

Linguagem Técnica de Programação

1. Dados o tipo de combustível e a quantidade de litros, calcule e exiba o valor a ser pago, seguindo:

Combustível	Preço por Litro
A – Alcool	1,7997
D – Diesel	0,9798
G - Gasolina	2,1009

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main()
5  {
6      int tipo;
7      float quant, pagar;
8
9      printf("1-Alcool / 2-Diesel / 3-Gasolina: \n");
10     printf("Qual o tipo de combustivel: \n");
11     scanf("%d", &tipo);
12
13     printf("Digite a quantidade em litros \n");
14     scanf("%f", &quant);
15
16     switch(tipo){
17         case 1:
18             pagar = quant * 1.7997;
19             break;
20         case 2:
21             pagar = quant * 0.9798;
22             break;
23         case 3:
24             pagar = quant * 2.1009;
25             break;
26         default:
27             printf("Entrada incorreta do tipo de combustivel ");
28     }
29
29     printf("O valor a ser pago: %f \n", pagar);
30     return 0;
31 }
```

2. Dada a quantidade de alunos de uma turma, calcular a média semestral de cada aluno da turma e mostrar uma mensagem para os alunos aprovados.

```

1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main()
5  {
6      int qtde_alunos, contador;
7      float notal, nota2, media;
8
9      printf("Digite a quantidade de alunos: ");
10     scanf("%d", &qtde_alunos);
11
12     contador=1;
13
14     while(contador<=qtde_alunos){
15
16         printf("Digita as notas do aluno %d \n", contador);
17
18         printf("Digite a notal:");
19         scanf("%f", &notal);
20
21         printf("Digite a nota2:");
22         scanf("%f", &nota2);
23
24         media = (notal+nota2)/2;
25
26         if(media>=5){
27             printf("Aprovado \n");
28         }else{
29             printf("Reprovado \n");
30         }
31         contador++;
32     }
33     return 0;

```

3. Dado um número inteiro positivo n , imprimir os n primeiros naturais ímpares. Exemplo:
Para $n = 4$, a saída deverá ser 1, 3, 5, 7

Escreva um algoritmo e, em seguida, implemente-o em Linguagem C

```

1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main()
5  {
6      int n, cont, apres;
7
8      apres = 1;
9
10     printf("Digite um numero inteiro: ");
11     scanf("%d", &n);
12
13     cont = 1;
14     while(cont <= n) {
15         printf("%d ", apres);
16         apres = apres + 2;
17         cont++; // cont = cont + 1;
18     }
19
20     // apres - 1 - 3 - 5 - 7
21     // n - 3
22     // cont - 1 - 2 - 3 - 4
23     // Tela: 1 3 5
24
25     return 0;
26 }

```

4. Escrever um programa em linguagem C que resolva o problema a seguir:

Uma pesquisa sobre a população de uma determinada região coletou os seguintes dados, referentes a cada habitante, para serem analisados:

- idade (em anos)
- sexo (m-masculino, f-feminino)
- Para cada habitante são informados os dados acima.
- A fim de indicar o final da entrada, após a sequência de dados dos habitantes, o usuário entrará com o valor -1 para a idade, o que deve ser interpretado pelo algoritmo como fim de entrada. Encontrar a maior idade de um conjunto

de indivíduos e o percentual de indivíduos do sexo feminino com idade entre 18 e 35 anos.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main()
5  {
6      int idade, maior;
7      float contf=0, cont_intervalo=0;
8      float perc_f_intervalo=0;
9      char sexo; //quantidade de caracteres (char)
10
11     printf("Digite a idade: ");
12     scanf("%d", &idade);
13
14     maior = idade;
15
16     while (idade != -1){ //diferent !=
17         setbuf(stdin, NULL); //limpar o buffer para ignorar o enter
18         printf("Digite o sexo: (f-feminino/m-masculino):");
19         scanf("%c", &sexo);
20
21         if (sexo=='f'){
22             contf++; //contf = contf + 1;
23         }
24
25         if(sexo=='f' && (idade >= 18 && idade <= 35)){
26             cont_intervalo++;
27         }
28
29         printf("Digite a idade: ");
30         scanf("%d", &idade);
31
32         if (idade > maior){
33             maior = idade;
34         }
35     }
36     perc_f_intervalo = (cont_intervalo/contf)*100;
37
38     printf("Programa finalizado! \n");
39     printf("A maior idade digitada: %d \n", maior);
40     printf("A quantidade de feminino: %.0f \n", contf);
41     printf("A quantidade de feminino no inter 18-35: %.0f \n", cont_intervalo);
42     printf("O percentual solicitado: %.2f \n", perc_f_intervalo);
43
44     return 0;
45 }
```

5. Leia o ano de nascimento de uma pessoa viva; só aceite valores “razoáveis”. Imprima o valor do ano e se ele é ou não ano bissexto.

