

Natureza da Avaliação	Valores		Disciplina: Algoritmos e Programação II	Curso	Identificação da Turma		
	Valor	Nota			Turno	Período	Turma
Revisional			Prof. Maíza Cristina de Souza Dias	SISTEMAS DE INFORMAÇÃO	NOT	4º	ÚNICA
NOME: Rivail Moreira da Silva						DATA: 01/09/2021	

## REVISÃO / DIAGNÓSTICO LISTA 02

### Laço Do While

1. Crie um programa que leia um número do teclado até que encontre um número igual a zero. No final, mostre a soma dos números digitados.

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

int main(){
    int soma, num;
    soma = 0;

    do {
        printf("Digite um numero, informe ZERO(0) para sair:");
        scanf("%d",&num);

        soma = soma + num;

    } while (num != 0);

    printf("\nA Soma de todos numeros digitado(s) foi = %d",soma);
    printf("\n\n");

    getchar();
    return 0;
}
```

2. Escreva um algoritmo que leia um número do teclado até que encontre um número menor ou igual a 1.

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

int main()
{
    int num=0 ;

    do{
        printf("Digite um numero:");
        scanf("%d",&num);

    }while(num > 1);

    printf("O ultimo numero <= 1 digitado foi:%d",num);
    getchar();
    return 0;
}
```

## Vetores

3. Elaborar um algoritmo que armazene o intervalo de 0 a 100 e armazene em um vetor.

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

int main()
{
    int i, vet[100];

    for(i=0; i < 100; i++){
        vet[i]=i+1;
    }
    for(i=0; i<100; i++){
        printf("%d \t", vet[i]);
    }
    return 0;
}
```

4. Escreva um algoritmo que armazene em um vetor todos os números de 100 até 1 (ordem decrescente). Após a leitura o programa deverá imprimir os valores lidos.

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

int main(){
    int i, vet[100];

    for(i=100; i>0; i--){
        vet[i]=i;
    }
    for(i=100; i>0; i--){
        printf("%d \t", vet[i]);
    }

    return 0;
}
```

5. Elaborar um algoritmo que leia 5 valores e armazene-os em um vetor.

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

int main()
{ int i, vet[5];

    for(i=0; i < 5; i++){
        printf("Digite o numero posicao VET[%d]: ", i);
        scanf("%d", &vet[i]);
    }

    printf("\nOs vetores foram preenchidos com os numeros:\n");
    for(i=0; i < 5; i++){
```

```
printf("\nVET[%d]: %d",i,vet[i]);
}
printf("\n");
return 0;
}
```

6. Elabore um algoritmo que leia dois vetores (A e B) de 5 posições e os preencha com valores inteiros. Mostre os números e armazene a soma de  $A[n]+B[n]$  em um vetor C.

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

int main()
{
    int soma,i,A[5],B[5],C[5];

    for(i=0; i<5; i++){
        printf("Informe do A[%d] numero:",i+1);
        scanf("%d",&A[i]);
    }
    printf("Numeros de A preenchido com SUCESSO!\n\n");

    for(i=0; i<5; i++){
        printf("Informe do B[%d] numero:",i+1);
        scanf("%d",&B[i]);
    }
    printf("Numeros de B preenchido com SUCESSO!\n");

    for(i=0; i<5; i++){
        C[i] = A[i] + B[i];
    }
    for(i=0; i<5; i++){
        soma=C[i];
        printf("\nSoma A[%d] + B[%d]: %d",i+1,i+1,soma);
    }
    printf("\n\n");
    return 0;
}
```

7. Elabore um algoritmo que leia valores em oito posições. Após a leitura, solicitar número do teclado. Pesquisar se este número existe no vetor. Se existir, imprimir em qual posição do vetor ela está. Se não existir, imprimir a mensagem que não existe.

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

int main(){
    int i, j = 0, num = 0, vet[8], a[8];

    for(i=0; i < 8; i++){
        printf("Digite os numeros para preencher os vetores vet[%d]:", i);
        scanf("%d", &vet[i]);
    }

    printf("\nInforme o numero que deseja verificar:");
    scanf("%d", &num);

    for(i = 0; i < 8; i++){
        if (vet[i] == num){
            a[j] = i;
            j++;
        }
    }

    if (j > 0){
        printf("\nO numero digitado existe nos indices:");

        for(i = 0; i < j; i++){
            printf("\nvet[%d] = %d", a[i], num);
        }
    }else{
        printf("\nEsse numero nao existe no indice do vetor!");
    }

    printf("\n\n");
    return 0;
}
```

8. O coordenador do curso deseja saber a maior nota de cada aluno. Sabendo-se que eles têm duas disciplinas – Programação e Práticas, faça um algoritmo que leia as notas as duas disciplinas e retorne em qual disciplina o aluno obteve a maior nota.

