

Programação de computadores II

Turma 65-66

Aula 01 – Revisão de programação estruturada

Elton M. Cardoso

Plano de Aula

- Revisão de programação estruturada
- POO – Conceitos e exemplos
- C++
 - Classes e objetos
 - Construtores e destrutores
 - Encapsulamento
 - Sobrecarga
 - Herança
 - Polimorfismo

Plano de Aula - Continuação

- C++ (Continuação)
 - Classes amigas
 - Templates

Avaliação

- Duas provas – 40% do total do semestre
 - 29-09
 - 08-12
- Trabalhos práticos a serem realizados durante a aula prática

Bibliografia

- STROUSTRUP B., "The C++ Programming Language" Addison-Wesley, 1997.
- DEITEL, H. M. e DEITEL, P. J. "C++: Como Programar." Prentice Hall. 1999.

Programação Estruturada

```
// Arquivo hello.c
#include <stdio.h>

main()
{
    printf("hello, world\n");
}
```

O Programa em C

Importações (includes)

Definições globais

main () {

Definições locais

}

Variáveis e tipos

- Toda variável deve ser declarada antes de usada.
- A toda variável deve ser atribuído um tipo
- Tipos : Int, Float, Char

Regras para nomes de variáveis e constantes

- Sempre deve começar com _ ou uma letra
- Pode conter letras e dígitos
- Sensível a maiúsculas e minúsculas

Obs.: para constantes use todas as letras em maiúsculas.

Operadores

- Aritméticos (+, -, * ,/, inc, dec)
- Relacionais (>, <, ==, !=)
- Booleanos (&&, !, ||)

Controle de Fluxo de execução

- If <condição> then ... eles ...
- While(<condição>){ ... }
- for(valor inicial; <condição>; passo){
- Switch (_{EXP}) {
 case *V1: statements*
 case *v2: statements*
 default: *statements*
}

Funções

- Tipo de retorno NomeDaFunção(parametro_1 ... parametro_n){ }
- Uma função pode não retornar nada **void**
- Um função não obrigatoriamente tem de receber parâmetros

Funções - Passagem de parâmetros

- Por valor (Cópia)
- Por Referência (ponteiro)

Exemplo passagem por valor

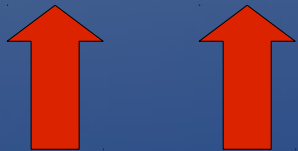
```
int soma(int x, int y){  
    x = x - 1;  
    y = y + 1;  
    return (x + y);  
}
```

```
int main(){  
    int r = Soma(2,3);  
    cout << "2 + 3 = " << r;  
}
```

Exemplo passagem por valor

- No momento da chamada de soma, os valores dos argumentos são copiados para as variáveis declaradas como parâmetros da função.

```
soma(int x, int y){ .... }
```



```
soma(2, 3)
```

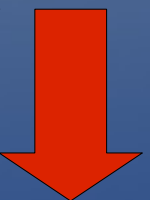
Exemplo passagem por valor

- Quando o comando **return** é usado em um valor, o mesmo é associado ao nome da função e o controle retorna para quem chamou a função.

```
soma(int x, int y){ .... return (x + y) }
```

```
.....
```

```
int r =
```



```
soma(2, 3);
```


Exemplo passagem por referencia

```
int soma(int& x, int& y){  
    x = x - 1;  
    y = y + 1;  
    return (x + y);  
}  
  
int main(){  
    int r = soma(2,3);  
    cout << "2 + 3 = " << r;  
}
```

Exemplo passagem por referência

- Ao invés de realizar uma cópia do valor, uma referência a variável original é utilizada.
- Mudanças nos valores destas variáveis dentro da função afetarão as variáveis originais.