for(ContadorTeste=2; ContadorTeste<=Contador/2; ContadorTeste++)
273         {
274
275             if(Contador % ContadorTeste == 0)
276             {
277
278                 // o numero no 'Contador' quando nao eh primo
279
280                 SerPrimo = 0;
281                 break;
282
283             }
284
285         }
286
287         // Se o numero dentro da variavel 'Contador' for primo, adicionar ao somatório
288
289         if(SerPrimo == 1)
290         {
291
292             Somatorio = Somatorio + Contador;
```

Figura 16 - Código de Mostrar o somatório dos números primos existentes num intervalo fechado

```

292         Somatorio = Somatorio + Contador;
293     }
294
295
296 }
297
298 //Dar print ao resultado do somatorio tendo em conta os valores selecionados
299
300 puts("-----");
301 printf("O somatorio de todos os numeros primos entre %d e %d eh: %d", NumeroMinimo, NumeroMaximo, Somatorio);
302 voltar_menu_inicial();
303 }
304

```

Figura 17 - Código de Mostrar o somatório dos números primos existentes num intervalo fechado (continuação).

Exemplo de Aplicação: Supondo que o intervalo fechado é de 2 a 10, o Output será:

```

Opcao Escolhida:      3. Mostrar o somatorio dos numeros primos existentes num intervalo fechado;
Introduza o numero minimo do intervalo:
2
Introduza o numero maximo do intervalo:
10
-----
0 somatorio de todos os numeros primos entre 2 e 10 eh: 17

      Introduza qualquer caracter para voltar ao menu principal:

```

Figura 18 - Output do Somatório de números primos

Menu 1, opção 4: Mostrar o primeiro número primo que seja maior ou igual a um valor inserido

```
Opcao Escolhida:      4. Mostrar o primeiro numero primo que seja maior ou igual a um valor inserido;
|-----|
| Introduza o numero inicial de busca: |
|-----|
```

Figura 19 - Output do menu 1, opção 4

Este programa tem como objetivo mostrar o primeiro número primo superior do número introduzido.

Caso o valor introduzido seja um número primo, o programa mostrará o mesmo número.

É pedido ao utilizador dois números, um para o valor mínimo e outro para o valor máximo do intervalo que deseja introduzir.

```
305 void MostrarPrimoMaiorOuIgualAoNumero()
306 {
307     int Contador, NumeroSelecioneado, SerPrimo, TestePrimo;
308
309     puts("\n|-----|");
310     printf("| Introduza o numero inicial de busca:\n");
311     puts("|-----|\n");
312     scanf("%d", &NumeroSelecioneado);
313
314     //Ciclo para incrementar o valor do contador
315     for( Contador=NumeroSelecioneado; Contador <= NumeroSelecioneado + 10; Contador++)
316     {
317
318         SerPrimo = 0;
319
320         //Incrementação da variavel TestePrimo fazendo a operação com o contador
321         for(TestePrimo = 2; TestePrimo < Contador; TestePrimo++)
322         {
323
324             if(Contador % TestePrimo == 0)
325             {
326
327                 SerPrimo = 1;
328                 break;
329
330             }
331
332         }
333
334         //Se o numero calculado for numero primo
335         if(SerPrimo == 0)
336         {
337
338             puts("|-----|");
```

Figura 20 - Código para mostrar o número primo seguinte.

```

339         printf("|O numero primo que seja maior ou igual a %d mais proximo eh: %d |\n", NumeroSelecioneado, Contador);
340         puts("|-----|");
341         break;
342     }
343
344
345     }
346
347     voltar_menu_inicial();
348 }

```

Figura 21 - Código para mostrar o número primo seguinte (continuação)

Exemplo de Aplicação: Supondo que o número introduzido é 50, o Output será:

```

Opcao Escolhida:      4. Mostrar o primeiro numero primo que seja maior ou igual a um valor inserido;
|-----|
| Introduza o numero inicial de busca:|
|-----|
50
|-----|
|O numero primo que seja maior ou igual a 50 mais proximo eh: 53 |
|-----|

Introduza qualquer caracter para voltar ao menu principal:

```

Figura 22 - Output do programa que mostra o número primo seguinte

Menu 1, opção 5: Mostrar o número primo mais próximo de um valor inserido pelo utilizador

```
Opcao Escolhida:      5. Mostrar o numero primo mais prximo de um valor inserido pelo utilizador;
|-----|
| Introduza o numero inicial de busca: |
|-----|
```

Figura 23 – Output do menu 1, opção 5

Este programa tem como objetivo mostrar apenas o número primo mais perto do número inserido.

É pedido ao utilizador um número, para que o programa possa ver qual é o número primo mais próximo do inserido pelo utilizador.

