unipampa

Universidade Federal do Pampa

Disciplina de Algoritmos e Programação

unipampa
Universidade Federal do Pampa

Aula Passada

- Teórica
  - Variáveis “string”
  - Comandos de Entrada e Saída Formatada
    - Utilização do comando de entrada (scanf)
    - Aprofundamento do comando de saída (printf)

unipampa
Universidade Federal do Pampa

Aula de Hoje

- Teórica
  - Operadores (atribuição e aritméticos) e Funções auxiliares (strings)
    - Operadores
      - Operador de atribuição (forma geral utilizada)
        - » Atribuições de valores constantes para variáveis.
        - » Atribuições de valores de variáveis para outras variáveis.
        - » Atribuição de resultados de expressões (aritmética, lógica e relacional) para variáveis
      - Operadores aritméticos (+ , - , * , / , %, ++ e --).
        - » Hierarquia das operações aritméticas.
    - Estruturas Condicionais
      - Operadores lógicos ou booleanos (&&, ||, !).
      - Operadores relacionais (> , >=, < , <= , == , !=).
      - Funções matemáticas especiais.

unipampa
Universidade Federal do Pampa

Operador

- Por definição, um operador é um símbolo que obriga o compilador a executar operações matemáticas, comparativas ou lógicas

unipampa
Universidade Federal do Pampa

Operador de atribuição

- Basicamente, é o sinal de igual "="
  - Com ele (o mais usado) podemos fazer ...
    - Atribuições de valores constantes para variáveis
      - Ex.: var1 = 5; // valor da direita é atribuído a variável da esquerda
    - Atribuições de valores de variáveis para outras variáveis
      - Ex.: var1 = var2;
    - Atribuições de resultados de expressões (aritmética, lógica e relacional) para variáveis
      - Ex.: var1 = var2 + 2;
    - Atribuições de retorno de funções para variáveis
      - Ex.: var1 = sqrt(var1); // veremos mais a frente

unipampa
Universidade Federal do Pampa

Operadores Aritméticos

Operador	Ação
-	Subtração
+	Adição
*	Multiplicação
/	Divisão
%	Resto da divisão
--/++	Decremento/incremento

unipampa
Universidade Federal do Pampa

Operadores Aritméticos

- Detalhes
  - Divisão x Resto da divisão
    - Operador % só faz sentido quando usado com valores inteiros (variáveis do tipo int)
      - Ex.: var1 = 5 / 2; // var1 (inteira) = 2;
        var1 = 5 % 2; // var1(inteira) = 1;

unipampa
Universidade Federal do Pampa

Operadores Aritméticos

- Detalhes
  - Decremento/Incremento
    - Operadores -- e ++ são usados para somar/subtrair um de uma variável (tarefa bastante comum)
      - Ex.: var1--; // var1=var1-1;
        var1++; // var1=var1+1;
    - Observação: na prática, estes operadores podem aparecer antes ou depois da variável, o que afeta o resultado da operação
      - Ex.: x=10; y=++x; // y=11 e x=11
        x=10; y=x++; // y=10 e x=11

unipampa
Universidade Federal do Pampa

Operadores Aritméticos

- Precedência (Hierarquia de Operações)

Hierarquia	Operação
1	Parênteses
2	Função
3	++ --
4	* / %
5	+ -

  - Observação: em geral, a execução ocorre da esquerda para a direita; portanto, duas ou mais operações de mesma hierarquia serão avaliadas nesta ordem

Operadores Aritméticos de Atribuição

- Combinam operações aritméticas com a operação de atribuição

+= -= *= /= %=

X op = exp equivale a X = X op (exp)

Exemplo	Equivale a
i += 2;	i = i + 2;
x *= y + 1;	x = x * (y + 1);
t /= 2.5;	t = t / 2.5;
p %= 5;	p = p % 5;
d -=3;	d = d - 3;

unipampa
Universidade Federal do Pampa

Operadores Relacionais

- Usados para comparação

Operador	Ação
>	Maior que
>=	Maior ou igual a
<	Menor que
<=	Menor ou igual
==	Igual a
!=	Diferente de

