

## AValiação RECUPERAÇÃO CONTÍNUA PROGRAMAÇÃO E ALGORITMO

**NOMES:** Matheus Latancio Polidoro

**DATA:** 07/12/2021

**Insira os códigos nos espaços entre os enunciados dos exercícios.**

- 1- Receba o salário de uma pessoa e o consumo de metros cúbicos de água da residência da mesma. Se o salário for menor que 1000 reais e o consumo for menor que 10 metros de água. Conceder um desconto de 15%. Se não conceder um desconto de 5%.

```
1 #include<stdlib.h>
2 #include<stdio.h>
3 #include<string.h>
4 #include<locale.h>
5
6 int main(){
7     // Desconto em porcentagem
8     setlocale(LC_ALL, "portuguese");
9     float salario, con_metros_cubicos, des_con_metros_cubicos;
10
11     printf("Informe o valor do salário: ");
12     scanf("%f", &salario);
13
14     printf("Informe o consumo de metros cubicos de água: ");
15     scanf("%f", &con_metros_cubicos);
16
17     if (salario < 1000 && con_metros_cubicos < 10){
18         des_con_metros_cubicos = con_metros_cubicos * 0.85;
19         printf("Conforme valores informados, você terá 15% de desconto.\n");
20     }
21     else {
22         des_con_metros_cubicos = con_metros_cubicos * 0.95;
23         printf("Conforme valores informados, você terá 5% de desconto.");
24     }
25
26     printf("De %.2f metros cúbicos consumidos, você pagara por %.2f.\n",
27         con_metros_cubicos, des_con_metros_cubicos);
28     system("pause");
29 }
```

- 2- Desenvolva um programa que sorteie de forma aleatória, números entre 1 e 60. Mostre quantas vezes um determinado número **INFORMADO PELO USÁRIO FOI SORTEADO**.

```
1 #include<stdio.h>
2 #include<stdlib.h>
3 #include<locale.h>
4 #include<conio.h>
5 #include <time.h>
6
7 int main(){
8     // Sorteio de números aleatórios
9     setlocale(LC_ALL, "portuguese");
10     int numero_escolhido, qtd_acerto, numero_sorteado[5];
11     srand(time(NULL));
12
13     printf("\nEntre com seu número da sorte para saber quantas vezes ele será sorteado:
14 ");
15     scanf("%i", &numero_escolhido);
16
17     for(int i = 0; i < 5; i++){
18         numero_sorteado[i] = rand() % 60;
19         if (numero_sorteado[i] == numero_escolhido){
20             qtd_acerto++;
21         }
22     }
23 }
```

## Etec Dr Emílio Hernandez Aguilar

```
22     printf("Os números sorteados foram: ");
23     for(int i = 0; i < 5; i++){
24         printf("%i, ", numero_sorteado[i]);
25     }
26
27     if (qtd_acerto != 1){
28         printf("\nSeu numero: %i foi sorteado %i vezes.\n", numero_escolhido, qtd_acerto);
29     }
30     else{
31         printf("\nSeu numero: %i foi sorteado %i vez.\n", numero_escolhido, qtd_acerto);
32     }
33     system("pause");
34 }
```

### 3- Receba uma palavra e escreva como o exemplo abaixo.

Palavra recebida.  
ETEC

E  
ET  
ETE  
ETEC

```
1  #include<stdlib.h>
2  #include<stdio.h>
3  #include<string.h>
4  #include<locale.h>
5
6  int main(){
7      // Iteração em String
8      setlocale(LC_ALL, "portuguese");
9      char palavra[30];
10
11     printf("Entre com uma palavra qualquer: ");
12     gets(palavra);
13     printf("Palavra recebida.\n%s\n\n", palavra);
14
15     for(int i = 0; i < strlen(palavra); i++){
16         for(int c = 0; c <= i; c++){
17             printf("%c", palavra[c]);
18         }
19         printf("\n");
20     }
21     system("pause");
22 }
```

### 4- Escreva na tela do usuário a seguinte sequência com uso de laço de repetição:

1.2.3.4.PAM  
6.7.8.9.PAM  
11.12.13.14.PAM  
16.17.18.19. PAM  
21.22.23.24. PAM  
26.27.28.29. PAM  
31.32.33.34. PAM  
36.37.38.39. PAM

```
1 #include<stdlib.h>
2 #include<stdio.h>
3 #include<string.h>
4 #include<locale.h>
5
6 int main(){
7     // Laço de repetição
8     setlocale(LC_ALL, "portuguese")
9
10    for(int i = 1; i <= 40; i++){
11        if (i % 5 != 0){
12            printf("%i.", i);
13        }
14        else{
15            printf("PAM\n");
16        }
17    }
18    system("pause");
19 }
```