```
351 void NumeroPrimoMaisPertoDoIntroduzido()
352 {
353     int Contador, NumeroSelecioneado, SerPrimo, TestePrimo;
354     puts("");
355     puts("|-----|");
356     printf("| Introduza o numero inicial de busca:\n");
357     puts("|-----|\n");
358     scanf("%d", &NumeroSelecioneado);
359
360     //Ciclo para incrementar o valor do contador
361     for( Contador=NumeroSelecioneado; Contador < NumeroSelecioneado + 10; Contador--)
362     {
363
364         SerPrimo = 0;
365         //Incrementação da variavel TestePrimo fazendo a operação com o contador
366         for(TestePrimo = 2; TestePrimo < Contador; TestePrimo++)
367         {
368
369             if(Contador % TestePrimo == 0)
370             {
371
372                 SerPrimo = 1;
373                 break;
374
375             }
376
377         }
378         //Se o numero calculado for numero primo
379         if(SerPrimo == 0)
380         {
381
382             puts("|-----|");
383             printf("|O numero primo mais proximo de %d eh: %d |\n", NumeroSelecioneado, Contador);
384             puts("|-----|");
385             break;
386
387         }
388     }
389     voltar_menu_inicial();
390 }
```

Figura 24 - Código para mostrar apenas o número primo mais perto do número inserido.

Exemplo de Aplicação: Supondo que o número introduzido é 160, o Output será:

```
Opcao Escolhida:      5. Mostrar o numero primo mais prximo de um valor inserido pelo utilizador;
|-----|
| Introduza o numero inicial de busca: |
|-----|

160
|-----|
| O numero primo mais proximo de 160 eh: 157 |
|-----|

      Introduza qualquer caracter para voltar ao menu principal:
```

Figura 25 - Output do Programa

Menu 1, opção 6: Mostrar o primeiro número primo que seja maior ou igual a um valor inserido

```
Opcao Escolhida:      6. Decompor um numero nos seus fatores primos;
Insira um numero:
```

Figura 26 - Output do menu 1, opção 6

Este programa tem como objetivo decompor qualquer número e mostrar a sua decomposição

É pedido ao utilizador um número, para que o programa possa decompor o mesmo e que diga quais foram os números.

```
392
393 void Decomposicao()
394 {
395     int NumEscolha, contador;
396     printf("\nInsira um numero: \n");
397     scanf("%d", &NumEscolha);
398     printf("\nA decomposicao em fatores primos do numero %d e: \n\n", NumEscolha);
399     printf(" D[%d]: ", NumEscolha);
400     //Enquanto o contador for menor que o numero introduzido
401     for (contador=1; contador<=NumEscolha; contador++)
402     {
403         //faz a divisão do contador pelo numero escolhido
404         if (NumEscolha % contador ==0)
405         {
406
407             printf("%d, ", contador);
408         }
409     }
410     voltar_menu_inicial();
411 }
```

Figura 27 - Código para a Decomposição

```
Opcao Escolhida:      6. Decompor um numero nos seus fatores primos;
Insira um numero:
120

A decomposicao em fatores primos do numero 120 e:

D[120]: 1, 2, 3, 4, 5, 6, 8, 10, 12, 15, 20, 24, 30, 40, 60, 120,

      Introduza qualquer caracter para voltar ao menu principal:
```

Figura 28 - Output da decomposição

Menu 1, opção 7: Calcular o máximo divisor comum de dois números

```
Opcão Escolhida:      7. Calcular o maximo divisor comum de dois numeros;
-----
Introduza o primeiro numero:
-----
```

Figura 29 - Output do Menu 1, opção 7

Este programa tem como objetivo calcular o Máximo Divisor Comum de 2 números, como tal, é necessário que o utilizador introduza dois números inteiros

