

Mas a vereda dos justos é como a luz da aurora, que vai brilhando mais e mais até ser dia perfeito.

Provérbios 4:18

PROGRAMAÇÃO ORIENTADA A OBJETOS

► Prof. José Antonio Gonçalves

Nestes slides

Aclimação ao Java: aplicação das estruturas clássicas de programação:

- laços;
- seletores;
- variáveis;
- operações matemáticas;
- operações com cadeia de caracteres (Strings);
- operações com datas;
- entrada de dados via teclado;
- etc.

Aclimação do Java: aplicação das estruturas clássicas de programação

Aplicação das estruturas clássicas de programação

Saída de dados (na tela)

```
public class Saida {  
    public static void main(String arg[]){  
        System.out.println("Facim, facim!!!");  
    }  
}
```

Exerc: Usando o modelo anterior:

- a.1)_Acrescente mais uma linha de exibição no programa anterior.
- a.2)_Crie um programa que imprima todos os seus dados cadastrais
- a.3)_Crie um programa que imprima a frase "Meu nome é: " que através de parâmetros passados durante a chamada (interpretação: java ...) do programa, imprima na tela esta frase + o nome passado.

Aplicação das estruturas clássicas de programação

Declaração de Variáveis e Operadores Matemáticos

```
public class Operad {  
  
    public static void main(String arg[]){  
        System.out.println("Impressão de Resultado de Cálculos:");  
        System.out.println(" ");  
        int a=3,b=2;  
        int c=a+b;  
        System.out.println("O resultado da soma de A e B é:" + c);  
    }  
}
```

Exerc: Usando o modelo anterior:

b.1)_ Implemente as outras operações matemática “simples” a saber

Multiplicação (*), divisão (/) e subtração (-), sendo com primeiro termo da operação a variável “a” e o segundo termo a variável “b”, imprimindo seu resultado a cada operação.

Aplicação das estruturas clássicas de programação

Laços (for)

```
public class Laco1 {  
    public static void main(String arg[]){  
        int a=0;  
        int b=10;  
        for(a=0; a<=b; a++){  
            System.out.println("O Valor de A dentro do FOR é: "+a);  
        }  
    }  
}
```

Exerc: Usando o modelo anterior:

c.1)_ Usando o laço “for”, construa um programa que ao invés de incrementar a variável “a” até que atinja o valor de da variável “b”, faça o inverso, decemente “b” até que acheque ao valor de “a”.

Aplicação das estruturas clássicas de programação

Laços (while)

```
public class Laco2 {  
    public static void main(String arg[]){  
        int a=0,  
        b=15;  
        while(a<=b){  
            a++;  
            System.out.println("O Valor de A dentro do while é: "+a);  
        }  
    }  
}
```

Exerc: Usando o modelo anterior:

d.1)_ Da mesma forma que implementou o exercício c.1,
implemente o d.1 usando a estrutura de repetição “while”

Aplicação das estruturas clássicas de programação

Laços (do..while)

```
public class Laco3 {  
    public static void main(String arg[]){  
        int a=0,  
        b=15;  
        do{  
            System.out.println("O valor de A dentro do DO é: "+a);  
            a++;  
        }while(a<=b);  
        System.out.println("");  
    }  
}
```

Exerc: Usando o modelo anterior:

e.1)_ Da mesma forma que implementou o exercício c.1,
implemente o e.1 usando a estrutura de repetição “do..while”

Aplicação das estruturas clássicas de programação

Seleção (if..elseif..else)

```
public class Selec1 {  
    public static void main(String arg[]){  
        int a=0, b=20;  
        for(a=0; a<=b; a++){  
            if(a==(b/2)){  
  
                System.out.println("A é igual a metade de B, logo A vale: "+a+" e B vale: "+b);  
  
            }  
            else if(a!=(b/2)){  
                System.out.println("A é diferente a metade de B, logo A vale: "+a+" e B vale: "+b);  
            }  
        }  
    }  
}
```

Exerc: Usando o modelo anterior:

f.1)_ Construa um programa que leia 2 valores inteiros e diga se o 1º é: maior, igual ou menor que o 2º valor.

