



Universidade de Brasília
Faculdade UnB Gama
Disciplina: Estrutura de Dados e Algoritmos - EDA

Revisão

Linguagem C

Prof. Nilton Correia da Silva

5 de setembro de 2023

Fundamentos Básicos de C

Introdução à Linguagem C

Variáveis e Tipos de Dados

Entrada(**scanf()**) e Saída(**printf()**)

Estruturas Condicionais

Estrutura de Decisão **if**

Estrutura **switch case**

Estrutura **switch case**

Estruturas de Repetição

Estrutura de Repetição **for**

Estrutura de Repetição **while**

Estrutura de Repetição **do while**

Exemplos Práticos

Fundamentos Básicos de C

Introdução à Linguagem C



- ▶ Estrutura básica de um programa em C.
- ▶ Compilação e execução.

Tipos de Dados Básicos

Tipo	Tamanho (bytes)	Valor Mínimo	Valor Máximo
char	1	-128	127
unsigned char	1	0	255
int	4	-2147483648	2147483647
unsigned int	4	0	4294967295
float	4	Depende da implementação	
double	8	Depende da implementação	

Tabela: Tamanhos e intervalos de tipos básicos da linguagem C.

Fundamentos Básicos de C

Variáveis e Tipos de Dados



```
1  #include <stdio.h>
2  int main() {
3      //Exemplos de declaração e inicialização dos tipos básicos
4      char caractere = 'A';
5      unsigned char uchar = 200;
6      int inteiro = 42;
7      unsigned int uinteiro = 1000;
8      float pontoFlutuante = 3.14;
9      double duplo = 12345.6789;
10     //Exemplos de impressão dos valores
11     printf("Char: %c\n", caractere);
12     printf("Unsigned Char: %u\n", uchar);
13     printf("Int: %d\n", inteiro);
14     printf("Unsigned Int: %u\n", uinteiro);
15     printf("Float: %f\n", pontoFlutuante);
16     printf("Double: %lf\n", duplo);
17     return 0;
18 }
```

Fundamentos Básicos de C

Variáveis e Tipos de Dados



```
1  #include <stdio.h>
2  int main() {
3      int a = 10;
4      int b = 5;
5      float c = 15.5;
6      float d = 2.5;
7      printf("Operações com inteiros:\n");
8      printf("%d + %d = %d\n", a, b, a + b);
9      printf("%d - %d = %d\n", a, b, a - b);
10     printf("%d * %d = %d\n", a, b, a * b);
11     printf("%d / %d = %d\n", a, b, a / b);
12     printf("\nOperações com floats:\n");
13     printf("%.2f + %.2f = %.2f\n", c, d, c + d);
14     printf("%.2f - %.2f = %.2f\n", c, d, c - d);
15     printf("%.2f * %.2f = %.2f\n", c, d, c * d);
16     printf("%.2f / %.2f = %.2f\n", c, d, c / d);
17     return 0;
18 }
```

Fundamentos Básicos de C

Entrada(`scanf()`) e Saída(`printf()`)



```
1  #include <stdio.h>
2  int main() {
3      char caractere;
4      int inteiro;
5      float pFlutuante;
6      // Exemplo de uso do scanf com char
7      scanf("%c", &caractere);
8      printf("Caractere digitado: %c\n", caractere);
9      // Limpa o buffer do teclado
10     while (getchar() != '\n');
11     // Exemplo de uso do scanf com int
12     scanf("%d", &inteiro);
13     printf("Número inteiro digitado: %d\n", inteiro);
14     // Exemplo de uso do scanf com float
15     scanf("%f", &pFlutuante);
16     printf("Ponto flutuante digitado: %.2f\n", pFlutuante);
17     return 0;
18 }
```

Estruturas Condicionais

Estrutura de Decisão if - A



```
1  #include <stdio.h>
2  int main() {
3      int idade;
4      //Exemplo de uso do if simples
5      printf("Digite sua idade: ");
6      scanf("%d", &idade);
7      if (idade >= 18) {
8          printf("Você é maior de idade.\n");
9      }
10     return 0;
11 }
```

os abre e fecha chaves dos if/else são dispensáveis.

Estruturas Condicionais

Estrutura de Decisão if - B



```
1  #include <stdio.h>
2  int main() {
3      int idade;
4      // Exemplo de uso do if-else
5      if (idade >= 18)
6          printf("Você é maior de idade.\n");
7      else
8          printf("Você é menor de idade.\n");
9      return 0;
10 }
```

Estruturas Condicionais

Estrutura de Decisão if - C



```
1  #include <stdio.h>
2  int main() {
3      int idade;
4      // Exemplo de uso do if-else if-else
5      if (idade < 13) {
6          printf("Você é uma criança.\n");
7      } else if (idade < 18) {
8          printf("Você é um adolescente.\n");
9      } else {
10         printf("Você é um adulto.\n");
11     }
12     return 0;
13 }
```

os abre e fecha chaves dos if/else são dispensáveis.

