

CI1055: Algoritmos e Estruturas de Dados I

Profs. Drs. Marcos Castilho e Bruno Muller Jr

Departamento de Informática/UFPR

27 de julho de 2020

Resumo

Lógica de programação

Técnicas elementares de programação

- ▶ Até o momento, os problemas apresentados geraram programas simples e pequenos;
- ▶ Programas maiores requerem técnicas que:
 - ▶ minimizam erros de lógica
 - ▶ agilizam o desenvolvimento
- ▶ Técnicas elementares de programação:
 - ▶ Lógica de programação
 - ▶ Teste de mesa
 - ▶ Técnica dos acumuladores
 - ▶ Árvore de Decisão

Lógica de programação

Problema: Contar a quantidade de números digitados pelo usuário no teclado.

- ▶ questões para implementar a solução:
 - ▶ quando sabe que terminou de digitar?
 - ▶ quando for digitado zero: 5 8 3 9 0
 - ▶ números seguidos de <enter>

Lógica de programação

Problema: Contar a quantidade de números digitados pelo usuário no teclado.

- ▶ questões para implementar a solução:
 - ▶ quando sabe que terminou de digitar?
 - ▶ quando for digitado zero: 5 8 3 9 0
 - ▶ números seguidos de <enter>

Lógica de programação

Problema: Contar a quantidade de números digitados pelo usuário no teclado.

- ▶ questões para implementar a solução:
 - ▶ quando sabe que terminou de digitar?
 - ▶ quando for digitado zero: `5 8 3 9 0`
 - ▶ números seguidos de `<enter>`

Lógica de programação

Problema: Contar a quantidade de números digitados pelo usuário no teclado.

- ▶ questões para implementar a solução:
 - ▶ quando sabe que terminou de digitar?
 - ▶ quando for digitado zero: `5 8 3 9 0`
 - ▶ números seguidos de <enter>

Lógica de programação

- ▶ Como construir a lógica (algoritmo) que pode ser formalizada em um programa?
- ▶ Pensando como o computador faria:
 1. aguarda digitação de um número;
 2. “trata” número digitado;
 3. aguarda digitação do próximo número;
- ▶ Mapear a lógica acima para os comandos conhecidos.
- ▶ Quais, em que ordem (quebra-cabeça).
- ▶ O que colocar em “trata” número digitado?

Lógica de programação

- ▶ Como construir a lógica (algoritmo) que pode ser formalizada em um programa?
- ▶ Pensando como o computador faria:
 1. aguarda digitação de um número;
 2. “trata” número digitado;
 3. aguarda digitação do próximo número;
- ▶ Mapear a lógica acima para os comandos conhecidos.
- ▶ Quais, em que ordem (quebra-cabeça).
- ▶ O que colocar em “trata” número digitado?

Lógica de programação

- ▶ Como construir a lógica (algoritmo) que pode ser formalizada em um programa?
- ▶ Pensando como o computador faria:
 1. aguarda digitação de um número;
 2. “trata” número digitado;
 3. aguarda digitação do próximo número;
- ▶ Mapear a lógica acima para os comandos conhecidos.
- ▶ Quais, em que ordem (quebra-cabeça).
- ▶ O que colocar em “trata” número digitado?

Lógica de programação

- ▶ Como construir a lógica (algoritmo) que pode ser formalizada em um programa?
- ▶ Pensando como o computador faria:
 1. aguarda digitação de um número;
 2. “trata” número digitado;
 3. aguarda digitação do próximo número;
- ▶ Mapear a lógica acima para os comandos conhecidos.
- ▶ Quais, em que ordem (quebra-cabeça).
- ▶ O que colocar em “trata” número digitado?

Lógica de programação

- ▶ Como construir a lógica (algoritmo) que pode ser formalizada em um programa?
- ▶ Pensando como o computador faria:
 1. aguarda digitação de um número;
 2. “trata” número digitado;
 3. aguarda digitação do próximo número;
- ▶ Mapear a lógica acima para os comandos conhecidos.
- ▶ Quais, em que ordem (quebra-cabeça).
- ▶ O que colocar em “trata” número digitado?