```
#include<stdio.h>
#include<stdlib.h>
int main (){
    float pratica , programacao;
    char a = 'y';
    char nome[15];

    while (a == 'y'){
        printf("Digite o nome do aluno :\\n");
        scanf("%s",nome);
        printf("Digite a nota do aluno na disciplina Programacao:\\n");
        scanf("%f",&programacao);
        printf("Digite a nota do aluno na disciplina Pratica:\\n");
        scanf("%f",&programacao);

        if ( programacao > pratica ){printf("A maior nota do aluno %s foi em Programacao com %.2f\\n", nome,
            programacao);}
        if ( programacao < pratica ){printf("A maior nota do aluno %s foi em Pratica com %.2f\\n", nome, pratica);}
        if ( programacao == pratica ){printf("A maior nota do aluno %s foi igual nas duas disciplinas com %.2f\\n",
            nome , programacao);}
        printf("Digite y se deseja fazer outra verificacao ou n se deseja sair: ");
        scanf( " %c", &a);
    }
    return 0;
}
```

9. A série de Fibonacci é formada pela sequência: **1, 1, 2, 3, 5, 8, 13, 21, 34, 55, ...** Escreva um algoritmo em PORTUGOL que armazene em um vetor os 50 primeiros termos da série de FIBONACCI. Após isso, o algoritmo deve imprimir todos os valores armazenados.

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

int main (){
    double fibonacci[50];
    setlocale(LC_ALL,"Portuguese");
    fibonacci[0] = fibonacci[1] = 1;
    for (int i = 2 ; i < 50 ; i++){//armazenamento dos números da sequencia fibonacci
        fibonacci[i] = fibonacci[i-1] + fibonacci[i-2];
    }
    printf("A sequência fibonacci de 50 números são:\\n");
    printf("%.lf",fibonacci[0]);
    for (int i = 1 ; i < 50 ; i++){//apresentação dos numeros
        printf(" , %.lf",fibonacci[i]); } return 0;
}
```

10. Em uma cidade do interior, sabe-se que, de janeiro a abril de 2021 (121 dias), não correu temperatura inferior a 15°C nem superior a 40°C. As temperaturas verificadas em cada dia estão disponíveis em uma unidade de entrada de dados. Fazer um algoritmo que calcule e imprima:

- A menor temperatura ocorrida;
- A maior temperatura ocorrida;
- A temperatura média;
- O número de dias nos quais a temperatura foi inferior à temperatura média.

```
#include<stdio.h>
#include<stdlib.h>
int main()
{
    int n;
    int a = 121;
    int vet[121] = {44, 24, 18, 25, 16, 44, 20, 45, 40, 35, 44, 41, 26, 26, 23, 26, 43, 17, 45, 15, 40, 32, 41,
                    40, 38, 17, 42, 30, 37, 42, 21, 19, 35, 44, 23, 42, 38, 45, 35, 26, 37, 18, 42, 45, 29, 36, 29, 33, 24, 29,
                    45, 45, 27, 16, 33, 37, 27, 30, 41, 21, 30, 34, 39, 19, 16, 23, 44, 44, 34, 43, 38, 29, 25, 27, 32, 39, 37,
                    29, 40, 21, 31, 45, 42, 15, 44, 22, 36, 20, 32, 33, 20, 29, 41, 17, 39, 35, 18, 35, 36, 25, 30, 34, 33, 26,
                    39, 35, 41, 31, 21, 28, 20, 34, 39, 33, 21, 43, 38, 38, 43, 38, 42};
    int i,menor,maior,media,cont=0;

    for(i=0;i<a;i++){
        printf("Insira o valor:\n");
        scanf("%d",&n);
        vet[i] = n;
    }
    if(i==a){
        for(i=0;i<a;i++){
            printf("\nvet[%d]",vet[i]);
        }
    }
    menor = vet[0];
    for(i=0;i<a;i++){
        if(vet[i] < menor){
            menor = vet[i];
        }
    }

    maior = vet[0];
    for(i=0;i<a;i++){
        if(vet[i] > maior){
```

```
}  
}  
  
media = 0;  
for(i=0;i<a;i++){  
  
    media += vet[i];  
    if(vet[i] > maior){  
        maior = vet[i];  
    }  
}  
  
for(i=0;i<a;i++){  
  
    if(vet[i] < (media/a)){  
        cont ++;  
    }  
}  
  
printf("O menor e: %d", menor);  
printf("\nO maior e: %d", maior);  
printf("\nA media e: %d", media/a);  
printf("\nO numero de dias que a temperatura foi menor que a media e: %d", cont);  
  
return 0;  
}
```