```
414 void CalcularMDC()
415 {
416     int Numero1, Numero2, Contador, MDC;
417
418     puts("");
419     puts("-----");
420     printf("Introduza o primeiro numero:\n");
421     puts("-----");
422     scanf("%d", &Numero1);
423     puts("-----");
424     printf("\nIntroduza o segundo numero:\n");
425     puts("-----");
426     scanf("%d", &Numero2);
427
428     //A variável contador não pode ser maior do que o numero 1 nem maior do que o numero 2 inserido
429     for(Contador = 1; Contador <= Numero1 && Contador <= Numero2; Contador++)
430     {
431
432         //A divisão do numero 1 e numero 2 der resto 0 simultaneamente, só assim é que se pode
433         //considerar um Máximo Divisor Comum
434         if(Numero1 % Contador == 0 && Numero2 % Contador == 0)
435             MDC = Contador;
436
437     }
438     puts("-----");
439     printf("O Maximo Divisor Comum de %d e %d eh: %d\n", Numero1, Numero2, MDC);
440     puts("-----");
441     voltar_menu_inicial();
442 }
```

Figura 30 - Código do MDC

Exemplo de Aplicação: Supondo que os números introduzidos são 120 e 110, o Output será:

```
Opcao Escolhida:      7. Calcular o maximo divisor comum de dois numeros;
-----
Introduza o primeiro numero:
-----
120
-----

Introduza o segundo numero:
-----
110
-----
O Maximo Divisor Comum de 120 e 110 eh: 10
-----

Introduza qualquer caracter para voltar ao menu principal:
```

Figura 31 - Output do MDC

Menu 1, opção 8: Calcular o menor múltiplo comum de dois números

```
Opcao Escolhida:      8. Calcular o menor multiplo comum de dois numeros;
-----
Introduza o primeiro numero:
-----
```

Figura 32 - Output do menu 1, opção 8

Por fim, á semelhança do anterior, este pede para calcular o menor múltiplo comum de dois números, como tal, necessitamos de inserir 2 números novamente.

```
445 void CalcularMMC()
446 {
447
448     int Numero1, Numero2, Contador, MDC, MMC;
449
450     puts("");
451     puts("-----");
452     printf("Introduza o primeiro numero:\n");
453     puts("-----");
454     scanf("%d", &Numero1);
455     puts("-----");
456     printf("\nIntroduza o segundo numero:\n");
457     puts("-----");
458     scanf("%d", &Numero2);
459
460     for(Contador = 1; Contador <= Numero1 && Contador <= Numero2; Contador++)
461     {
462
463         //Se estas condições forem cumpridas, a variavel ira ganhar o valor que estiver na outra variavel Contador
464         if(Numero1 % Contador == 0 && Numero2 % Contador == 0){
465
466             MDC = Contador;
467
468         }
469     }
470
471 }
472
473 MMC = (Numero1 * Numero2) / MDC; //Calcular o Menor Multiplo Comum usando o Maximo Divisor Comum
474
475 puts("-----");
476 printf("Tendo em conta que o Maximo Divisor Comum de %d e %d eh: %d...\n", Numero1, Numero2, MDC);
477 printf("O Menor Multiplo Comum de %d e %d eh: %d\n", Numero1, Numero2, MMC);
478 puts("-----");
479
480 voltar_menu_inicial();
481 }
```

Figura 33 - Código para calcular Menor Múltiplo comum de dois números

Exemplo de Aplicação: Supondo que os números introduzidos são 5 e 1, o Output será:

```
Opcao Escolhida:      8. Calcular o menor multiplo comum de dois numeros;
-----
Introduza o primeiro numero:
-----
5
-----

Introduza o segundo numero:
-----
1
-----
Tendo em conta que o Maximo Divisor Comum de 5 e 1 eh: 1...
O Menor Multiplo Comum de 5 e 1 eh: 5
-----

                Introduza qualquer caracter para voltar ao menu principal:
```

Figura 34 - Output do menor multiplo

Vetores

Escolhendo a opção 2 do menu principal, a imagem seguinte ilustra o subprograma que poderá escolher relacionados com Vetores/Arrays