Lógica de programação

- ▶ O computador processa um único número por vez
- ▶ A resposta em si não é fundamental
- ▶ A questão é entender o processo de solução
- ▶ Depois implementá-la
- ▶ Vejamos como o computador recebe os dados

Ler números até aparecer um zero

5

Ler números até aparecer um zero

8

Ler números até aparecer um zero

3

Ler números até aparecer um zero

9

Ler números até aparecer um zero

0

Lógica de programação

- ▶ Quantos números apareceram?
- ▶ A resposta não importa!
- ▶ Importa é o processo!!!

Lógica de programação

► “trata” número digitado:

1. se digitou zero, imprime a quantidade de números digitados;
2. como contar? Utiliza uma variável (iniciar, somar, imprimir);
3. Solução:

```
program contar_numeros_v1;  
var cont, n: integer;  
  
begin  
  cont:= 1;  
  read (n) ;  
  while n <> 0 do  
  begin  
    read (n) ;  
    cont:= cont + 1;  
  end;  
  writeln (cont - 1) ;  
end.
```

Lógica de programação

- ▶ “trata” número digitado:
 1. se digitou zero, imprime a quantidade de números digitados;
 2. como contar? Utiliza uma variável (iniciar, somar, imprimir);
 3. Solução:

```
program contar_numeros_v1;  
var cont, n: integer;  
  
begin  
  cont:= 1;  
  read (n) ;  
  while n <> 0 do  
  begin  
    read (n) ;  
    cont:= cont + 1;  
  end;  
  writeln (cont - 1) ;  
end.
```

Lógica de programação

- ▶ “trata” número digitado:
 1. se digitou zero, imprime a quantidade de números digitados;
 2. como contar? Utiliza uma variável (iniciar, somar, imprimir);
 3. Solução:

```
program contar_numeros_v1;  
var cont, n: integer;  
  
begin  
  cont:= 1;  
  read (n) ;  
  while n <> 0 do  
  begin  
    read (n) ;  
    cont:= cont + 1;  
  end;  
  writeln (cont - 1) ;  
end.
```

Lógica de programação

- ▶ “trata” número digitado:
 1. se digitou zero, imprime a quantidade de números digitados;
 2. como contar? Utiliza uma variável (iniciar, somar, imprimir);
 3. Solução:

```
program contar_numeros_v1;  
var cont, n: integer;  
  
begin  
  cont:= 1;  
  read (n) ;  
  while n <> 0 do  
  begin  
    read (n) ;  
    cont:= cont + 1;  
  end;  
  writeln (cont - 1) ;  
end.
```

Lógica de programação

- ▶ “trata” número digitado:
 1. se digitou zero, imprime a quantidade de números digitados;
 2. como contar? Utiliza uma variável (iniciar, somar, imprimir);
 3. Solução:

```
program contar_numeros_v1;  
var cont, n: integer;  
  
begin  
    cont:= 1;  
    read (n) ;  
    while n  $\diamond$  0 do  
        begin  
            read (n) ;  
            cont:= cont + 1;  
        end;  
    writeln (cont - 1) ;  
end.
```

Lógica de programação

► Versão alternativa:

```
program contar_numeros_v2;  
var cont, n: integer;  
  
begin  
    cont:=0;  
    read (n) ;  
    while n > 0 do  
        begin  
            cont:= cont + 1;  
            read (n) ;  
        end;  
    writeln (cont) ;  
end.
```


Fim

- ▶ este material está no livro no capítulo 6, seção 6.1

Licença

- ▶ Slides feitos em \LaTeX usando beamer
- ▶ Licença

Creative Commons Atribuição-Uso Não-Comercial-Vedada a Criação de Obras Derivadas 2.5 Brasil License.<http://creativecommons.org/licenses/by-nc-nd/2.5/br/>

Creative Commons Atribuição-Uso Não-Comercial-Vedada a Criação de Obras Derivadas 2.5 Brasil License.<http://creativecommons.org/licenses/by-nc-nd/2.5/br/>