### Manipulação de Strings

11. Escrever seu nome na tela 10 vezes. Um nome por linha.

```
#include<stdio.h>  
#include<stdlib.h>  
  
int main(){  
    int i;  
    char nome[15];  
  
    printf("Digite seu nome:");  
    scanf("%s",&nome);  
  
    for(i=0;i<10;i++){  
        printf("\n%d Nome: %s",i+1,nome);
```

```
}
printf("\n");
return 0;
}
```

### Matrizes

12. Criar um algoritmo que leia os elementos de uma matriz inteira 4 x 4 e escreva os elementos da diagonal principal.

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

int main(){
    int i, j, mat[4][4];

    for (i=0; i < 4; i++){
        for (j=0; j < 4; j++){
            printf("Digite o valor de mat[%d][%d]:", i+1, j+1);
            scanf("%d", &mat[i][j]);
        }
        printf("\n");
    }
    printf("\nDiagonal principal da MATRIZ:\n\n");
    for (i=0; i < 4; i++){
        for (j=0; j < 4; j++){

            if(i == j){
                printf("%d \t", mat[i][j]);
            }else{
                printf("\t");
            }
        }
        printf("\n");
    }
    return 0;
}
```

13. Criar um algoritmo que leia os elementos de uma matriz inteira 10 x 10 e escreva todos os elementos, exceto os elementos da diagonal principal.

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

int main(){
    int i, j, mat[10][10];

    for (i=0; i < 10; i++){
        for (j=0; j < 10; j++){
            printf("Digite o valor de mat[%d][%d]:", i+1, j+1);
            scanf("%d", &mat[i][j]);
        }
        printf("\n");
    }
    printf("\nMATRIZ preenchida!!! Exceto os elementos da diagonal principal. :\n\n");
    for (i=0; i < 10; i++){
```



```
for (j=0;j < 10 ; j++){

    if(i != j){
        printf("%d \t",mat[i][j]);
    }else{
        printf(" \t");
    }
}
printf("\n");
}
return 0;
}
```

14. Criar um algoritmo que leia os elementos de uma matriz inteira 10 x 10 e escreva somente os elementos acima da diagonal principal.

```
#include<stdio.h>
#include<stdlib.h>
const int A =10;

int main(){
    int i, j, mat[A][A];

    for(i = 0; i < A; i++){
        for(j = 0; j < A; j++){
            printf("Informe o valor mat[%d][%d]:",i+1,j+1);
            scanf("%d",&mat[i][j]);

        }
        printf("\n");
    }
    printf("Matriz preenchida com sucesso!!! Print Elementos acima da diagonal principal.\n\n ");
    for(i = 0; i < A; i++){
        for(j = 0; j < A; j++){
            if(i != j && i < j+1){
                printf("%d \t",mat[i][j]);
            }else{
                printf("X \t");
            }
        }
        printf("\n");
    }
    return 0;
}
```

15. Criar um algoritmo que leia os elementos de uma matriz inteira 10 x 10 e escreva somente os elementos abaixo da diagonal principal.

```
#include<stdio.h>
#include<stdlib.h>
const int A = 10;

int main(){
    int i, j, mat[A][A];

    for(i = 0; i < A; i++){
        for(j = 0; j < A; j++){
            printf("Informe o valor mat[%d][%d]:", i+1, j+1);
            scanf("%d", &mat[i][j]);
        }
        printf("\n");
    }
    printf("Matriz preenchida com sucesso!!! Print Elementos acima da diagonal principal.\n\n");
    for(i = 0; i < A; i++){
        for(j = 0; j < A; j++){
            if(i != j && j < i+1){
                printf("%d \t", mat[i][j]);
            }else{
                printf("X \t");
            }
        }
        printf("\n");
    }
    return 0;
}
```