

Curso de programação em C - FK

- 1 – Estrutura, variáveis e operadores.
- 2 – Estrutura condicional (If, If-Else e Switch-Case).
- 3 – Estrutura de repetição (For, While e Do-While).
- 4 – Vetores, Strings, Matrizes e Structs.
- 5 – Funções e Arquivos.

Lista 2

10 exercícios

Utilizando operações matemáticas, série Fibonacci, MDC, MMC, número primo, fatorial, número por extenso, dígito verificador.

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
/* 1- Fazer um programa para receber do usuário um tempo em segundos, correspondente à duração de um evento qualquer (por ex. jogo de futebol).
```

```
Calcular e mostrar ao usuário o tempo equivalente em horas, minutos e segundos*/
```

```
int main()
```

```
{
```

```
    int segundo, hora=0, minuto=0;
```

```
    printf("Digite a duracao do evento em segundos: ");
```

```
    scanf("%d",&segundo);
```

```
    if (segundo >= 60)
```

```
    {
        minuto = segundo / 60; //1min = 60 segundos
```

```
    if (segundo >= 3600)
```

```
    {
        hora = segundo / 3600; //1 hora = 3600 segundos
```

```
    if (minuto >= 1)
```

```
    {
        segundo = segundo % 60;
```

```
    if (minuto % 60 == 0)
```

```
    {
        minuto = 0;
```

```
    else if (hora >= 1)
```

```
    {
        minuto = minuto % 60;
```

```
    printf("O tempo do evento e de %dh%dm%ds",hora, minuto, segundo);
```

```
    return 0;
```

```
}
```

Digite a duracao do evento em segundos: 3662

O tempo do evento e de 1h1m2s

Process exited after 13.73 seconds with return value 0

Pressione qualquer tecla para continuar. . . |

Digite a duracao do evento em segundos: 1052

O tempo do evento e de 0h17m32s

Process exited after 8.292 seconds with return value 0

Pressione qualquer tecla para continuar. . . |

```
#include <stdio.h>
#include <stdlib.h>
```

/ 2 - Fazer um programa para mostrar os 15 primeiros termos da série de Fibonacci.
Os primeiros números de Fibonacci são: 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89,
144, 233, 377, 610, 987, 1597, 2584, 4181, 6765,
A regra básica da sequência é: do terceiro termo em diante, cada novo é a
soma dos dois anteriores.
Ela Inicia com o número 1, sendo o segundo termo também o número 1, o terceiro
é o número 2, pois $1 + 1 = 2$. */*

```
int main()
{
    int x = 1, y = 1, proximo, i;

    printf("A sequencia de Fibonacci e:\n");

    for (i=3;i<=15;i++)
    {
        proximo = x + y; // 1 1 2 3 5 8
        x = y;
        y = proximo;
        printf(" %d",proximo);
    }

    return 0;
}
```

A sequencia de Fibonacci e:

1 1 2 3 5 8 13 21 34 55 89 144 233 377 610

Process exited after 0.01268 seconds with return value 0

Pressione qualquer tecla para continuar. . . |

```
#include <stdio.h>
#include <stdlib.h>
```

/ 3 - O número 3025 possui a seguinte característica: $30 + 25 = 55 \rightarrow 55 * 55 = 3025$.
Fazer um programa para obter todos os números de 4 algarismos com a mesma
característica do número 3025.*/*

```
int main()
{
    int i, bloco1, bloco2, soma, multi;

    for(i=1000; i<=9999; i++)
    {
        bloco1 = i/100;
        bloco2 = i%100;
        soma = bloco1 + bloco2;
        multi = soma * soma;

        if (multi == i)
        {
            printf("%d + %d = %d -> %d x %d = %d\n", bloco1, bloco2, soma, soma, soma, i);
        }
    }

    return 0;
}
```

```
20 + 25 = 45 -> 45 x 45 = 2025  
30 + 25 = 55 -> 55 x 55 = 3025  
98 + 1 = 99 -> 99 x 99 = 9801
```

```
-----  
Process exited after 0.01098 seconds with return value 0  
Pressione qualquer tecla para continuar. . . |
```

```
#include <stdio.h>
#include <stdlib.h>
```

/ 4 - Seja a seguinte série: 1, 4, 9, 16, 25, 36, ...
Escreva um programa que gere esta série até o N-ésimo termo.
Este N-ésimo termo é digitado pelo usuário.*/*

