

ATIVIDADE 1

Enos Andrade Diniz Sousa

- 1** Escreva um programa que verifique se um ano é bissexto ou não. Um ano é bissexto se (i) é múltiplo de 4 e não termina em 00 ou (ii) se termina em 00 e é múltiplo de 400.

Solução

```
1 #include <stdio.h>
2
3 int main(int argc, char const *argv[])
4 {
5     //variavel ano e continuar para controlar o loop
6     int ano, continuar = 1;
7
8     // enquanto a resposta do usuario ao final do calculo for igual a 1, o algoritmo
9     // sera repetido
10    while(continuar == 1){
11
12        //Pede um ano pelo teclado
13        printf("\nDigite um ano: ");
14
15        // armazena o ano digitado na variavel ano
16        scanf("%i", &ano);
17
18        // faz o teste para verificar se um ano e bissexto e imprime a resposta
19        if (ano % 4 == 0 && ano % 100 != 0 || ano % 400 == 0)
20            printf("\n%i e bissexto!\n", ano);
21        else
22            printf("\n%i nao e bissexto!\n", ano);
23
24        //pergunta se o usuario quer repetir o teste com outro ano
25        printf("\nTestar outro ano?\n1 para sim\n0 para nao\n:");
26
27        //armazenar a resposta na variavel de controle continuar
28        scanf("%i", &continuar);
29    }
30    return 0;
31 }
```

- 2** Existe um algoritmo interessante para obtermos a raiz quadrada de um número quando ela é exata. Para isso, basta subtrairmos números ímpares consecutivos do número do qual desejamos obter a raiz quadrada, até que o resultado seja menor ou igual a 0. O número de vezes será a raiz do número. Por exemplo, a raiz de 25 é 5, pois ao diminuirmos de 25 5 números ímpares, obtemos o valor 0, ou seja, $25 - 1 - 3 - 5 - 7 - 9 = 0$. Elabore esse algoritmo que leia um inteiro e escreva sua raiz quadrada. Por exemplo, se o valor lido for 49, ele escreve 7.

Solução

```
1 #include <stdio.h>
2
3 int main(int argc, char const *argv[])
4 {
```

```
5  int n, raiz, repetir = 1;
6  int raiz_quadrada(int n);
7
8  while(repetir == 1){
9      printf("Deseja calcular a raiz de qual numero: ");
10     scanf("%d", &n);
11     raiz = raiz_quadrada(n);
12     printf("\nA raiz de %i e %i\n", n, raiz);
13     printf("\nCalcular outra raiz?\n1 para sim\n0 para nao\n:");
14
15     scanf("%i", &repetir);
16 }
17 return 0;
18 }
19 int raiz_quadrada(int n){
20     int aux = 1, raiz = 0;
21     while(n > 0){
22         raiz++;
23         n -= aux;
24         aux+=2;
25     }
26     if (n < 0) {
27         printf("\nA raiz nao e exata! O resultado do calulo nao esta correto!");
28     }
29     return raiz;
30 }
```

3 Deseja-se fazer uma pesquisa a respeito do consumo mensal de energia elétrica em uma determinada cidade. Para isso, são fornecidos os seguintes dados de vários consumidores: código do consumidor, quantidade de kWh consumidos durante o mês e tipo de consumidor (1 – residencial; 2 – comercial; 3 - industrial). Escreva um algoritmo que calcule e escreva os seguintes resultados:

- (a) O menor consumo dentre os consumidores residenciais e seu código;
- (b) O maior consumo dentre os consumidores comerciais e seu código;
- (c) A média geral de consumo industrial.

O algoritmo para de ler dados quando o tipo do consumidor fornecido for 0.

