

7. Estruturas de repetição 1

DIM0321

2015.1

Outline

- 1 Motivação
- 2 Laços while
- 3 Laços do...while

DIM0321

7. Estruturas de repetição 1

2015.1 1 / 23

DIM0321

7. Estruturas de repetição 1

2015.1 2 / 23

- 1 Motivação
- 2 Laços while
- 3 Laços do...while

Exercícios

Assunto

Escreva um programa para imprimir os 100 primeiros números naturais (0 a 99).

```
#include <stdio.h>
```

```
int main(void)
{
    printf("0;");
    printf("1;");
    printf("2;");
    printf("3;");
    printf("4;");
    printf("5;");
    printf("6;");
    printf("7;");
    printf("8;");
    printf("9;");
    printf("10;");
    printf("11;");
    printf("12;");
    printf("13;");
    printf("14;");
    printf("15;");
}
```

DIM0321

7. Estruturas de repetição 1

2015.1 3 / 23

DIM0321

7. Estruturas de repetição 1

2015.1 4 / 23

Exercícios 2

Assunto

Escreva um joguinho que o jogador deve tentar adivinhar um número n entre 1 e 100.

- O número n deve ser primeiramente declarado, e numa linha seguinte ser atribuído o valor $(n \% 100 + 1)$.
- A atribuição indicada acima gera um número aleatório uma vez que uma variável declarada, mas não inicializada, pode receber um valor qualquer.
- O jogador tem 5 tentativas para acertar o número

Comentários

- Esses exercícios contêm tarefas **repetitivas**.
- Muitas linhas de código.
- Nos exemplos, o número de repetição é conhecido **a priori** mas geralmente não é o caso.

Exemplo

Programa que calcula a média dos alunos de uma turma

- Lê o número de alunos na turma
- Lê as notas de cada aluno
- Calcula a média
- Imprime a média

A quantidade de dados de entrada é variável: ela depende do tamanho da turma.

DIM0321

7. Estruturas de repetição 1

2015.1 5 / 23

Solução

```
#include <stdio.h>
int main(void)
{
    int nstudents, i;
    float grade1, grade2, grade3;

    printf("How many students are there ?\n");
    scanf("%d", &nstudents);

    i = 1;

    while (i <= nstudents) {
        printf("Enter 3 grades of students %d: ", i);
        scanf("%f %f %f", &grade1, &grade2, &grade3);
        printf("Average for students %d is %.2f",
            i, (grade1 + grade2 + grade3) / 3);
        }
    return 0;
}
```

DIM0321

7. Estruturas de repetição 1

2015.1 7 / 23

Estruturas de repetição

Objetivo

- Estruturas de repetição permitem executar um mesmo bloco de código várias vezes
- O número de repetição pode ser determinado ou não (i.e. depender de uma **condição de parada**)
- A condição de parada pode ser testada no início (**while**) ou no fim (**do...while**).

Em C

- **while** = enquanto...faça
- **do...while** faça ... enquanto
- **for** próxima aula

DIM0321

7. Estruturas de repetição 1

2015.1 8 / 23

1 Motivação

2 Laços while

3 Laços do...while

Descrição

Sintaxe

```
while (c) comando; // 1 comando C
```

```
while (c) { // 1 comando agrupador = 1 bloco  
    comando1;  
    comando2;  
}
```

Exemplo (Contador de 1 até 10)

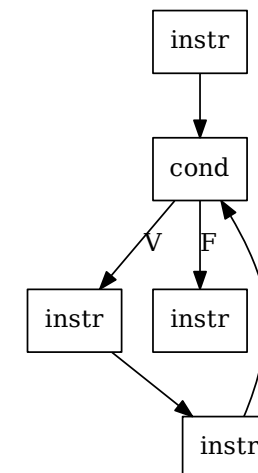
```
int contador = 1;  
while (contador != 10) {  
    printf("%d\n", contador);  
    contador++;  
}
```

- 10 é incluído ou excluído ?

Semântica

- 1 Avaliar <cond>
- 2 Se a avaliação de <cond> != 0 (i.e. condição é verdadeira):
 - 1 executar o corpo.
 - 2 ir à 1
- 3 Se <cond> == 0, terminar a execução, seguir com o primeiro comando fora do while.