```
int main()
{
    int num, i;

    printf("Digite um numero para a serie: ");
    scanf("%d",&num);

    for(i = 1; i <= num; i++)
    {
        printf("%d ",i * i);
    }

    return 0;
}
```

Digite um numero para a serie: 10

1 4 9 16 25 36 49 64 81 100

Process exited after 3.095 seconds with return value 0

Pressione qualquer tecla para continuar. . . |

```
#include <stdio.h>
#include <stdlib.h>

/* 5 - Fazer um programa para receber dois números inteiros do usuário
e mostrar o seu MMC (mínimo múltiplo comum) e MDC(Máximo divisor comum).*/
```

```
int main()
{
    int a, b, mmc = 1, i = 2; //Variáveis do MMC
    int c, d, mdc = 1, j = 2; //Variáveis do MDC

    printf("Digite o primeiro numero: ");
    scanf("%d",&a);
    printf("Digite o segundo numero: ");
    scanf("%d",&b);
    c = a; //c e d recebem os valores digitados pelo usuário.
    d = b; //as variáveis a e b.
```

```
while (i <= 1000)
{
    if ((a%i == 0) && (b%i == 0))
    {
        mmc = mmc * i;
        a = a/i;
        b = b/i;
    }
    else if (a%i == 0)
    {
        mmc = mmc*i;
        a = a/i;
    }
    else if (b%i == 0)
    {
        mmc = mmc*i;
        b = b/i;
    }
}
```

```
    }
    else
    {
        i++;
    }
}

printf("O mmc e: %d", mmc);

while ((j <= c) || (j <= d))
{
    if ((c%j == 0) && (d%j == 0))
    {
        mdc = mdc * j;
        c = c/j;
        d = d/j;
    }
    else if (c%j == 0)
    {
        c = c/j;
    }
    else if (d%j == 0)
    {
        d = d/j;
    }
    else
    {
        j++;
    }
}

printf("\nO mdc e: %d", mdc);

return 0;
}
```

Digite o primeiro numero: 12

Digite o segundo numero: 18

0 mmc e: 36

0 mdc e: 6

Process exited after 3.765 seconds with return value 0

Pressione qualquer tecla para continuar. . . |

Digite o primeiro numero: 17

Digite o segundo numero: 31

0 mmc e: 527

0 mdc e: 1

Process exited after 8.266 seconds with return value 0

Pressione qualquer tecla para continuar. . . |

```
#include <stdio.h>
#include <stdlib.h>
```

/ 6 - Fazer um programa para solicitar ao usuário um número entre 0 e 99 e mostrar este número por extenso. Se o usuário introduzir um número que não está neste intervalo, mostre: "número inválido". */*

```
int main()
{
    int num, unidades, dezenas;
    printf("Digite um numero: ");
    scanf("%d", &num);

    if (num == 10)
    {
        printf("dez");
        return 0;
    }
    else if (num == 0)
    {
        printf("zero");
        return 0;
    }
    else if ((num < 0) || (num > 99))
    {
        printf("Numero invalido!");
        return 0;
    }
}
```

```
if ((num > 10) && (num < 20))
{
    switch (num)
    {
        case 11:
            printf("onze");
            break;
        case 12:
            printf("doze");
            break;
        case 13:
            printf("treze");
            break;
        case 14:
            printf("quatorze");
            break;
        case 15:
            printf("quinze");
            break;
        case 16:
            printf("dezesesseis");
            break;
        case 17:
            printf("dezesesete");
            break;
        case 18:
            printf("dezoito");
            break;
        case 19:
            printf("dezenove");
            break;
        default:
            break;
    }
    return 0;
}
```

```
unidades = num % 10;
dezenas = num / 10;

if ((num > 19) && (num < 100))
{
    switch (dezenas)
    {
        case 2:
            printf("vinte ");
            break;
        case 3:
            printf("trinta ");
            break;
        case 4:
            printf("quarenta ");
            break;
        case 5:
            printf("cinquenta ");
            break;
        case 6:
            printf("sessenta ");
            break;
        case 7:
            printf("setenta ");
            break;
        case 8:
            printf("oitenta ");
            break;
        case 9:
            printf("noventa ");
            break;
        default:
            break;
    }
}
```