- 5- Receber uma senha para cadastro. Se o a quantidade de caracteres for menor que 7 caracteres, impedir o cadastro até que a senha tenha uma quantidade maior ou igual a 7 dígitos.

```
1 #include<stdlib.h>
2 #include<stdio.h>
3 #include<string.h>
4 #include<locale.h>
5 #include<windows.h>
6
7 int main(){
8     // Validar senha
9     setlocale(LC_ALL, "portuguese");
10    char senha[30];
11
12    do{
13        printf("Entre com a senha: ");
14        gets(senha);
15        if (strlen(senha) < 7){
16            printf("Senha invalida.");
17            Sleep(1000);
18            system("cls");
19        }
20    }while (strlen(senha) < 7);
21
22    printf("Senha cadastrada com sucesso.");
23    system("pause");
24 }
```

- 6- Comente forma breve para que serve cada linha de código abaixo. (Não precisa repetir linhas com o mesmo código)

```
1 // inclusão de referências as bibliotecas.
2 #include<stdio.h> // biblioteca de entrada e saída de dados (input/ output).
3 #include<stdlib.h> // biblioteca padrão do C (funções de controle de memória, processos,
4 // conversões e outras).
5 #include<string.h> // biblioteca de manipulação de Strings (textos).
6 #include<locale.h> // biblioteca de localização, utilizada para modificar padrão de
7 // formatos textuais (acentos).
8 #include<windows.h> // biblioteca de funções da API do Windows, macros comuns do Windows.
9
10 int main (){
11     setlocale(LC_ALL,"portuguese"); // definir o padrão textual para português.
12     char nome[30], nome1[30]; // declarar variáveis (nome, nome1) do tipo char (caractere)
13     // como array (vetor) de 30 espaços.
```

## Etec Dr Emílio Hernandez Aguilar

```
11  int letras, i; // definir variáveis (letras, i) do tipo int (números inteiros).
12  do{ // início do laço de repetição (na primeira volta ele entra sem critério).
13      system("cls"); // limpeza do terminal do Windows.
14      printf("\nDigite seu nome:\n\a"); // exibe no terminal o texto entre (""), (\n)
    quebra de linha, (\a) som de alerta.
15      gets(nome); // pegar o valor digitado e incluir na variável (nome).
16      printf("\nConfirme valor Digitado\n\a");
17      gets(nome1);
18      // enquanto a condição for VERDADE (True) repetira as instruções entre do{ }while.
19      }while((strlen(nome)<6)|| (strlen(nome1)<6)); // enquanto (tamanho de nome for menor
    que 6 OU tamanho de nome1 for menor que 6) REPITA.
20
21      // estrutura condicional.
22      // strcmp compara o valor das strings e devolve um inteiro = 1(1ª texto maior), -1(2ª
    texto maior) ou 0(textos iguais).
23      if(!strcmp(nome,nome1)){ // SE NÃO ("!" é uma negação) strcmp for 0 (false) o valor
    vira (true).
24          printf("\n valores iguais\n");
25      }else{
26          printf("\n valores diferentes\n");
27      }
28      system("pause"); // aguarda que o usuário digite qualquer tecla para encerrar o
    programa e sair do terminal.
29 }
```

### 7- Qual a diferença do vetor em C para matriz em C.

**Vetor (array)** é unidimensional, ou seja, possui apenas largura, os espaços são alinhados um ao lado do outro, como um trem.

**Matriz (array multi-dimensional)** é um vetor de vetores, possui altura e largura, os espaços são empilhados um em cima do outro e alinhados um ao lado do outro, como um prédio.

### 8- O que é função e procedimento em linguagem de programação.

**Procedimento:** Bloco de código com início e fim que ao ser chamado é executado.

**Função:** bloco de código especial, com início e fim, pode possuir parâmetros de entrada e retornam um valor no nome da função que passa a ser a variável da expressão.

### 9- Cite um exemplo onde função ou procedimento são importantes ou necessários?

Na reutilização de códigos as funções e procedimentos são necessárias, deixando o código mais limpo e organizado.

### 10- Qual sua avaliação em aprendizagem no componente curricular de programação e algoritmo? Entre com uma menção MB, B, R ou I.

**MB**