

Curso de C

Introdução

Adriano Joaquim de Oliveira Cruz

Universidade Federal do Rio de Janeiro

27 de outubro de 2015

Section Summary

- 1 Introdução
- 2 Primeiros Passos: Repetições e Testes
- 3 Dados
- 4 Vetores

- 1 Introdução
- 2 Algoritmos
- 3 Tipos de Constantes e Variáveis
- 4 Entrada e Saída pelo Console
- 5 Operadores e Expressões
- 6 Comandos de Controle
- 7 Vetores e Cadeias de Caracteres
- 8 Funções
- 9 Ponteiros
- 10 Estruturas
- 11 Entrada e Saída por Arquivos

- 1 Projetada para criar programas rápidos.
- 2 É das mais baixo nível das alto níveis.
- 3 Gera códigos mais perto do que os computadores entendem.

Como fazer?

alo.c

```
#include<stdio.h>

int main ()
{
    printf("Alo mundo!\n");
    return 0;
}
```

Programa fonte

No terminal

```
> gcc -o alo alo.c
>
```

Compilar

No terminal

```
> ./alo
Alo mundo!
>
```

Executar

alo

```
01001
00100
00011
11001
11111
```

Executável

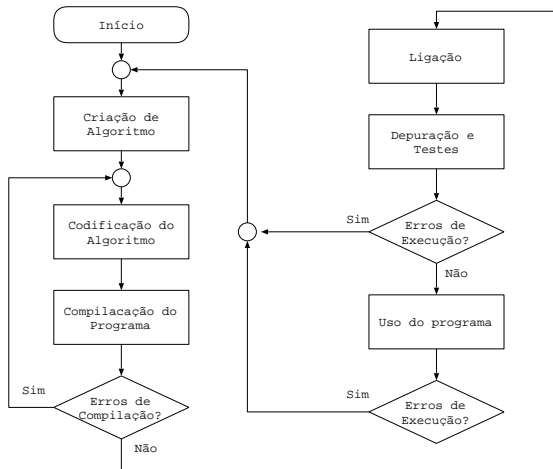
- 1 Algoritmos são sequências de instruções que definem **precisamente** o que deve ser executado.

- 1 Algoritmos são sequências de instruções que definem **precisamente** o que deve ser executado.

Algoritmos devem obedecer aos seguintes limites:

- 1 sempre dar alguma resposta;
- 2 sempre dar a resposta correta e nunca uma resposta incorreta;
- 3 terminar em um número finito de passos;
- 4 trabalhar em todos os exemplos da classe de problemas que o algoritmo se propõe a resolver.

Ciclo de Desenvolvimento



Um programa completo

```
/*
 * Programa: ano.c
 * Descrição: Imprime que estamos no ano 2015,
 *             o ano em que fomos ao futuro.
 * Autor: Adriano Cruz
 * (c) 2015
 */
#include <stdio.h>
int main (void) {
    int ano = 2015;

    /* Imprime o valor do ano */
    printf("Estamos no ano %d", ano);
    return 0;
}
```

O que faz?

```
float media;  
  
scanf("%f", &media);  
if (media >= 7.0) {  
    printf("Aprovado\n");  
}
```

E este?

```
int i = 10;

while (i > 0) {
    puts("Se aprende a programar programando.");
    i = i - 1;
}
```

E este?

```
char nome[20];  
puts("Entre com o seu nome.");  
scanf("%19s", nome);  
printf("Caro %s, bom curso de C.\n");
```

Section Summary

- 1 Introdução
- 2 Primeiros Passos: Repetições e Testes**
- 3 Dados
- 4 Vetores

inicio

Levantar e abaixar braço direito

Levantar e abaixar braço esquerdo

Levantar e abaixar perna esquerda

Levantar e abaixar perna direita

fin

- As instruções do algoritmo são executadas passo a passo.
- A não ser que seja indicado, somente uma instrução por vez é executada.
- Se for necessário podemos ter:
 - Repetições:
 - Repita 10 vezes este conjunto de instruções.
 - Decisões:
 - Se estiver doente não faça os exercícios.

Linguagem Natural: Os algoritmos são expressos diretamente em linguagem natural, como nos exemplos anteriores.

Fluxograma Convencional: Esta é uma representação gráfica que emprega formas geométricas padronizadas para indicar as diversas ações e decisões que devem ser executadas para resolver o problema.

Pseudo-linguagem: Emprega uma linguagem intermediária entre a linguagem natural e uma linguagem de programação para descrever os algoritmos.