```
Operacoes com numeros primos, multiplos e divisores
2. Vetores,
    -1. Inserir e mostrar os respectivos valores num vetor de inteiros;
    -2. Mostrar o vetor de inteiros;
    -3. Preencher um vetor de inteiros com numeros aleatorios;
    -4. Verificar se um valor existe no vetor;
    -5. Mostrar o maior elemento do vetor e a sua posicao;
    -0. Voltar ao menu anterior.
Escolha outra vez outro numero:
```

Figura 35 - Menu 2

```
485 //SEPARADOR DE MENU -----
486 //Menu 2
487 void menu2 ()
488 {
489     printf("Operacoes com numeros primos, multiplos e divisores");
490     printf("\n2. Vetores,");
491     printf("\n\t-1. Inserir e mostrar os respectivos valores num vetor de inteiros;");
492     printf("\n\t-2. Mostrar o vetor de inteiros;");
493     printf("\n\t-3. Preencher um vetor de inteiros com numeros aleatorios;");
494     printf("\n\t-4. Verificar se um valor existe no vetor;");
495     printf("\n\t-5. Mostrar o maior elemento do vetor e a sua posicao;");
496     printf("\n\t-0. Voltar ao menu anterior.");
497     printf("\nEscolha outra vez outro numero:");
498
499     char escolha;
500     scanf(" %c", &escolha);
501     switch (escolha)
502     {
503     case '1':
504     {
505         printf("\n\nOpcao Escolhida:\t1. Inserir valores num vetor de inteiros;");
506         ColocarNumVetor();
507         break;
508     }
509     case '2':
510     {
511         printf("\n\nOpcao Escolhida:\t2. Mostrar o vetor de inteiros;");
512         break;
513     }
514     case '3':
515     {
516         printf("\n\nOpcao Escolhida:\t3. Preencher um vetor de inteiros com numeros aleatórios;");
517         PreencherVetorRandom();
518         break;
519     }
520     case '4':
```

Figura 36 - Código do Menu 2


```

520 case '4':
521 {
522     printf("\n\nOpcao Escolhida:\t4. Verificar se um valor existe no vetor;");
523     EstaAiAlgunNumero();
524     break;
525 }
526 case '5':
527 {
528     printf("\n\nOpcao Escolhida:\t5. Mostrar o maior elemento do vetor e a sua posicao;");
529     MaiorNumeroVetor();
530     break;
531 }
532 case '0':
533 {
534     printf("\n\nOpcao Escolhida:\t0. Voltar ao menu anterior.\n");
535     break;
536 }
537 }
538 }
539

```

Figura 37 - Continuação

Exemplo de aplicação: Opção 4 - Verificar se um valor existe no vetor.

```

Operacoes com numeros primos, multiplos e divisores
2. Vetores,
    -1. Inserir e mostrar os respectivos valores num vetor de inteiros;
    -2. Mostrar o vetor de inteiros;
    -3. Preencher um vetor de inteiros com numeros aleatorios;
    -4. Verificar se um valor existe no vetor;
    -5. Mostrar o maior elemento do vetor e a sua posicao;
    -0. Voltar ao menu anterior.
Escolha outra vez outro numero:4

Opcao Escolhida:      4. Verificar se um valor existe no vetor;
Introduza quantos numeros quer colocar:

```

Figura 38 - Exemplo de escolha do menu 2

Menu 2, opção 1: Inserir e mostrar os respetivos valores num vetor de inteiros

```
Opcao Escolhida:      1. Inserir valores num vetor de inteiros;
Introduza o tamanho do Array:
```

Figura 39 - Output do Menu 2, opção 1

Este programa tem como objetivo perguntar o tamanho do Array, ou seja, perguntar quantos números o utilizador quer escrever, depois o utilizador tem de escrever todos os números e só depois disso é que o programa mostra os números inseridos.

