Introdução à Programação C

Fabio Mascarenhas - 2014.2

http://www.dcc.ufrj.br/~fabiom/introc

Objetivo

- O objetivo desse curso é apresentar os conceitos básicos de programação na linguagem C para alunos que já tiveram alguma experiência com programação em outras linguagens

- Alguns tópicos que veremos ao longo do curso:

  - O ambiente de desenvolvimento C; funções, variáveis e operadores; entrada e saída; estruturas de controle; recursão; vetores; estruturas; ponteiros; alocação dinâmica; arquivos

Avaliação

- Teremos três provas:
  - P1 em 09/10
  - P2 em 02/12
  - P3 em 09/12
- A média final é a média aritmética das duas maiores notas, alunos com nota maior ou igual a 5 estão aprovados

Estrutura de um programa C

```c
#include <stdio.h>

#define TAMANHO 5

static void gera_conjunto(double vetor[], int tamanho, double inicial) {
    int i;
    for (i = 0 ; i < tamanho; i++) {
        vetor[i] = inicial;
        inicial *= 2;
    }
}

int main()
{
    double vetor[TAMANHO], num;
    int i;

    puts("Este programa gera um vetor de numeros inteiros.\n");
    puts("Entre com o numero inicial do conjunto. ");
    scanf("%lf", &num);

    /* Geracao do conjunto */
    gera_conjunto(vetor, TAMANHO, num);

    /* Impressao do conjunto */
    for (i = 0; i < TAMANHO; i++)
        printf("Elemento %d = %lf\n", i, vetor[i]);

    return 0;
}
```

Comandos #include e #define

- Disponibilizam as funções de alguma biblioteca auxiliar
  - stdio.h: entrada e saída padrão
  - math.h: funções matemáticas
  - string.h: manipulação de cadeias caracteres
- Um arquivo .h é um arquivo de *cabeçalho* de funções
- O comando #define define constantes, normalmente números ou strings úteis para o programa

Funções auxiliares

- Veremos como funcionam as funções de C em detalhes em breve

- Elas têm a seguinte estrutura:

```
static tipo_de_retorno nome_da_função(parâmetros) {
  variáveis locais

  comandos
}
```

Função principal

- A função principal sempre é chamada main, e tem a seguinte estrutura:

```
int main() {
    variáveis locais

    comandos
}
```

- É na função principal que faremos a *entrada* e *saída* do programa, chamando as funções auxiliares para processamento da entrada

Variáveis e tipos

- Variáveis em C precisam ser declaradas, e sempre têm um *tipo* associado, que diz quais valores essa variável pode receber

- Os tipos básicos representam números de vários intervalos (correspondendo a diferentes quantidades de memória do computador)

- O tipo char representa números entre -128 e 127, e é muito usado como caractere em cadeias de caractere ASCII ou UTF-8

- O tipo int é o tipo básico para números inteiros, indo de -2.147.483.648 a 2.147.483.647, e corresponde aos inteiros de Python

- O tipo double representa números decimais de ponto flutuante, e corresponde aos números reais de Python

Exemplos de tipos de dados

- Constantes int: 5, 101, 77, -943
- Constantes char: ‘a’, ‘b’, ‘?’ (aspas simples)
- Constantes double: 15.3, -0.37, 1.3e5
- Constantes de cadeias de caractere (vetores de char): “banana”, “ola mundo” (aspas duplas)

Operadores aritméticos

- C também possui os operadores aritméticos +, -, *, / e % (resto da divisão inteira)
- Qualquer uma das quatro operações aritméticas básicas tem resultado inteiro se os operandos forem inteiros, e double se um dos operandos for double
- C também possui os operadores de atribuição +=, -=, *= e /=, que fazem a operação da variável do lado esquerdo com o valor da expressão no lado, e atribuem o resultado à variável
- Variáveis de tipo char e int também têm os operadores ++ e --, que podem aparecer antes ou depois da variável, e respectivamente somam e subtraem 1 da variável