Um algoritmo com repetições

inicio

repita

 Levantar e abaixar braço direito

 Levantar e abaixar braço esquerdo

 Levantar e abaixar perna esquerda

 Levantar e abaixar perna direita

até *completar 10 vezes*

fin

Um algoritmo com repetições e testes

início

se *está doente* então

| Vá para casa

senão

| repita

| | Levantar e abaixar braço direito

| | Levantar e abaixar braço esquerdo

| | Levantar e abaixar perna esquerda

| | Levantar e abaixar perna direita

| até *completar 10 vezes*

fim se

fin

Um algoritmo mais nerd

inicio

se *está doente* **então**

 Vá para casa

senão

 vezes $\leftarrow 0$

repita

 Levantar e abaixar braço direito

 Levantar e abaixar braço esquerdo

 Levantar e abaixar perna esquerda

 Levantar e abaixar perna direita

 vezes \leftarrow vezes + 1

até vezes > 10

fim se

fin

Section Summary

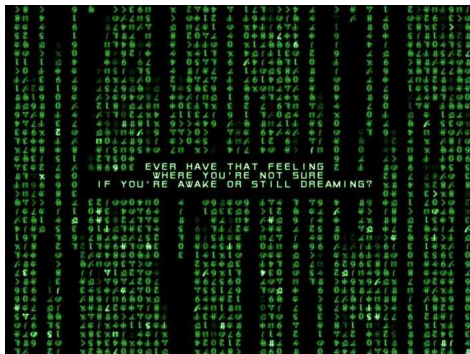
- 1 Introdução
- 2 Primeiros Passos: Repetições e Testes
- 3 Dados**
- 4 Vetores

Dados?

- Algoritmos podem precisar de dados!
- Quem fornecerá os dados?
- Algoritmos podem precisar fornecer resultados!
- Quem receberá os resultados?

Dados?

- Algoritmos podem precisar de dados!
- Quem fornecerá os dados?
- Algoritmos podem precisar fornecer resultados!
- Quem receberá os resultados?



Equação primeiro grau

- Considere a equação:
- $ax + b = 0$
- Solução:
- $x = \frac{-b}{a}$

Entrada: Coeficientes a e b da equação $ax + b = 0$

Saída: Resultado x da Equação

inicio

-- Lendo dados de algum lugar

ler a

ler b

$x \leftarrow -b/a$

-- Fornecendo resultados

imprimir "A raiz da equação vale ", x

fin

Solução em forma de algoritmo

Entrada: Coeficientes a e b da equação $ax + b = 0$

Saída: Resultado x da Equação

inicio

-- Lendo dados de algum lugar

ler a

ler b

se $a = 0$ então

-- Fornecendo resultados

imprimir “A equação nao tem solução”

senão

$x \leftarrow -b/a$

-- Fornecendo resultados

imprimir “A raiz da equação vale ”, x

fim se

fin

Média?

- Considere o problema de calcular a média de um aluno que fez três provas.
- Como fazer isto?



Solução em forma de algoritmo

Entrada: Três notas de um aluno, (*notaAluno1*, *notaAluno2*, *notaAluno3*).

Saída: Média das notas do aluno, (*mediaAluno*)

inicio

- Lê notas. Considere usuário inteligente.

ler *notaAluno1*, *notaAlunos2*, *notaAluno3*

$mediaAluno \leftarrow (notaAluno1 + notaAluno2 + notaAluno3)/3$

- Imprime média

imprimir "A média é ", *mediaAluno*

fin

Entrada: Três notas de um aluno, (*notaAluno*).

Saída: Média das notas do aluno, (*mediaAluno*)

inicio

ler *notaAluno*

mediaAluno \leftarrow *notaAluno*

ler *notaAluno*

mediaAluno \leftarrow *mediaAluno* + *notaAluno*

ler *notaAluno*

mediaAluno \leftarrow *mediaAluno* + *notaAluno*

mediaAluno \leftarrow *mediaAluno* / 3

imprimir "A média é ", *mediaAluno*

fin

Aprovado?

- Considere o problema de calcular a média de um aluno que fez três provas e também informar se o aluno foi aprovado.
- Como fazer isto?