```
535 void ColocarNumVetor()
536 {
537
538     int i;
539     char valueArray;
540     printf("\nIntroduza o tamanho do Array: ");
541
542     //ler quantos numeros o utilizador quer colocar
543     scanf("%d", &valueArray);
544     printf("\n\nIntroduza %d numeros:\n", valueArray);
545
546     //inicialização do array com o valor inserido pelo utilizador
547     int arrayDeNumeros[valueArray] [256];
548
549     //enquanto o contador i não chegar ao valor introduzido pelo utilizador
550     //o programa está sempre a pedir numeros
551     for (i = 0; i < valueArray; i++)
552         scanf("%s", arrayDeNumeros[i]);
553
554     //Quando contador for igual ao valor inserido, o programa já tem tudo o que precisava
555     //agora mostra o que tem dentro do Array, nesta caso, de nome ValueArray
556     for (int c = 0, i=0; c < valueArray, i<valueArray; c++, i++)
557     {
558         printf("Na posicao %d do Array esta: %s\n", c, arrayDeNumeros[i] );
559     }
560     voltar_menu_inicial();
561 }
```

Figura 40 - Pedir e Mostar o Array

Exemplo de Aplicação: Escrever 6 números seguidos de 10 a 16

```
Introduza 6 numeros:
11
12
13
14
15
16
Na posicao 0 do Array esta: 11
Na posicao 1 do Array esta: 12
Na posicao 2 do Array esta: 13
Na posicao 3 do Array esta: 14
Na posicao 4 do Array esta: 15
Na posicao 5 do Array esta: 16

Introduza qualquer caracter para voltar ao menu principal:
```

Figura 41 - Output do Programa

Menu 2, opção 2: Mostrar valores Aleatórios

```
Opcao Escolhida:      2. Preencher um vetor de inteiros com numeros aleat%rios;
Introduza o tamanho do array
```

Este programa tem como objetivo perguntar o tamanho do Array, ou seja, perguntar quantos números o utilizador quer escrever, mostrar números aleatórios até ao valor que o utilizador escolheu.

```
563 void PreencherVetorRandom()
564 {
565     //funcao rand() = 1 to 32 000
566     int randomAte, i;
567     srand(0);
568     printf("\nIntroduza o tamanho do array\n");
569     scanf("%d", &randomAte);
570     int random[randomAte];
571
572     //Encher o Array até à posição que o utilizador quer
573     for(i=1; i<=randomAte; i++)
574     {
575         random[randomAte]=rand(); //função para gerar numeros aleatorios
576         printf("\nO que esta na posicao [%d] do array e: %d", i, random[randomAte]);
577     }
578     voltar_menu_inicial();
579 }
```

Exemplo de Aplicação

```
Opcao Escolhida:      2. Preencher um vetor de inteiros com numeros aleat%rios;
Introduza o tamanho do array
4

O que esta na posicao [1] do array e: 38
O que esta na posicao [2] do array e: 7719
O que esta na posicao [3] do array e: 21238
O que esta na posicao [4] do array e: 2437

Introduza qualquer caracter para voltar ao menu principal:
```

Menu 2, opção 2: Mostrar valores Aleatórios

```
Opcao Escolhida:      4. Mostrar o maior elemento do vetor e a sua posicao;  
Introduza quantos numeros quer colocar:
```

```
581 void EstaAiAlgunNumero()  
582 {  
583     int array[30];  
584     int numeroEscolhido, i, dimensaoArray;  
585     int existe=0;  
586     printf("\nIntroduza quantos numeros quer colocar: ");  
587     scanf("%d", &dimensaoArray);  
588     printf("Introduza os %d numeros", dimensaoArray);  
589  
590     //Preenche o array com os valores do Utilizador  
591     for(i=0; i<dimensaoArray; i++)  
592     {  
593         scanf("%d", &array[i]);  
594     }  
595  
596     //O programa pede para introduzir um numero para verificar se ele existe  
597     printf("\nIntroduza um numero para fazer a verificacao: ");  
598     scanf("%d", &numeroEscolhido);  
599  
600     //Percorrer o array á procura do valor  
601     for(i=0; i<dimensaoArray; i++)  
602     {  
603         //Se existe, incrementa um á variavel existe  
604         if(numeroEscolhido==array[i])  
605             existe++;  
606     }  
607  
608     if(existe>0) //Se a variavel existe continuar a 0, é porque o numero não existe no Array  
609     {  
610         printf("Esse numero %d existe", numeroEscolhido);  
611     }  
612     else printf("Nao %d existe", numeroEscolhido);  
613  
614     voltar_menu_inicial();  
615 }
```

Exemplo de Aplicação

```
Opcao Escolhida:      4. Mostrar o maior elemento do vetor e a sua posicao;  
Introduza quantos numeros quer colocar: 4
```