  - Observação: o resultado de uma expressão que envolva operadores relacionais é 0 se falso ou 1 se verdadeiro
    - Ex.: i=1<2; // i=1 pois 1 é menor que 2
      i=1>2; // i=0 pois 1 não é maior que 2
      - NOTA: Em geral, o valor 0 significa falso e qualquer outro valor é interpretado como verdadeiro

unipampa
Universidade Federal do Pampa

Estruturas Condicionais

- Estruturas de condição
  - Estrutura condicional simples
    - Utilização da estrutura de condição “if” com expressões lógicas simples
    - Utilização do comando “if” com expressões lógicas compostas (&& e ||)
  - Estrutura condicional composta
    - Utilização da cláusula “else” na estrutura “if”
    - Comandos “if” aninhados
  - Estrutura de seleção múltipla
    - Utilização da estrutura de condição “switch”

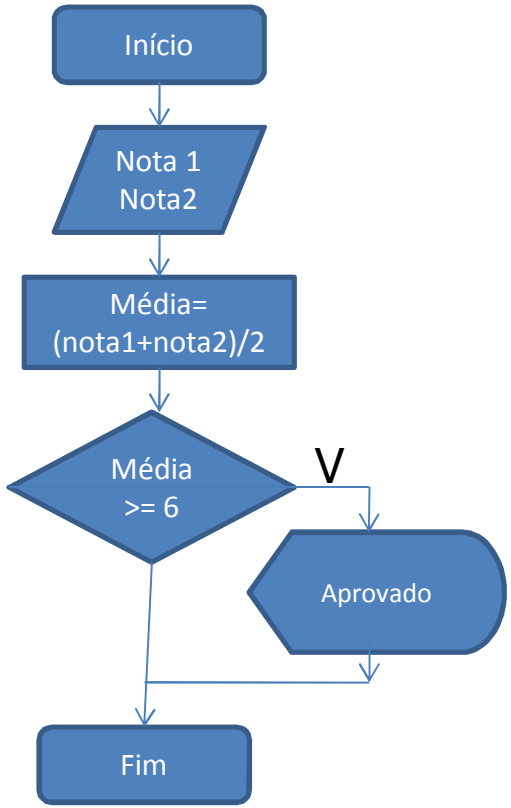
Estrutura condicional simples

- Único comando:

```
if (condição)
        comando;
```

- Múltiplos comandos:

```
if (condição) {
        comando_1;
        comando_2;
        comando_n;
}
```



- OBS.: condição : verdadeiro (!=0) / falso (==0)

unipampa
Universidade Federal do Pampa

Recordando:
Operadores Relacionais

- Usados para comparação

Operador	Ação
>	Maior que
>=	Maior ou igual a
<	Menor que
<=	Menor ou igual
==	Igual a
!=	Diferente de

  - Observação: o resultado de uma expressão que envolva operadores relacionais é 0 se falso ou 1 se verdadeiro
    - Ex.: i=1<2; // i=1 pois 1 é menor que 2
      i=1>2; // i=0 pois 1 não é maior que 2
      - NOTA: Em geral, o valor 0 significa falso e qualquer outro valor é interpretado como verdadeiro

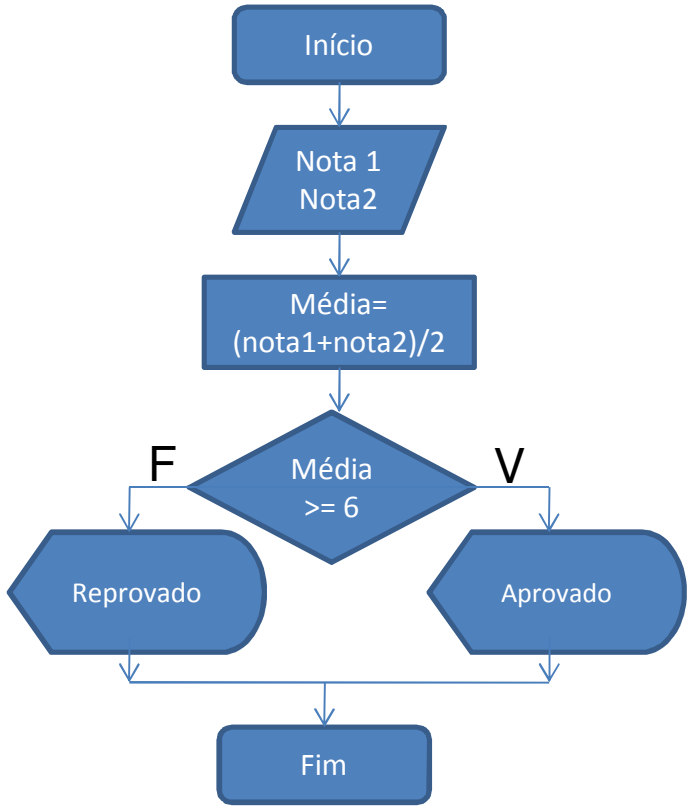
Estrutura condicional composta

- Inclusão do “else” (“senão”):

```
if (condição)
        comando_verdadeiro;
else
        comando_falso;
```