```
if ((num > 19) && (unidades != 0))
    printf("e ");

switch (unidades)
{
    case 1:
        printf("um");
        break;
    case 2:
        printf("dois");
        break;
    case 3:
        printf("tres");
        break;
    case 4:
        printf("quatro");
        break;
    case 5:
        printf("cinco");
        break;
    case 6:
        printf("seis");
        break;
    case 7:
        printf("sete");
        break;
    case 8:
        printf("oito");
        break;
    case 9:
        printf("nove");
        break;
    default:
        break;
}

return 0;
}
```

```
#include <stdio.h>
#include <stdlib.h>
```

/ 7 - Fazer um programa no qual o usuário vai entrando sucessivamente com valores positivos. Quando o usuário entrar com um valor negativo o programa para de pedir valores e calcula a média dos valores já fornecidos.*/*

```
int main()
{
    float num, cont, soma = 0, media = 0;

    while (1)
    {
        printf("Digite um numero: ");
        scanf("%f",&num);

        if (num < 0)
            break;

        cont++;
        soma = soma + num;
        media = soma/cont;
    }

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

    return 0;
}
```

```
Digite um numero: 10  
Digite um numero: 5  
Digite um numero: 3  
Digite um numero: 4  
Digite um numero: 2  
Digite um numero: -4  
A media e: 4.80
```

```
-----
```

```
Process exited after 14.66 seconds with return value 0  
Pressione qualquer tecla para continuar. . . |
```

```
#include <stdio.h>
#include <stdlib.h>
```

/ 8 - Escreva um programa que lê um inteiro positivo e verifica se ele contém o dígito 3 em qualquer posição. */*

```
int main()
{
    int numero, cont=0;
    scanf("%d",&numero);

    if (numero == 3)
        cont++;

    if ((numero/10 == 3) || (numero%10 == 3))
        cont++;

    if ((numero/100 == 3) || (numero%100 == 3))
        cont++;

    if (cont == 0)
        printf("Nao consta o digito 3 no numero %d", numero);
    else
        printf("O digito 3 aparece no numero %d", numero);

    return 0;
}
```

73

0 digito 3 aparece no numero 73

Process exited after 4.662 seconds with return value 0
Pressione qualquer tecla para continuar. . . |

512

Nao consta o digito 3 no numero 512

Process exited after 3.628 seconds with return value 0
Pressione qualquer tecla para continuar. . . |

```
#include <stdio.h>
#include <stdlib.h>
```

/ 9 - Escreva um programa para calcular o fatorial do número N, cujo valor é obtido através do usuário pelo teclado.*/*

```
int main()
{
    int num, fat;

    scanf("%d",&num);

    for(fat = 1; num > 1; num--)
    {
        fat = fat * num;
    }

    printf("\n0 fatorial e: %d", fat);

    return 0;
}
```

6

0 fatorial e: 720

Process exited after 2.775 seconds with return value 0

Pressione qualquer tecla para continuar. . .

0

0 fatorial e: 1

Process exited after 1.694 seconds with return value 0

Pressione qualquer tecla para continuar. . .

```
#include <stdio.h>
#include <stdlib.h>
```

```
/* 10 - Escrever um programa para ler um número inteiro do usuário e exibir
o maior número primo que seja menor do que o número digitado. */
```

```
int main()
{
    int num, primo, i, j, cont=0;
    scanf("%d", &num);

    primo = num - 1;

    if (num < 2)
    {
        printf("Numero invalido!!!");
        return 0;
    }

    for (i = 2; i < num; i++)
    {
        for (j = 2; j <= i; j++)
        {
            if (i % j == 0)
                cont++;
        }
        if (cont == 1)
            primo = i;

        cont=0;
    }
    printf("O maior numero primo e: %d", primo);

    return 0;
}
```

28

0 maior numero primo e: 23

Process exited after 5.588 seconds with return value 0

Pressione qualquer tecla para continuar. . . |

10

0 maior numero primo e: 7

Process exited after 10.43 seconds with return value 0

Pressione qualquer tecla para continuar. . . |

Fim da lista 2

- Autor: [Fernando Eduardo](#).
- Dúvidas encaminhe e-mail para:
dunano@outlook.com