Solução em forma de algoritmo

início

```
ler notaAluno
mediaAluno  $\leftarrow$  notaAluno
ler notaAluno
mediaAluno  $\leftarrow$  mediaAluno + notaAluno
ler notaAluno
mediaAluno  $\leftarrow$  mediaAluno + notaAluno
mediaAluno  $\leftarrow$  mediaAluno/3
imprimir "A média é ", mediaAluno
se mediaAluno  $\geq$  5.0 então
    | imprimir "Aprovado"
senão
    | imprimir "Reprovado "
fim se
```

fin

- O algoritmo repete três vezes os mesmos comandos.
- Vamos usar repetição.
- O algoritmo somente serve para três notas.
- Vamos pedir ao usuário quantas provas o aluno fez.

inicio

mediaAluno \leftarrow 0

imprimir “Quantas provas?”

ler *quantProvas*

notasLidas \leftarrow 0

enquanto *notasLidas* < *quantProvas* **faça**

ler *notaAluno*

mediaAluno \leftarrow *mediaAluno* + *notaAluno*

notasLidas \leftarrow *notasLidas* + 1

fim enqto

mediaAluno \leftarrow *mediaAluno* / *quantProvas*

imprimir “A média é ”, *mediaAluno*

se *mediaAluno* \geq 5.0 **então**

imprimir “Aprovado”

senão

imprimir “Reprovado ”

fim se

fin

Descobrir a maior nota?

- Considere o problema de descobrir a maior nota de um aluno que fez três provas.
- Como fazer isto?



Solução em forma de algoritmo

Entrada: Três notas de um aluno, (*notaAluno*).

Saída: Maior das notas do aluno, (*maiorNota*)

inicio

ler *notaAluno*

maiorNota \leftarrow *notaAluno*

ler *notaAluno*

se *notaAluno* > *maiorNota* então

 | *maiorNota* \leftarrow *notaAluno*

fim se

ler *notaAluno*

se *notaAluno* > *maiorNota* então

 | *maiorNota* \leftarrow *notaAluno*

fim se

imprimir “A maior nota das notas é”, *maiorNota*

fin

Descobrir a maior nota?

- Considere o problema de descobrir a maior nota de um aluno que fez 10 provas.
- Como fazer isto?



Solução em forma de algoritmo

Entrada: Muitas notas de um aluno, (*notaAluno*).

Saída: Maior das notas do aluno, (*maiorNota*)

inicio

ler *notaAluno*

maiorNota \leftarrow *notaAluno*

notas \leftarrow 1

enquanto *notas* < 10 **faça**

ler *notaAluno*

se *notaAluno* > *maiorNota* **então**

maiorNota \leftarrow *notaAluno*

fim se

notas \leftarrow *notas* + 1

fim enqto

imprimir "A maior nota das notas é ", *maiorNota*

fin

Quantos aprovados?

- Considere o problema de descobrir quantos alunos foram aprovados em uma turma de 10 alunos.
- O programa deverá ler as 10 notas finais.
- Como fazer isto?



Solução em forma de algoritmo

Entrada: Notas de 10 alunos, (*notaAluno*).

Saída: Quantidade de aprovados, (*aprovados*)

inicio

aprovados \leftarrow 0

notasLidas \leftarrow 0

enquanto *notasLidas* < 10 **faça**

ler *notaAluno*

se *notaAluno* \geq 5.0 **então**

aprovados \leftarrow *aprovados* + 1

fim se

notasLidas \leftarrow *notasLidas* + 1

fim enqto

imprimir *aprovados*, “alunos foram aprovados”

fin

Section Summary

- 1 Introdução
- 2 Primeiros Passos: Repetições e Testes
- 3 Dados
- 4 Vetores**

O que são?

- Vetores são usados para tratamento de conjuntos de dados que possuem as mesmas características.
- Por exemplo, podemos precisar ter a disposição as notas de todos os alunos.
- Uma das vantagens de vetores é que o conjunto recebe um nome comum.
- Os elementos do conjunto são referenciados através de índices.

- Em Matemática a notação usualmente empregada é, por exemplo:

$$M = \frac{\sum_{i=0}^{N-1} n_i}{N}$$

- Elemento em Matemática: n_3
- Em Computação usamos colchetes:
- $n[3]$ ou $n[i]$

Quantos acima da média?

- Considere o problema de descobrir quantos alunos tiveram notas acima de média da sua turma.
- A quantidade de alunos da turma é 20.
- As notas dos alunos devem ser lidas.
- Como fazer isto?



Solução em forma de algoritmo

início

mediaTurma \leftarrow 0

notasLidas \leftarrow 0

acimaMedia \leftarrow 0

enquanto *notasLidas* < 20 **faça**

ler *notaAluno*[*notasLidas*]

mediaTurma = *mediaTurma* + *notaAluno*[*notasLidas*]

notasLidas = *notasLidas* + 1

fim enqto

mediaTurma = *mediaTurma*/20

notasLidas \leftarrow 0

enquanto *notasLidas* < 20 **faça**

se *notaAluno*[*notasLidas*] > *mediaTurma* **então**

acimaMedia \leftarrow *acimaMedia* + 1

fim se

notasLidas \leftarrow *notasLidas* + 1

fim enqto

imprimir *acimaMedia*, “alunos tiraram notas acima média.”

fin

The End