Solução

```
1  #include <stdio.h>
2
3  typedef struct consumidor
4  {
5      int tipo;
6      int codigo;
7      float qtdkwh;
8  }Consumidor;
9
10 int main(int argc, char const *argv[])
11 {
12     int tipo = -1, aux=0, qtdConsumidores = 0;
13     Consumidor consumidores[30];
```

```
14
15 //Adicionar consumidores
16 while(tipo!=0 && aux < 30){
17     Consumidor consumidor;
18
19     //Codigo para receber o tipo
20     do{
21         printf("\nEscolha o tipo de consumidor\n\n");
22         printf("1 - Residencial\n");
23         printf("2 - Comercial\n");
24         printf("3 - Industrial\n");
25         printf("0 - Mostrar relatorio e sair do programa\n:");
26         scanf("%d", &consumidor.tipo);
27         if (consumidor.tipo > 3 || consumidor.tipo < 0) {
28             printf("\nOPCAO INVALIDA!\n");
29         }
30
31     }while(consumidor.tipo > 3 || consumidor.tipo < 0);
32
33     if (consumidor.tipo == 0) {
34         break;
35     }
36
37     //Codigo para receber codigo do consumidor
38     printf("\nCodigo do consumidor: ");
39     scanf("%d", &consumidor.codigo);
40
41     //Codigo para receber consumo
42     printf("\nQuantidade de kWh consumidos no mes: ");
43     scanf("%f", &consumidor.qtdkwh);
44
45     //Salvar Consumidor na lista
46     consumidores[aux] = consumidor;
47     qtdConsumidores++;
48     aux++;
49
50 }
51 //Mostrar dados requeridos e finalizar programa
52 printf("\n===== Relatorio =====\n");
53
54 //Menor consumo Residencial
55 Consumidor *menorConsumoResidencial = NULL;
56
57 for(int i = 0; i < qtdConsumidores; i++)
58 {
59     if (consumidores[i].tipo == 1) {
60         if (menorConsumoResidencial == NULL){
61             menorConsumoResidencial = &consumidores[i];
62         }
63         else
64             if (consumidores[i].qtdkwh < menorConsumoResidencial->qtdkwh)
65                 menorConsumoResidencial = &consumidores[i];
66     }
67 }
68 if(menorConsumoResidencial != NULL){
69     printf("\nMenor consumo residencial\n");
70     printf("Codigo: %d\n", menorConsumoResidencial->codigo);
71     printf("Consumo: %.1fkWh\n", menorConsumoResidencial->qtdkwh);
72 }else
```

```
73 {
74     printf("\nNao houve consumo residencial\n");
75 }
76 //maior consumo Comercial
77 Consumidor *maiorConsumoComercial = NULL;
78
79 for(int i = 0; i < qtdConsumidores; i++)
80 {
81     if (consumidores[i].tipo == 2) {
82         if (maiorConsumoComercial == NULL){
83             maiorConsumoComercial = &consumidores[i];
84         }
85         else
86             if (consumidores[i].qtdkwh > maiorConsumoComercial->qtdkwh)
87                 maiorConsumoComercial = &consumidores[i];
88     }
89 }
90 if (maiorConsumoComercial != NULL) {
91     printf("\nMaior consumo comercial\n");
92     printf("Codigo: %d\n", maiorConsumoComercial->codigo);
93     printf("Consumo: %.1fkWh\n", maiorConsumoComercial->qtdkwh);
94 }else{
95     printf("\nNao houve consumo comercial\n");
96 }
97 //Media consumo industrial
98 float mediaConsumoIndustrial;
99 int qtdIndustrial = 0;
100
101 for(int i = 0; i < qtdConsumidores; i++)
102 {
103     if (consumidores[i].tipo == 3) {
104         qtdIndustrial++;
105         mediaConsumoIndustrial+=consumidores[i].qtdkwh;
106     }
107 }
108 mediaConsumoIndustrial /=qtdIndustrial;
109 printf("\nMedia de consumo Comercial: %.1fkWh\n\n", mediaConsumoIndustrial);
110
111 return 0;
112 }
```

- 4** Escreva um algoritmo que leia um número inteiro e escreva como saída “divisível por 3 e 7” se o número for divisível por 3 e por 7 ou “não é divisível por 3 e 7” caso contrário.

