



EXERCÍCIOS CONDICIONAIS



primo ou nao.cpp

```
1 //Conferir se um número é ou não primo
2
3 #include <stdio.h>
4 #include <locale.h>
5
6 int main(void) {
7     setlocale(LC_ALL, "portuguese");
8
9     int numero;
10
11     printf("É PRIMO OU NÃO É?");
12     printf("\nDigite um número: ");
13     scanf("%i", &numero);
14
15     //[Anotações no caderno de regras de divisibilidade]
16
17     if (numero <= 1) {
18         printf("O número digitado: (%d) Não é primo!\n", numero);
19     } else if (numero == 2 || numero == 3) {
20         printf("O número digitado: (%d) É primo!\n", numero);
21     } else if (numero % 2 == 0 || numero % 3 == 0) {
22         printf("O número digitado: (%d) Não é primo!\n", numero);
23     } else if (numero % 5 == 0 && numero != 5) {
24         printf("O número digitado: (%d) Não é primo!\n", numero);
25     } else if (numero % 7 == 0 && numero != 7) {
26         printf("O número digitado: (%d) Não é primo!\n", numero);
27     } else if (numero % 11 == 0 && numero != 11) {
28         printf("O número digitado: (%d) Não é primo!\n", numero);
29     } else {
30         printf("O número digitado: (%d) É primo!\n", numero);
31     }
32
33     return 0;
34 }
```

Pra ser primo existem algumas regrinhas que podemos passar para o código

- Se o número for menor ou igual a 1, ele não é primo
- Se o número for 2 ou 3, ele é primo

setlocale();

Regras de DIVISIBILIDADE

Defined in header <locale.h>	
setlocale	gets and sets the current C locale (function)
localeconv	queries numeric and monetary formatting details of the current locale (function)
lconv	formatting details, returned by localeconv ¹² (struct)
Locale categories	
LC_ALL	locale categories for setlocale ¹² (macro constant)
LC_COLLATE	
LC_CTYPE	
LC_MONETARY	
LC_NUMERIC	
LC_TIME	

• Documentação

Regras de Paridade

```
1  /* Faça um algoritmo que leia um número
2  e verifique se ele é par ou ímpar, imprima a frase "é par" ou "é ímpar"*/
3
4  #include<stdio.h>
5  #include<locale.h>
6
7  int main(void){
8
9      int inumero;
10
11      setlocale(LC_ALL,"portuguese");
12
13      printf("PROGRAMA DE PAR OU ÍMPAR");
14
15      printf("\nDigite um número:");
16      scanf("%i",&inumero);
17
18      if(inumero % 2 == 0){
19
20          printf("O número é par!");
21
22      }else{
23          printf("O número é ímpar!");
24      }
25
26 }
```

paridade


Qualidade do que é par: características do que é igual ou semelhante.
Comparação efetuada para provar que uma coisa pode ser igual ou semelhante a outra.

0, 2, 4, 6, 8



NÚMEROS PARES

1, 3, 5, 7, 9



NÚMEROS ÍMPARES

0	2	4	6	8
10	12	14	16	18
20	22	24	26	28
30	32	34	36	38
40	42	44	46	48
50	52	54	56	58
60	62	64	66	68
70	72	74	76	78
80	82	84	86	88
90	92	94	96	98

Par: #221
0, 2, 4, 6 e 8

Ímpar:
1, 3, 5, 7 e 9

 Mais imagens

Paridade :

Um número inteiro qualquer é dito par se, ao ser dividido pelo número dois, resulta em um número inteiro, ou seja, seu resultado é um número sem casas decimais, caso contrário esse número é dito ímpar. Alguns números pares são 2, 4, 6, 8, 10 e assim por diante. [Wikipédia](#)

Feedback

```
1  /*3- Ler um número inteiro e determinar se é divisível por 5. Se for divisível,
2  escrever uma mensagem na tela "o número xx é divisível por 5". Se não for divisível
3  imprima "o número xx não é divisível por 5". No lugar de xx deve aparecer o número que
4  foi lido pelo usuário.*/
5
6  #include<stdio.h>
7  #include<locale.h>
8
9
10 int main(void){
11
12     int inumero;
13
14     setlocale(LC_ALL,"portuguese");
15
16     printf("Digite um número:");
17     scanf("%i",&inumero);
18
19     //[Condição que vai determinar se é ou não divisível por 5
20     if(inumero % 5 == 0){
21
22         printf("O número %i é divisível por 5",inumero);
23
24     }else{
25
26         printf("O número %i não é divisível por 5",inumero);
27
28     }
29
30
31
32
33
34
35
36 }
```



```

1  /*4- Dado um ano D.C (Depois de Cristo), identifique se este é um ano bissexto ou não.
2  Considere que para o ano ser bissexto basta que seja divisível por 400. Caso contrário,
3  este precisará ser divisível por 4 e não divisível por 100.*/
4
5  #include<stdio.h>
6  #include<locale.h>
7
8  int main(void){
9
10     int ano;
11
12     printf("CÁLCULO DE ANO BISSEXTO");
13     printf("\nDigite um ano:");
14     scanf("%i",&ano);
15
16     //Construção da condição
17
18     if(ano % 400 == 0 && ano % 4 == 0){
19
20         printf("Ano bissexto!");
21
22     }else if(ano % 100 != 0){
23
24         printf("Ano bissexto!");
25
26     }else{
27
28         printf("Ano não bissexto!");
29
30     }
31
32
33
34     return 0;
35 }

```

CÁLCULO DO ANO BISSEXTO



Ano bissexto: origem e como calcular

Acesse e descubra como é feito o cálculo do ano bissexto, isto é, o ano em que fevereiro...