- Múltiplos comandos com “else”:

```
if (condição) {
        comando_verdadeiro_1;
        comando_verdadeiro_n;
} else {
        comando_falso_1;
        comando_falso_n;
}
```



Estrutura condicional composta

- Comandos “if” aninhados

```
if (condição)
        if (condição)
                comando_verdadeiro;
        else
                comando_falso;
else
        if (condição)
                comando_verdadeiro;
        else
                comando_falso;
```

unipampa
Universidade Federal do Pampa

Estrutura de seleção múltipla

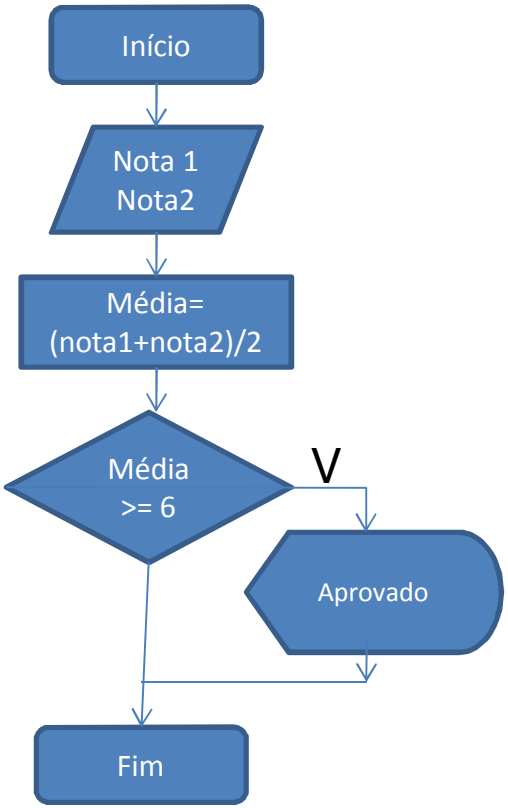
- Comando “switch”

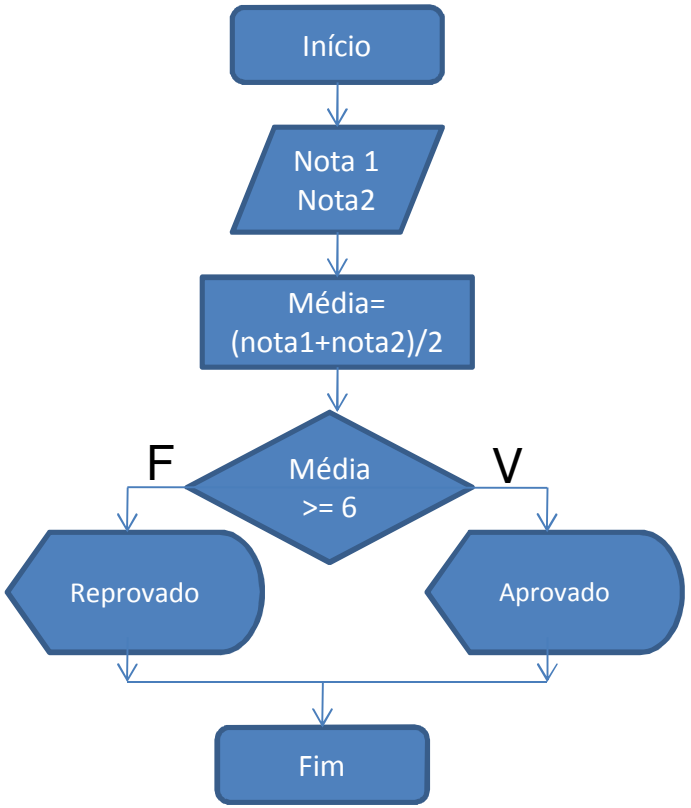
```
switch (variável) { // int ou char
        case valor_1:
                comando_1;
                comando_n;
                break; // interrupção no comando
        case valor_n:
                comando_1;
                comando_n;
                break;
        default:
                comando_1;
                comando_n;
}
```

- Conceito:
  - testa uma variável sucessivamente contra uma lista de constantes inteiras ou caractere; se encontrar uma coincidência, executa os comandos associados à constante; caso contrário, executa os comandos associados à opção default (opcional)