Fluxograma



Exercício

- O que faz o seguinte programa ?

```
#include <stdio.h>

int main(void)
{
    int i = 1;
    int n, s;

    s = 0;
    scanf("%i", &n);
    while(i <= n) {
        s += i;
        i++;
    }
    printf("%d\n", s);
    return 0;
}
```

DIM0321

7. Estruturas de repetição 1

2015.1 13 / 23

Exercício

- O que faz o seguinte programa ?

```
#include <stdio.h>
#include <math.h>

int main(void)
{
    int i = 1;
    int n, lim;

    scanf("%i", &n);
    lim = (int) sqrt(n);
    while (i <= lim) {
        if (n % i == 0) {
            printf ("%i %i\n", i, n / i);
        }
        i++;
    }
    return 0;
}
```

DIM0321

7. Estruturas de repetição 1

2015.1 14 / 23

Exercício

- O que faz o seguinte programa ?

```
#include <stdio.h>

int main(void)
{
    int i, j, n;
    scanf("%i\n", &n);
    i = 0;
    j = 1;
    while (i <= n) {
        printf("%i\n", i);
        j = i + j;
        i = j - i;
    }
    return 0;
}
```

DIM0321

7. Estruturas de repetição 1

2015.1 15 / 23

Solução para o segundo exercício

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int main(void)
{
    int n, guess, ntries;

    /* Init random generator with current time */
    srand(time(NULL));

    n = 1 + rand() % 100;
    ntries = 0;
    while (ntries < 5) {
        ntries++;
        printf("Guess a value n, 1 <= n <= 100 (try %d)\n", ntries);
        scanf("%d", &guess);
        if (guess == n) {
            printf("Well done! Found %d in %d tries\n", guess, ntries);
            break;
        }
    }
    if (guess != n) printf("The value was %d.\n", n);
    return 0;
}
```

- srand e rand são declaradas em stdlib.h

DIM0321

7. Estruturas de repetição 1

2015.1 16 / 23

1 Motivação

2 Laços while

3 Laços do...while

Descrição

Sintaxe

```
do comando; while (c); // 1 comando C

do {
    comando1;
    comando2;
} while (c); // 1 comando agrupador = 1 bloco
```

Exemplo (Contador de 1 até 10)

```
int contador = 1;
do {
    printf("%d\n", contador);
    contador++;
} while (contador != 10);
```

- 10 é incluído ou excluído ?

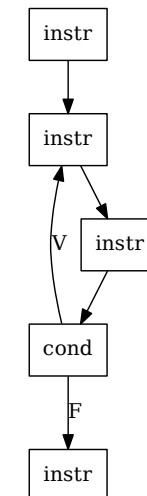
Semântica

1 Executar o corpo

2 Avaliar condição

- ▶ Se for verdadeira (i.e. $\neq 0$) voltar a 1
- ▶ Senão executar o primeiro comando apos o do...while

Fluxograma



Exemplo

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int main(void)
{
    int n, guess, ntries;

    /* Init random generator with current time */
    srand(time(NULL));

    n = 1 + rand() % 100;
    ntries = 0;
    do {
        ntries++;
        printf("Guess a value n, 1 <= n <= 100 (try %d)\n", ntries);
        scanf("%d", &guess);
        if (guess == n) {
            printf("Well done! Found %d in %d tries\n", guess, ntries);
            break;
        }
    } while (ntries < 5);

    if (guess != n) printf("The value was %d.\n", n);
    return 0;
}
```

DIM0321

7. Estruturas de repetição 1

2015.1 21 / 23

Referências

Precisões

[KR88] Seções 3.5 – 3.7

[Bac13] 5.1, 5.2, 5.4

- 📄 André Backes, *Linguagem C completa e descomplicada*, Elsevier, 2013.
- 📄 Brian W. Kernighan and Dennis M. Ritchie, *The (ANSI) C Programming Language*, 2nd ed., Prentice Hall Professional Technical Reference, 1988.

DIM0321

7. Estruturas de repetição 1

2015.1 22 / 23

Perguntas ?



<http://dimap.ufrn.br/~richard/dim0321>

DIM0321

7. Estruturas de repetição 1

2015.1 23 / 23