Aplicação das estruturas clássicas de programação

Seleção (ternário)

```
public class Tern{  
    public static void main(String args[]){  
        int a=5;  
        int b=2;  
        int c = (a>b ? a : b);  
        System.out.println(c);  
        // ou → System.out.println(a>b ? a : b );  
    }  
}
```

Aplicação das estruturas clássicas de programação

Seleção (switch..case):

```
public class Selec2 {  
    public static void main(String arg[]){  
        int valor = 1;  
        switch(valor){  
            case 0:  
                System.out.println("Primeira Opção (Valor igual a zero) "+valor);  
                break;  
            case 1:  
                System.out.println("Segunda Opção (Valor igual a um) "+valor);  
                break;  
            default:  
                System.out.println("Outras Opções (Valor maior que um) "+valor);  
                break;  
        }  
    }  
}
```

Aplicação das estruturas clássicas de programação

Operadores Lógicos “e” e “ou” (&& e ||)

```
public class Logic {  
    public static void main(String arg[]){  
        int a=0, c=0, valor=15;  
        float b=15;  
        for(a=0; a<=b; a++){  
            valor--;  
            c++;  
            if(c<=(b/2)&& c<=valor){  
                System.out.println("C eh MENOR que a metade de B 'E' MENOR IGUAL a VALOR");  
                System.out.println("C vale.....: "+c);  
                System.out.println("B / 2 vale.....: "+(b/2));  
                System.out.println("VALOR vale...: "+valor);  
            }else if (c>=(b/2)|| (c>=valor)){  
                System.out.println("C eh MAIOR que a metade de B 'OU' MAIOR IGUAL a VALOR");  
                System.out.println("C vale.....: "+c);  
                System.out.println("B / 2 vale.....: "+(b/2));  
                System.out.println("VALOR vale...: "+valor);  
            }  
        }  
    }  
}
```

Aplicação das estruturas clássicas de programação

Entrada de dados via console

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;

public class EntDados {
    public static void main(String arg[]){
        InputStreamReader c = new InputStreamReader(System.in);
        BufferedReader cd = new BufferedReader(c);
        String s = ""; String t = "";
        System.out.println("Entre com o 1º valor: ");
        try{
            s = cd.readLine();
        }
        catch(IOException e){
            System.out.println("Erro de entrada");
        }
        System.out.println("Entre com o 2º valor: ");
        try{
            t = cd.readLine();
        }
        catch(IOException e){
            System.out.println("Erro de entrada");
        }
        System.out.println("A soma dos dois valores eh : "+(s+t));
    }
}
```

Aplicação das estruturas clássicas de programação

Entrada de dados via console (convertendo “tipo”):

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;

public class ConvertNum {
    public static void main(String arg[]){
        System.out.println("Entre com um valor: ");
        InputStreamReader c = new InputStreamReader(System.in);
        BufferedReader cd = new BufferedReader(c);
        String s = "";
        try{
            s = cd.readLine();
        }
        catch(IOException e){
            System.out.println("Erro de entrada");
        }
        int w = Integer.parseInt(s);
        System.out.println("O valor de entrada foi: "+w);
    }
}
```

Aplicação das estruturas clássicas de programação

Exercícios:

G1)_ Construa um programa que:

a)_ Permita o usuário entrar com 2 valores (que sejam armazenados em duas variáveis, por exemplo, “a” e “b”);

b)_ Apresente “menu de opções” com 4 opções:

Exemplo:

Menu de Opções:

1 – Somar (a+b)

2 – multiplicar (a*b)

3 – subtrair (a-b)

5 – dividir (a/b)

Escolha uma opção:

c)_ Após o usuário escolher uma opção, apareça uma linha informando o resultado do cálculo que foi escolhido:

Exemplo (sendo escolhida a 1ª opção e “a” valendo 2 e “b” valendo 1):

“A soma de a + b é: 3”

Para tanto use as estruturas de entrada de dados via console, conversão de tipos e a estrutura de seleção switch..case.

Aplicação das estruturas clássicas de programação

Alguns métodos da Classe Math.

```
public class Maths1 {  
    public static void main(String arg[]){  
  
        double f=2;  
        double g=1.34;  
        double h=9;  
  
        System.out.println("\nO seno da var 'f' eh: " + Math.sin(f));  
        System.out.println("\nO cosseno da var 'f' eh: " + Math.cos(f));  
        System.out.println("\nA tangente da var 'f' eh: " + Math.tan(f));  
        System.out.println("\nO valor mínimo entre as vars 'f' 'g' eh: " + Math.min(f,g));  
        System.out.println("\nO valor máximo entre as vars 'f' 'g' eh: " + Math.max(f,g));  
        System.out.println("\nO valor arredondado da var 'g'(1.434) eh: " + Math.round(g));  
        System.out.println("\nO valor arredondado para cima da var 'g'(1.434) eh: " + Math.ceil(g));  
        System.out.println("\nO valor arredondado para baixo da var 'g'(1.434) eh: " + Math.floor(g));  
  
        System.out.println("\nA raiz da var 'h' eh: " + Math.sqrt(h));  
        System.out.println("\nA var 'f' elevada a var 'h' eh: " + Math.pow(f,h));  
        System.out.println("\nO valor absoluto da var 'g'(1.434) eh: " + Math.abs(g));  
        System.out.println("\nO valor do 'PI' eh: " + Math.PI);  
        System.out.println("\nGerar um valor aleatório ente 0 e 1: " + Math.random());  
  
        System.out.println("\n\nA próxima linha não faz parte da classe Math, trata-se do operador 'Módulo');  
        System.out.println("\nO resto da divisão entre as vars 'h' e 'f' eh: " + (f%h));  
    }  
}
```

Aplicação das estruturas clássicas de programação

Exercícios

h1)_ Utilizando a Linguagem de programação Java, implemente um programa que permita a entrada de um valor inteiro e informe se este valor é “par” ou “ímpar”.