- Entende-se por “razoável” por exemplo:
 - número inteiro;

- anos anteriores ou igual ao ano atual;
- é razoável considerar que alguém viva até 130 anos?
- Considere que um ano múltiplo de quatro é bissexto.

```

1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main()
5  {
6      int ano_nasc, valida;
7
8      do{
9          printf("Digite o ano de nascimento:\n");
10         scanf("%d", &ano_nasc);
11     }while(ano_nasc < 1890 || ano_nasc > 2020);
12
13     // nao < 1890.....2020 > nao
14

```

```

15     valida = ano_nasc % 4;
16
17     // 5 / 2 = 2 resto 1
18     // 5 % 2 = 1 (MOD)
19
20     if (valida == 0){
21         printf("Ano Bissexto");
22     }else{
23         printf("Ano Nao Bissexto");
24     }
25
26     return 0;
27 }

```

6. Ler um número inteiro n. Escrever a soma de todos os números de 1 até n.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main()
5  {
6      int num, cont, soma=0;
7
8      printf("Digite um numero positivo: \n");
9      scanf("%d", &num);
10
11     //Exemplo: num = 5 -> 1+2+3+4+5 = 15
12
13     cont = 1;
14     do{
15         soma=soma+cont;
16         cont++;
17     }while(cont <= num);
18
19     printf("A soma de todos os numeros de 1 a %d = %d", num, soma);
20
21     return 0;
22 }
```

7. Ler um número inteiro n. Escrever a soma de todos os números pares de 2 até n.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main()
5  {
6      int num, cont, soma=0;
7
8      printf("Digite um numero positivo: \n");
9      scanf("%d", &num);
10
11     //Exemplo: num = 5 -> 1+2+3+4+5 = 15
12
13     //inicio, final e passo
14     for(cont=1; cont<=num; cont++){
15         soma=soma+cont;
16     }
17
18     printf("A soma de todos os numeros de 1 a %d = %d", num, soma);
19
20     return 0;
21 }
```

8. Construa um programa que receba um número de 1 a 7 e apresente o dia da semana referente.

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  int main()
5  {
6      int num;
7
8      printf("Digite um numero para informar o dia da semana:");
9      scanf("%d", &num);
10
11     switch(num) {
12         case 1:
13             printf("Domingo \n");
14             break;
```

```
15         case 2:
16             printf("Segunda-feira \n");
17             break;
18         case 3:
19             printf("Terca-feira \n");
20             break;
```

```
21
22         case 4:
23             printf("Quarta-feira \n");
24             break;
25         case 5:
26             printf("Quinta-feira \n");
27             break;
28         case 6:
29             printf("Sexta-feira \n");
30             break;
31         case 7:
32             printf("Sabado \n");
33             break;
34         default:
35             printf("Entrada incorreta \n");
```

```
35     }
36     return 0;
37 }
```


FUNÇÕES E PROCEDIMENTOS

- **Modularizar** um programa consiste em dividi-lo em partes que serão desenvolvidas em separado. Essas partes executam tarefas menores que depois serão acopladas para formar o programa. A cada uma dessas partes chamamos de subprograma.
- Os subprogramas permitem uma melhor legibilidade e manutenabilidade do programa.
- A forma de modularizar um programa em C é através de Funções e Procedimentos.

Pode-se então dizer que: Funções e Procedimentos são blocos de código que podem ser nomeados e chamados de dentro de um programa.

Forma Geral

tipo_de_retorno nome_função (parâmetros)

{

declarações

comandos

}

```
1  #include <stdio.h>
2  #include <stdlib.h>
3
4  //Procedimento (sem passagem de parâmetro)
5  void imprimeLinha_3 ()
6  {
7      int i;
8      for(i=0; i < 3; i++) printf("+");
9      printf ("\n");
10 }
11
```

```
12 void imprimeLinha_5()
13 {
14     int i;
15     for(i=0; i < 5; i++) printf("+");
16     printf ("\n");
17 }
18
19 void imprimeLinha_7()
20 {
21     int i;
22     for(i=0; i < 7; i++) printf("=");
23     printf ("\n");
24 }
25
```

```
26 int main()
27 {
28     imprimeLinha_3();
29     imprimeLinha_5();
30     imprimeLinha_7();
31     imprimeLinha_5();
32     imprimeLinha_3();
33     return 0;
34 }
```

```

1  #include <stdio.h>
2  #include <stdlib.h>
3
4  //Procedimento (com passagem de parâmetro)
5  void imprimeCh(char ch, int n)
6  {
7      int i;
8      for(i=0; i < n; i++)
9          printf("%c", ch);
10         printf("\n");
11     }
12     int main()
13     {
14         imprimeCh('+', 3);
15         imprimeCh('+', 5);
16         imprimeCh('=', 7);
17         imprimeCh('+', 5);
18         imprimeCh('+', 3);
19         return 0;
20     }

```

```

1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <locale.h>
4
5  //Função
6  int verifPar(int k)
7  {
8      if (k%2 == 0) return 1;
9      else return 0;
10 }
11
12 int main()
13 {
14     setlocale(LC_ALL, "Portuguese");//habilita a acentuação para o português
15

```

```

16     int num, resultado;
17     printf("Digite um numero ");
18     scanf("%d", &num);
19     resultado = verifPar(num);
20     if(resultado == 1)
21         printf("\nO numero que voce digitou é par\n");
22     else
23         printf("\nO numero que voce digitou não é par\n");
24     return 0;
25 }

```

