

9002 — Aula 10

Algoritmos e Programação de Computadores

Instituto de Engenharia – UFMT

Segundo Semestre de 2014

20 de outubro de 2014

Roteiro

- 1 Revisão
- 2 Comandos repetitivos aninhados
- 3 Exercícios

Nas aulas anteriores...

- vimos como escrever programas capazes de executar comandos repetidamente.
- vimos que processo de construção está relacionado com a resposta de três simples perguntas: O que deve ser repetido? Quantas vezes? Qual condição?
- estudamos os comandos **for** e **while**.
- vimos, ainda, como simular e testar estes algoritmos.

Comandos repetitivos aninhados

- É possível incluir um comando de repetição dentro de outro?
- Como simular códigos deste tipo?

Exemplo

Qual é a saída do código a seguir?

```
int i, j;
for(i = 1; i <= 5; i++) {
    printf("Linha %d: ", i);
    for(j = 0; j < 5; j++) {
        printf("%d ", j);
    }
    printf("\n");
}
```

Saída:

```
Linha 1: 0 1 2 3 4
Linha 2: 0 1 2 3 4
Linha 3: 0 1 2 3 4
Linha 4: 0 1 2 3 4
Linha 5: 0 1 2 3 4
```

Exemplo

Qual é a saída do código a seguir?

```
int i, j;
for(i = 1; i <= 5; i++) {
    printf("Linha %d: ", i);
    for(j = 0; j < 5; j++) {
        printf("%d ", j);
    }
    printf("\n");
}
```

Saída:

```
Linha 1: 0 1 2 3 4
Linha 2: 0 1 2 3 4
Linha 3: 0 1 2 3 4
Linha 4: 0 1 2 3 4
Linha 5: 0 1 2 3 4
```

Exercício

Escreva um programa que imprima uma tabela de números ímpares com m linhas e n colunas (m e n são lidos do teclado) como no seguinte formato de exemplo (para $m = 3$ e $n = 4$).

```
01 03 05 07
15 13 11 09
17 19 21 23
```

Dica: você pode utilizar o seguinte formato do `printf` para sempre imprimir o número com dois algarismos (completando com zero antes se necessário): `printf("%02d", ...);`

Problema

Imprimir todas as possibilidades de resultados ao se jogar 4 dados de 6 faces.

- Para cada possibilidade do primeiro dado, devemos imprimir todas as possibilidades dos 3 dados restantes.
- Para cada possibilidade do primeiro e segundo dado, devemos imprimir todas as possibilidades dos 2 dados restantes....
- Você consegue pensar em uma solução com laços aninhados?

Resposta

```
int main(){
    int d1, d2, d3, d4;

    printf("\nD1  D2  D3  D4\n");
    for(d1 = 1; d1 <= 6; d1++)
        for(d2 = 1; d2 <= 6; d2++)
            for(d3 = 1; d3 <= 6; d3++)
                for(d4 = 1; d4 <= 6; d4++)
                    printf("%d  %d  %d  %d\n",d1,d2,d3,d4);

    return 0;
}
```

Exercício

- Na Mega-Sena, um jogo consiste de 6 números distintos com valores entre 1 e 60.

Problema

Imprimir todos os jogos possíveis da Mega-Sena

Exercício

- Partimos da mesma idéia dos dados: Gerar todos os possíveis valores para cada um dos 6 números do jogo.
- Problema: Os números do jogo devem ser distintos.

Exercício

```
int main(){
    int d1, d2, d3, d4, d5, d6;

    for(d1 = 1; d1 <= 60; d1++)
        for(d2 = 1; d2 <= 60; d2++)
            for(d3 = 1; d3 <= 60; d3++)
                for(d4 = 1; d4 <= 60; d4++)
                    for(d5 = 1; d5 <= 60; d5++)
                        for(d6 = 1; d6 <= 60; d6++)
                            if( (d1!=d2) && (d1!=d3) &&.....)
                                printf("%d, %d, %d, %d, %d, %d\n",d1,d2,d3,d4,d5,d6);

    return 0;
}
```

Após incluir todos os testes para garantir que os números são distintos, temos a solução?

Exercício

- Não temos uma solução válida, pois o programa irá imprimir jogos como:
12, 34, 8, 19, 4, 45
34, 12, 8, 19, 4, 45
34, 12, 19, 8, 4, 45
- Na verdade, todos estes jogos são um único jogo: 4, 8, 12, 19, 34, 45.
- Podemos assumir que um jogo é sempre apresentado com os números em ordem crescente.
- Dado que fixamos o valor de **d1**, **d2** necessariamente é maior que **d1**. E com **d1** e **d2** fixados, d3 é maior que d2 etc.

Exercício

Solução correta:

```
int main(){
    int d1, d2, d3, d4, d5, d6;

    for(d1 = 1; d1 <= 60; d1++)
        for(d2 = d1 + 1; d2 <= 60; d2++)
            for(d3 = d2 + 1; d3 <= 60; d3++)
                for(d4 = d3 + 1; d4 <= 60; d4++)
                    for(d5 = d4 + 1; d5 <= 60; d5++)
                        for(d6 = d5 + 1; d6 <= 60; d6++)
                            printf("%d, %d, %d, %d, %d, %d\n", d1, d2, d3, d4, d5, d6);

    return 0;
}
```

Exercício

- Faça um programa que leia um número n e imprima n linhas na tela com o seguinte formato (exemplo se $n = 6$):

1

1 2

1 2 3

1 2 3 4

1 2 3 4 5

1 2 3 4 5 6

Exercício

- Faça um programa que leia um número n e imprima n linhas na tela com o seguinte formato (exemplo se $n = 6$):

```
+ * * * * *  
* + * * * *  
* * + * * *  
* * * + * *  
* * * * + *  
* * * * * +
```


Nas próximas aulas...

- Na próxima aula, resolveremos mais exercícios.
- FIM!