Aplicação das estruturas clássicas de programação

Capturando a “data” do Sistema

```
import java.util.GregorianCalendar;

public class Data{
    public static void main(String arg[]){

        GregorianCalendar data = new GregorianCalendar(); //trabalhando com datas

        int dia = data.get(data.DAY_OF_MONTH);
        int mes = data.get(data.MONTH)+1;
        int ano = data.get(data.YEAR);
        System.out.println("\nDATA ATUAL DO SISTEMA: "+dia+"/"+mes+"/"+ano+"\n\n");
        System.out.println("\nDATA ATUAL DO SISTEMA:
"+data.get(data.DAY_OF_MONTH)+
"/"+data.get(data.MONTH)+"/"+data.get(data.YEAR)+"\n\n");
    }
}
```

Aplicação das estruturas clássicas de programação

Alguns métodos da Classe String

```
public class Stringe {  
    public static void main(String arg[]){  
        String frase="Eis-me aqui SENHOR, envia-me a miM";  
        System.out.println("\t\t\t\t\t Eis-me aqui SENHOR, envia-me a mim");  
        System.out.println("\n\n A frase anterior tem "+frase.length()+" caracteres");  
        System.out.println("\n\n Transformando-a toda em MAIUSCULA:");  
        System.out.print(frase.toUpperCase());  
        System.out.println("\n\n TRANSFORMANDO-A TODA EM minuscula:");  
        System.out.print(frase.toLowerCase());  
        System.out.println("\n\n Procurando a letra da 10 posição:");  
        System.out.print(frase.charAt(13));  
        System.out.println("\n\n Retornando a substring dentro da frase:");  
        System.out.println(frase.substring(0,11));  
        System.out.println("\n\n 1ª Vez que encontrou a letra 'O' na frase:");  
        System.out.println(frase.indexOf('O'));  
        System.out.println("\n\n Tirando os espaços em branco:");  
        System.out.println(frase.trim());  
        System.out.println("\n\n Trocando as letras 'm' por 'M':");  
        System.out.println(frase.replace('m','M'));  
    }  
}
```

Aplicação das estruturas clássicas de programação

Array unidimensional (vetor) – Entrada automática.

```
public class Vet1{
    public static void main(String arg[]){
        int larg=30;
        int vtr[]= new int[larg];
        System.out.println("O tamanho do vetor eh: "+vtr.length);
        int cont=0;
        for(int i=0;i<vtr.length;i++){
            cont=cont+2;
            vtr[i]=cont;
        }
        for(int i=0;i<vtr.length;i++){
            System.out.println("Vetor no local "+i+" tem valor "+vtr[i]);
        }
    }
}
```

Aplicação das estruturas clássicas de programação

Array Bidimensional (Matriz) – Entrada automática.

```
public class Mat1 {  
    public static void main(String arg[]){  
        int lin=5, col=5, cont=1, i = 0, c = 0;  
        int mtrz[][]=new int[lin][col];  
        System.out.println(mtrz.length); //tamanho da matriz  
        for(i =0; i<lin; i++){  
            for(c=0; c<col; c++){  
                mtrz[i][c]=cont++;  
            }  
        }  
  
        for(i =0; i<lin; i++){  
            for(c=0; c<col; c++){  
                System.out.println("Val. da "+ i + "linha eh "+ c + "coluna eh: "+ mtrz[i][c]);  
            }  
        }  
    }  
}
```

Aplicação das estruturas clássicas de programação

Exercícios (sempre utilizando a linguagem Java):

Strings

1)_ Construa um programa que permita ao usuário entrar com determinada frase, depois permita “escolher” uma letra qualquer e: caso a letra escolhida esteja na frase (seja maiúscula ou minúscula) diga quantas vezes ela apareceu e em que posição da frase. Senão, apareça uma frase informando que esta letra não existe na frase.

Matrizes

1)_ Crie um programa, no qual terá um vetor de inteiros, cujo tamanho será definido pelo valor de uma variável local, que permita ao usuário entrar com os valores. Depois, estes valores serão apresentados na ordem inversa à da entrada.

2)_ Faça o mesmo procedimento do exercício anterior, porém desta vez estará usando uma matriz bidimensional