```
Introduza os 4 numeros
```

```
1  
5  
77  
2
```

```
O valor mais alto e 77
```

```
Posicao do array 2
```

```
Introduza qualquer caracter para voltar ao menu principal:
```

Menu 2, opção 3: Mostrar valores Aleatórios

```
Opcao Escolhida:      3. Verificar se um valor existe no vetor;  
Introduza quantos numeros quer colocar: 4
```

```
581 void EstaAiAlgunNumero()  
582 {  
583     int array[30];  
584     int numeroEscolhido, i, dimensaoArray;  
585     int existe=0;  
586     printf("\nIntroduza quantos numeros quer colocar: ");  
587     scanf("%d", &dimensaoArray);  
588     printf("Introduza os %d numeros", dimensaoArray);  
589  
590     //Preenche o array com os valores do Utilizador  
591     for(i=0; i<dimensaoArray; i++)  
592     {  
593         scanf("%d", &array[i]);  
594     }  
595  
596     //O programa pede para introduzir um numero para verificar se ele existe  
597     printf("\nIntroduza um numero para fazer a verificacao: ");  
598     scanf("%d", &numeroEscolhido);  
599  
600     //Percorrer o array á procura do valor  
601     for(i=0; i<dimensaoArray; i++)  
602     {  
603         //Se existe, incrementa um á variavel existe  
604         if(numeroEscolhido==array[i])  
605             existe++;  
606     }  
607  
608     if(existe>0) //Se a variavel existe continuar a 0, é porque o numero não existe no Array  
609     {  
610         printf("Esse numero %d existe", numeroEscolhido);  
611     }  
612     else printf(" %d não existe", numeroEscolhido);  
613  
614     voltar_menu_inicial();  
615 }
```

```
Opcao Escolhida:      3. Verificar se um valor existe no vetor;  
Introduza quantos numeros quer colocar: 4  
Introduza os 4 numeros2  
55  
77  
39  
  
Introduza um numero para fazer a verificacao: 100  
100 não existe  
  
Introduza qualquer caracter para voltar ao menu principal:
```

Menu 2, opção 4: Mostrar o maior elemento do vetor e a sua posição

```
Opcao Escolhida:      4. Mostrar o maior elemento do vetor e a sua posicao;  
Introduza quantos numeros quer colocar: 4
```

```
617 void MaiorNumeroVetor()  
618 {  
619     //declaração de variaveis  
620     int array[30];  
621     int i, dimensaoArray;  
622     int ValorMaisAlto=0;  
623     ValorMaisAlto;  
624  
625     //ler quantos numeros o utilizador quer colocar  
626     printf("\nIntroduza quantos numeros quer colocar: ");  
627     scanf("%d", &dimensaoArray);  
628  
629     //enquanto o contador i não chegar ao valor introduzido pelo utilizador  
630     //o programa está sempre a pedir numeros  
631     printf("\nIntroduza os %d numeros\n", dimensaoArray);  
632     for(i=0; i<dimensaoArray; i++)  
633     {  
634         scanf("%d", &array[i]);  
635     }  
636  
637     //Quando contador for igual ao valor inserido, o programa já tem tudo o que precisava  
638     //agora mostra o que tem dentro do Array, nesta caso, de nome ValueArray  
639     for(i=0; i<dimensaoArray; i++)  
640     {  
641         if(array[i] > array[ValorMaisAlto])  
642             ValorMaisAlto=i;  
643     }  
644     printf("\nO valor mais alto e %d\nPosicao do array %d",array[ValorMaisAlto], ValorMaisAlto);  
645  
646     voltar_menu_inicial();  
647 }
```

```
Opcao Escolhida:      4. Mostrar o maior elemento do vetor e a sua posicao;  
Introduza quantos numeros quer colocar: 4
```

