

Revisão de Estruturas de Controle de Fluxo para
Estrutura de Dados

Programação Pascal

Esqueleto de Programa

```
Program Teste;  
  Type  
    ...  
  Var  
    ...  
Begin  
  ...  
End.
```

Variáveis

```
Program Teste;
```

```
...
```

```
Var
```

```
    a, i, j: integer;
```

```
    b: string;
```

```
    c: boolean;
```

```
    d: real;
```

```
    e: char;
```

```
...
```

Variáveis

Program Teste;

...

Begin

 a := 12;

 b := 'teste teste';

 c := true;

 d := 12.56;

 e := 'e';

...

Saída

Program Teste;

...

Begin

```
writeln('valor de a = ', a);
```

```
writeln('valor de b = ', b);
```

```
writeln('valor de c = ', c);
```

```
write('valor de d = ', d:10:2);
```

```
write('valor de e = ', e);
```

...

Entrada

```
Program Teste;
```

```
...
```

```
Begin
```

```
    write('Informe a: ');
```

```
    read(a);
```

```
...
```

Decisão

```
Program Teste;
```

```
...
```

```
Begin
```

```
    if a mod 2 = 0 then
```

```
        begin
```

```
            writeln('a e par');
```

```
        end
```

```
    else
```

```
        begin
```

```
            writeln('a e impar');
```

```
        end;
```

Decisão

```
Program Teste;
```

```
...
```

```
Begin
```

```
    case e of
```

```
        'a': writeln('intervalo maluco');
```

```
        'e': writeln('tres');
```

```
        'h': writeln('quatro');
```

```
        else writeln('diferente de todos');
```

```
    end;
```

```
    ...
```


Repetição

```
Program Teste;
```

```
...
```

```
Begin
```

```
    for i := 0 to 10 do
```

```
    begin
```

```
        writeln(i);
```

```
    end;
```

```
...
```

Repetição

```
Program Teste;
```

```
...
```

```
Begin
```

```
    j := 0;
```

```
    while j <= 10 do
```

```
    begin
```

```
        writeln(j);
```

```
        j := j + 1;
```

```
    end;
```

```
...
```

Repetição

```
Program Teste;
```

```
...
```

```
Begin
```

```
    j := 0;
```

```
    repeat
```

```
    begin
```

```
        writeln(j);
```

```
        j := j + 1;
```

```
    end;
```

```
    until j > 10;
```

```
    ...
```

Atividade 1

Instruções: Os materiais produzidos pelos estudantes devem ser entregues ao professor, em papel, impressos ou manuscritos, até a data estabelecida. De posse da apostila 1 (Implementação de Algoritmos), resolva cada um dos exercícios indicados, usando Linguagem Pascal, em arquivos separados e nomeados de forma que fique claro, a partir do nome do arquivo, a que exercício se refere:

- 12, 13 e 14 (página 23)
- 14, 15, 16, 17 e 18 (páginas 28 e 29)
- 2 e 3 (páginas 33 e 34)
- 10 a 17 (páginas 39 e 40)
- 10 a 17 (exercícios sobre FOR)

Estruturas de dados homogêneas – Arrays

`idades: array[1..10] of integer;`

`nomes: array[1..10] of string;`

`matriz: array[1..10, 1..10] of
integer;`

Estruturas de dados homogêneas – Arrays

```
for i := 1 to 10 do  
begin  
    write('Idade ', i, ':');  
    read(idade[i]);  
end;
```

Estruturas de dados homogêneas – Arrays

```
soma := 0;  
for i := 1 to 10 do  
begin  
    soma := soma + idade[i];  
end;  
write('Media: ', soma / i);
```

Estruturas de dados homogêneas – Arrays

```
for i := 1 to 10 do  
begin  
    writeln('Idade[', i, ']=',  
           idade[i]);  
end;
```


Estruturas de dados homogêneas – Ordenação de Arrays

```
qtd := <tamanho do array>;  
For l := 1 to qtd - 1 do  
  Begin  
    For c := l + 1 to qtd do  
      Begin  
        If nomes[l] > nomes[c] then  
          Begin  
            aux := nomes[l];  
            nomes[l] := nomes[c];  
            nomes[c] := aux;  
          end;  
        end;  
      end;  
    end;  
  end;  
end;
```

Estruturas de dados homogêneas – Arrays Bidimensionais

```
for i := 1 to 10 do
begin
  for j := 1 to 10 do
  begin
    ba[i, j] := sqr(i+j);
    write(ba[i, j]:5);
  end;
  writeln;
end;
```

Atividade 2

Instruções: Os materiais produzidos pelos estudantes devem ser entregues ao professor, em papel, impressos ou manuscritos, até a data estabelecida. De posse da apostila 2 (Técnicas de Programação II), resolva cada um dos exercícios indicados, usando Linguagem Pascal, em arquivos separados e nomeados de forma que fique claro, a partir do nome do arquivo, a que exercício se refere:

- 1 a 7 (página 10)
- 1 a 5 (página 13)
- 1 a 7 (página 16)

Estruturas de dados heterogêneas – Registros (Tipo)

```
Program Teste;  
  Type  
    ta = record  
      codigo: integer;  
      nome: string;  
      peso: real;  
    end;  
  
  Var  
    pessoa: ta;
```

Estruturas de dados heterogêneas – Registros (Tipo)

```

pessoa.codigo := 123;
pessoa.nome := 'Jose';
pessoa.peso := 45.5;

writeln(pessoa.codigo);
writeln(pessoa.nome);
writeln(pessoa.peso);

```

Estruturas de dados híbridas – Array de Registros

```
Program Teste;
```

```
  Type
```

```
    ta = record  
      codigo: integer;  
      nome: string;  
      peso: real;  
    end;
```

```
  Var
```

```
    pessoas: array[1..2] of ta;
```

Estruturas de dados híbridas – Array de Registros

```
peessoas[1].codigo := 123;  
peessoas[1].nome := 'Jose';  
peessoas[1].peso := 45.5;
```

```
peessoas[2].codigo := 124;  
peessoas[2].nome := 'Jose Amara';  
peessoas[2].peso := 34.5;
```

```
for i := 2 downto 1 do  
begin  
    writeln(peessoas[i].codigo);  
    writeln(peessoas[i].nome);  
    writeln(peessoas[i].peso);  
end;
```

Atividade 3

Instruções: Os materiais produzidos pelos estudantes devem ser entregues ao professor, em papel, impressos ou manuscritos, até a data estabelecida. De posse da apostila 2 (Técnicas de Programação II), resolva cada um dos exercícios indicados, usando Linguagem Pascal, em arquivos separados e nomeados de forma que fique claro, a partir do nome do arquivo, a que exercício se refere:

- 1 e 2 (página 21)