Exercício

- Faça dois programas que correspondam aos fluxogramas apresentados na aula de hoje





unipampa
Universidade Federal do Pampa

Operadores Lógicos

- Usados para análise lógica de expressões simples ou compostas
  - Uma "expressão" na Linguagem C é qualquer combinação válida de operadores, constantes e variáveis

Operador	Ação
&&	AND(e)
||	OR(ou)
!	NOT(não)

  - Ex.: i=a>0 && a<10; // i=1 se a for maior que 0 **E** menor que 10
    i=a<10 || a>20; // i=1 se a for menor que 10 **OU** maior que 20
    i=!a; // i=1 se a for 0;

unipampa
Universidade Federal do Pampa

Operadores Relacionais e Lógicos

- Precedência (Hierarquia de Operações)

Hierarquia	Operação
1	!
2	> >= < <=
3	== !=
4	&&
5	||

  - Observação: em geral, a execução ocorre da esquerda para a direita; portanto, duas ou mais operações de mesma hierarquia serão avaliadas nesta ordem

unipampa
Universidade Federal do Pampa

Observações Complementares

- O operador de atribuição pode ser usado em expressões, junto com operadores matemáticos, lógicos, relacionais, chamadas de funções, e outros
  - Ex.: d = (c = a + b) < 0; // d=1 se c for negativo
    - Ou seja, primeiro é atribuído o valor de a + b para a variável c e só depois o resultado será avaliado
      - 1º passo: c = a + b;
      - 2º passo: d = c < 0;

unipampa
Universidade Federal do Pampa

Observações Complementares

- Conversão de Tipos em Expressões
  - Quando uma expressão é composta de variáveis de tipos diferentes, “C” converte todos os operandos (as variáveis) para o tipo do maior operando: char -> int ; float -> double;
    - Ex.:

```
char ch;
int i;
float f,res;
double d;
res = (float) (ch + i) + (f * d) - (f + i);
              int    double  double
                  double
                        double
```

unipampa
Universidade Federal do Pampa

Observações Complementares

- Operadores matemáticos de atribuição

Operador	Exemplo	Comentário
+=	x += y;	x =x+y
-=	x -= y;	x =x-y
*=	x *= y;	x =x*y
/=	x /= y;	x =x/y
%=	x %= y;	x =x%y

unipampa
Universidade Federal do Pampa

Funções Matemáticas Auxiliares

- Exemplos

Função	Sintaxe	Biblioteca
abs(valor absoluto inteiro)	int abs(int x);	math.h e stdlib.h
sin(seno)	double sin(double x);	math.h
cos(cosseno)	double cos(double x);	math.h
exp(Expoente- e^x)	double exp(double x);	math.h
pow(Portência - x^y)	double pow(double x, double y);	math.h
sqrt(Raiz quadrada)	double sqrt(double x);	math.h
max(Valor máximo)	int max(int a, int b);	stdlib.h
min(Valor mínimo)	int min(int a, int b);	stdlib.h
log(Logaritmo natural)	double log(double x);	math.h

unipampa
Universidade Federal do Pampa

Exercícios

1. Faça um programa que leia dois valores inteiros e que mostre na tela:
   - Resultado da divisão do primeiro pelo segundo número
   - Resto da divisão
   - Resultado da soma destes números
     - Use o operador matemático de atribuição +=
2. Programa “idade_futura.c”.
   - Faça um programa que leia o nome completo (com espaços) e o ano de nascimento de uma pessoa, o ano atual e um ano no futuro, e que calcule e mostre na tela conforme o exemplo:
     - Nome da pessoa completo (com espaços); idade atual da pessoa; idade que a pessoa terá no ano futuro lido.
     - Ex. de saída:

       Fulano de tal tem 30 anos atualmente e fará 40 anos em 2019.

unipampa
Universidade Federal do Pampa

Exercícios

3. Programa “inteira_fracionaria.c”
   - Faça um programa que leia um número real, e que calcule e mostre na tela:
     - a parte inteira do número
     - a parte fracionária do número
     - o arredondamento do número
     - Ex. de saída:

       Parte inteira = 5

       Parte fracionária = 0.5

       Arredondado = 6

unipampa
Universidade Federal do Pampa

Exercícios

4. Programa “horas_minutos.c”
   - Faça um programa que leia um número real (correspondente a uma hora formada por horas – a parte inteira do número - e minutos – a parte fracionária), e que calcule e mostre na tela a hora convertida para minutos.
     - Ex. de saída:

       A hora lida corresponde a 190 minutos.