```
Introduza os 4 numeros
```

```
21
```

```
635
```

```
210
```

```
6987
```

```
O valor mais alto e 6987
```

```
Posicao do array 3
```

```
Introduza qualquer caracter para voltar ao menu principal: <
```

Estruturas

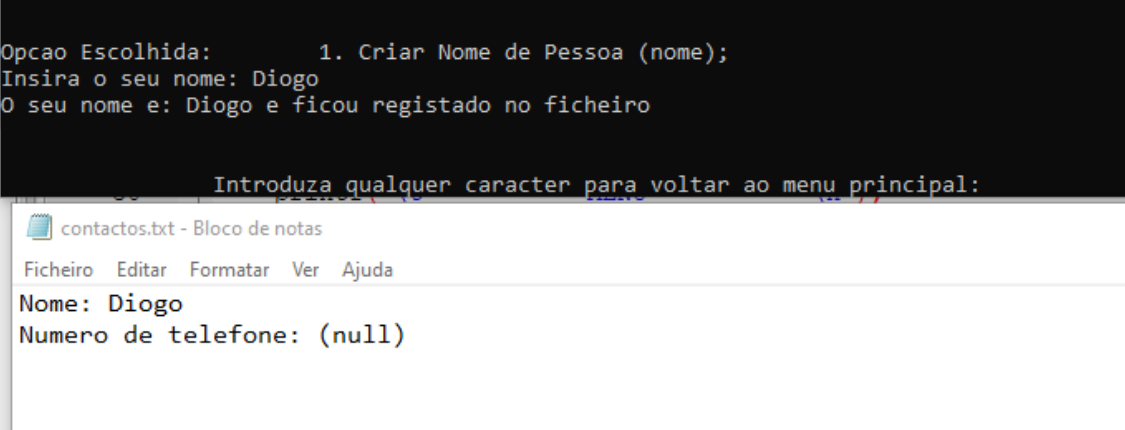
```
694 typedef struct ListaDeContactos
695 {
696     char nome[50+1];
697     char mail[20+1];
698     char numeroDeTelefone[9+1];
699     char morada[100+1];
700     char localidade[15];
701 } LISTAGEM;
```

```
703 void NomeDePessoa()
704 {
705     LISTAGEM L;
706     printf("\nInsira o seu nome: ");
707     scanf("%s", L.nome);
708     printf("O seu nome e: %s e ficou registado no ficheiro\n", L.nome);
709     AbrirEFecharFicheiro(L.nome, 0);
710     voltar_menu_inicial();
711 }
712
```

Exemplo de Aplicação

```
Opcao Escolhida:      1. Criar Nome de Pessoa (nome);
Insira o seu nome: Diogo
O seu nome e: Diogo e ficou registado no ficheiro

Introduza qualquer caracter para voltar ao menu principal:
```



contactos.txt - Bloco de notas

Ficheiro Editar Formatar Ver Ajuda

Nome: Diogo

Numero de telefone: (null)


```

715 void NomeAContacto()
716 {
717
718     LISTAGEM L;
719     printf("\nInsira o seu nome: ");
720     scanf("%s", L.nome);
721     printf("\nInsira o seu contacto: ");
722     scanf("%s", L.numeroDeTelefone);
723     printf("O seu nome e: %s e ficou registado no ficheiro\n", L.nome);
724     printf("O seu numero e: %s e ficou registado no ficheiro\n", L.numeroDeTelefone);
725     AbrirEFecharFicheiro(L.nome, L.numeroDeTelefone);
726     voltar_menu_inicial();
727 }
728

```

Exemplo de Aplicação

```

Opcao Escolhida:      2. Inserir um novo contacto, sempre associado a pessoa;
Insira o seu nome: Diogo

Insira o seu contacto: 123456789
O seu nome e: Diogo e ficou registado no ficheiro
O seu numero e: 123456789 e ficou registado no ficheiro

Introduza qualquer caracter para voltar ao menu principal:

```



contactos.txt - Bloco de notas

Ficheiro Editar Formatar Ver Ajuda

```

Nome: Diogo
Numero de telefone: (null)Nome: 3
Numero de telefone: adNome: Diogo
Numero de telefone: 123456789

```

Função main

```

747 int main()
748 {
749     //CARREGAR FICHEIROS
750     menu();
751     //voltar_menu_inicial();
752     printf("\nFim do Programa");
753     return 0;
754 }

```

Função para retornar para o menu principal

```
733 void voltar_menu_inicial()  
734 {  
735     printf("\n\n\t\tIntroduza qualquer caracter para voltar ao menu principal: ");  
736     char back=0;  
737     scanf(" %c", &back);  
738     if(back!=0)  
739     {  
740         system("cls"); //limpar ecran  
741         printf("Ecran limpo\n\n");  
742         menu();  
743     }  
744 }  
745
```