K Escola Kids /

```

1  /*5-Depois da liberação do governo para as mensalidades dos planos de saúde,
2  as pessoas começam a fazer pesquisas para descobrir um bom plano, não muito caro.
3  Um vendedor de um plano de saúde apresentou a Tabela 2. Criar um programa que receba a
4  idade de uma pessoa e imprima o valor que ela deverá pagar, segundo a Tabela 2*/
5
6  #include<stdio.h>
7  #include<locale.h>
8
9  int main(void){
10
11     int idade;
12     float valor_plano;
13
14     setlocale(LC_ALL,"portuguese");
15
16     // [Formatação monetária]
17     setlocale(LC_MONETARY,"pt_BR.utf8");
18     struct lconv *lc = localeconv();
19
20
21     printf("SISTEMA DE PLANO DE SAÚDE");
22     printf("\nDigite sua idade para melhor valor a ser pago no plano de saúde:");
23     scanf("%i",&idade);
24
25     // [Condições baseadas nas tabelas]
26     if(idade <= 0){
27         printf("Idade inválida! [Não consta na tabela do plano de saúde]");
28     }else if(idade <= 10){
29         valor_plano = 30;
30         printf("Valor a ser pago no plano de saúde: %s %.2f",lc->currency_symbol,valor_plano);
31
32     }else if(idade > 10 && idade <= 29){
33         valor_plano = 60;
34         printf("Valor a ser pago no plano de saúde: %s %.2f",lc->currency_symbol,valor_plano);
35
36     }else if(idade > 29 && idade <= 45){
37         valor_plano = 120;
38         printf("Valor a ser pago no plano de saúde: %s %.2f",lc->currency_symbol,valor_plano);
39
40     }else if(idade > 45 && idade <= 59){
41         valor_plano = 150;
42         printf("Valor a ser pago no plano de saúde: %s %.2f",lc->currency_symbol,valor_plano);
43
44     }else if(idade > 59 && idade <= 65){
45         valor_plano = 250;
46         printf("Valor a ser pago no plano de saúde: %s %.2f",lc->currency_symbol,valor_plano);
47
48     }else if(idade > 65){
49         valor_plano = 400;
50         printf("Valor a ser pago no plano de saúde: %s %.2f",lc->currency_symbol,valor_plano);
51     }
52
53
54     return 0;
55 }

```

Importante!

Utilização da biblioteca <locale.h>

setlocale(LC_MONETARY,"pt_BR.utf8");

struct lconv *lc = localeconv();

Pra utilizar o caracter (R\$)

lc -> currency_symbol

```

1  /*6-Um posto de combustível vende três tipos de combustível : álcool, diesel e gasolina.
2  O preço de cada litro dos combustíveis deve ser lido(informado) pelo usuário. Faça um algoritmo que
3  leia um caractere que representa o tipo de combustível comprado (a, d ou g) , o preço do combustível e
4  a quantidade em litros. Utilizar o comando Switch / escolha.
5
6  O programa deve imprimir o valor em reais a ser pago pelo combustível.*/
7
8  #include<stdio.h>
9  #include<locale.h>
10 #include<ctype.h>
11
12 int main(void){
13
14     setlocale(LC_ALL,"portuguese");
15     setlocale(LC_MONETARY,"pt-BR.utf8");
16     struct lconv *lc = localeconv();
17
18     char tipo_combustivel;
19     float valor_combustivel, quantidade;
20
21     printf("BEM-VINDO AO SISTEMA DO POSTO BITU!");
22     printf("\nOpções de combustível: [Álcool,Diesel e Gasolina]");
23     printf("\nDigite a primeira letra do tipo de combustível que deseja:");
24     scanf("%c",&tipo_combustivel);
25
26     printf("Digite o valor do combustível:");
27     fflush(stdin);
28     scanf("%f",&valor_combustivel);
29
30     printf("Digite a quantidade de combustível:");
31     fflush(stdin);
32     scanf("%f",&quantidade);
33
34     tipo_combustivel = toupper(tipo_combustivel);
35     float valor_total;
36     switch(tipo_combustivel){
37
38     case 'A':
39         printf("Combustível escolhido: [Álcool]");
40         valor_total = quantidade * valor_combustivel;
41         printf("\nValor a ser pago: %s %.2f",lc->currency_symbol,valor_total);
42         break;
43
44     case 'D':
45         printf("Combustível escolhido: [Diesel]");
46         valor_total = quantidade * valor_combustivel;
47         printf("\nValor a ser pago: %s %.2f",lc->currency_symbol,valor_total);
48         break;
49
50     case 'G':
51         printf("Combustível escolhido: [Gasolina]");
52         valor_total = quantidade * valor_combustivel;
53         printf("\nValor a ser pago: %s %.2f",lc->currency_symbol,valor_total);
54         break;
55     default:
56
57         printf("Escolha de combustível inválida!");
58
59     }
60
61     return 0;
62 }
63

```

• Utilização da estrutura condicional 'Switch'