Solução

```
1 #include <stdio.h>
2
3 int main(int argc, char const *argv[])
4 {
5     int n;
6     printf("Valor: ");
7     scanf("%i", &n);
8
9     if (n%3 == 0 && n%7 == 0)
10         printf("Divisivel por 3 e 7");
```

```
11     else
12         printf("Nao divisivel por 3 e 7");
13     return 0;
14 }
```

- 5** A Prefeitura de Patos abriu uma linha de crédito para os funcionários estatutários. O valor máximo da prestação de um empréstimo não pode ultrapassar 30% do salário bruto do funcionário. Escreva um algoritmo que leia o nome de um funcionário, seu salário bruto e o valor da prestação do empréstimo que ele solicitou e, em seguida, escreva como saída o nome do funcionário seguido da mensagem “teve o crédito concedido” se o empréstimo solicitado puder ser concedido ou seguido da mensagem “crédito negado” caso contrário.

Solução

```
1  #include <stdio.h>
2
3  int main(int argc, char const *argv[])
4  {
5      char nome[30];
6      float salario, prestacao;
7
8      printf("Nome: ");
9      scanf("%s", &nome);
10     printf("Salario bruto: ");
11     scanf("%f", &salario);
12     printf("Valor da prestacao do emprestimo: ");
13     scanf("%f", &prestacao);
14     printf("Enos, voce teve o ");
15     if (prestacao <= 0.3f * salario)
16         printf("credito concedido\n");
17     else
18         printf("credito negado\n");
19     return 0;
20 }
```

- 6** Escreva um algoritmo que leia três números inteiros positivos, n, a e b, e escreva, em ordem crescente, os n primeiros inteiros positivos que são múltiplos de a ou b ou ambos. Por exemplo, se n = 6, a = 2 e b = 3, o algoritmo deve escrever como saída os números 2, 3, 4, 6, 8 e 9.

Solução

```
1  #include <stdio.h>
2
3  int main(int argc, char const *argv[])
4  {
5      int n, a, b, contador = 0, aux = 1;
6      printf("Valor de n: ");
7      scanf("%d", &n);
8      printf("Valor de a: ");
9      scanf("%d", &a);
10     printf("Valor de b: ");
11     scanf("%d", &b);
```

```
12     printf("Esses sao os %d primeiros multiplos de %d ou %d: ", n, a, b);
13     while( contador != n){
14         if (aux%a == 0 || aux%b == 0) {
15             printf(" %d ",aux);
16             contador++;
17         }
18         aux++;
19     }
20     return 0;
21 }
```

7 O número 3025 goza da seguinte propriedade:

$$30 + 25 = 55$$

$$55^2 = 3025$$

Escreva um algoritmo determine e escreva todos os números de quatro dígitos que possuem a propriedade acima. Note que este algoritmo não possui nenhum dado de entrada.

Solução

```
1  #include <stdio.h>
2
3  int main(int argc, char const *argv[]){
4      for(int i = 10; i < 100; i++){
5          for(int j = 0; j < 100; j++){
6
7              if ((i+j)*(i+j) == (i*100)+j) {
8                  printf("%d\n", (i+j)*(i+j));
9              }
10         }
11     }
12 }
```

8 Calcule a média final (usando a ponderação da UEPB) dadas as notas das 3 avaliações e que produza uma saída com a média e a situação do aluno de acordo com o seguinte critério: média ≥ 7 , aprovado; $5 \leq$ média < 7 , recuperação; média < 5 , reprovado.

Solução

```
1  #include <stdio.h>
2
3  int main(int argc, char const *argv[]){
4      float a1, a2, a3, media;
5
6      //Recebe nota das avaliacoes do usuario
7      printf("Informe as notas de tuas avaliacoes:\nAvaliacao 1: ");
8      scanf("%f", &a1);
9      printf("Avaliacao 2: ");
10     scanf("%f", &a2);
11     printf("Avaliacao 3: ");
```

```
12     scanf("%f", &a3);
13
14     //Faz o calculo da media e exibe
15     media = (a1 + a2 + a3)/3;
16     printf("A media e %.1f\n", media);
17
18     //Calcular a situacao do aluno
19     printf("Situacao: ");
20     if (media >= 7) {
21         printf("APROVADO");
22     }else{
23         if (media >= 5) {
24             printf("RECUPERACAO");
25         }else{
26             printf("REPROVADO");
27         }
28     }
29 }
```