Estruturas Condicionais

Estrutura switch case - A



```
1  #include <stdio.h>
2  int main() {
3      int opcao;
4      printf("Escolha uma opção (1 a 3): ");
5      scanf("%d", &opcao);
6      switch (opcao) {
7          case 1:
8              printf("Você escolheu a opção 1.\n"); break;
9          case 2:
10             printf("Você escolheu a opção 2.\n"); break;
11          case 3:
12             printf("Você escolheu a opção 3.\n"); break;
13          default:
14             printf("Opção inválida.\n"); break;
15      }
16      return 0;
17 }
```

Estruturas Condicionais

Estrutura switch case - B



```
1  #include <stdio.h>
2  int main() {
3      char opcao;
4      printf("Escolha uma opção (a, b ou c): ");
5      scanf(" %c", &opcao);
6      switch (opcao) {
7          case 'a':
8              printf("Você escolheu a opção a.\n"); break;
9          case 'b':
10             printf("Você escolheu a opção b.\n"); break;
11          case 'c':
12             printf("Você escolheu a opção c.\n"); break;
13          default:
14             printf("Opção inválida.\n"); break;
15      }
16      return 0;
17 }
```

A estrutura de repetição `for` é usada para executar um bloco de código repetidamente por um número específico de vezes.

Sintaxe geral:

Sintaxe

```
for (inicialização; condição; atualização) {  
    // Bloco de código a ser repetido  
}
```

- ▶ **Inicialização**: Inicializa uma variável de controle.
- ▶ **Condição**: Avalia a condição a cada iteração. Se for verdadeira, o bloco de código é executado.
- ▶ **Atualização**: Executa após cada iteração. Geralmente é usado para atualizar a variável de controle.

Estruturas de Repetição

Estrutura de Repetição for



```
1  #include <stdio.h>
2  int main() {
3      int i;
4      printf("Contagem ascendente:\n");
5      for (i = 1; i <= 5; i++) { //imprimir números de 1 a 5
6          printf("%d ", i);
7      }
8      printf("Contagem descendente:\n");
9      for (i = 10; i >= 1; i--) { //imprimir números de 10 a 1
10         printf("%d ", i);
11     }
12     printf("Números pares:\n");
13     for (i = 2; i <= 10; i += 2) { //imprimir números pares de 2
14         printf("%d ", i);
15     }
16     return 0;
17 }
```

A estrutura de repetição **while** é usada para executar um bloco de código repetidamente enquanto uma condição for verdadeira.

Sintaxe geral:

Sintaxe

```
while (condição) {  
    // Bloco de código a ser repetido  
}
```

- ▶ **Condição:** Avalia a condição antes de cada iteração. Se for verdadeira, o bloco de código é executado.

Observação: Certifique-se de que a condição se torne falsa em algum momento para evitar um loop infinito.

Estruturas de Repetição

Estrutura de Repetição `while`



```
1  #include <stdio.h>
2  int main() {
3      int i = 1;
4      printf("Contagem ascendente:\n");
5      while (i <= 5) printf("%d ", i++);
6      printf("\n");
7
8      i = 10;
9      printf("Contagem descendente:\n");
10     while (i >= 1) printf("%d ", i--);
11     printf("\n");
12
13     i = 2;
14     printf("Números pares:\n");
15     while (i <= 10) printf("%d ", i += 2);
16     return 0;
17 }
```


A estrutura de repetição `do while` é usada para executar um bloco de código repetidamente enquanto uma condição for verdadeira. A diferença em relação ao `while` é que o bloco de código é executado pelo menos uma vez, pois a condição é verificada após a execução. Sintaxe geral:

Sintaxe

```
do {  
    // Bloco de código a ser repetido  
} while (condição);
```

- **Condição:** Avalia a condição após cada iteração. Se for verdadeira, o bloco de código é executado novamente.

Observação: Certifique-se de que a condição se torne falsa em algum momento para evitar um loop infinito.

Estruturas de Repetição

Estrutura de Repetição `do while`



```
1  #include <stdio.h>
2  int main() {
3      int i = 1;
4      printf("Contagem ascendente:\n");
5      do
6          printf("%d ", i++);
7      while (i <= 5);
8      i = 10; printf("Contagem descendente:\n");
9      do
10         printf("%d ", i--);
11     while (i >= 1);
12     i = 2; printf("Números pares:\n");
13     do
14         printf("%d ", i += 2);
15     while (i <= 10);
16
17     return 0;
18 }
```

- ▶ Usando as estruturas condicionais e de repetição (implemente um menu), crie um programa de calculadora que lê operações e valores do usuário e mostre o resultado na tela.

- ▶ Aplique estruturas de decisão e laços para verificar se um número fornecido pelo usuário é primo ou não.

- Implemente um menu com três opções: C - Fahrenheit para Celsius, F - Celsius para Fahrenheit e S - Sair. Depois que o usuário optar por uma das duas primeiras opções, leia um valor **float** e mostre seu valor na respectiva escala. Caso o usuário opte pela opção S, encerre o programa.