

MC-102 — Aula 01

Introdução à Programação de Computadores

Instituto de Computação – Unicamp

Primeiro Semestre de 2012

Roteiro

- 1 Por que aprender a programar?
- 2 Objetivos da Disciplina

Por que aprender a programar?

Por que aprender a programar?



Por que aprender a programar?

Para se tornar um bilionário!?



Por que aprender a programar?

Para se tornar um hacker!?



Por que aprender a programar?

Para se tornar um nerd!?



Por que aprender a programar?

Eu sou das engenharias! Existem diversos motivos, sendo exemplos:

- Como engenheiro você deverá ser capaz de criar programas para gerenciar e automatizar algum processo que hoje é manual.



1940



2000

Por que aprender a programar?

- Como engenheiro você deverá ser capaz de desenvolver novas ferramentas ou protótipos.
 - Para criar ferramentas/protótipos você deverá fazer simulações computacionais para fazer testes preliminares.
- Você poderá enxergar situações onde uma solução computacional pode trazer benefícios.
 - Mesmo que você não implemente (programe) a solução você poderá propô-la e será capaz de “conversar” com o pessoal de TI para implementar a solução.

Por que aprender a programar?

Eu sou das áreas científicas! Matemática, Física, Química etc.
Existem diversos motivos, sendo exemplos:

- Como cientistas vocês devem propor uma hipótese e testá-la.
 - Em vários casos onde os sistemas são “bem modelados matematicamente”, são criados programas que fazem a simulação do sistema para checagem de uma hipótese.
- Você deverá resolver sistemas de equações complexos que não necessariamente podem ser resolvidos por softwares padrões (como MatLab).
 - Vocês deverão implementar seus próprios resolvedores.

Por que aprender a programar?

- Simulações.
 - Muitos dos modelos propostos para explicar algum fenômeno são simulados computacionalmente. Implementar os modelos é uma tarefa básica.

Objetivos da Disciplina

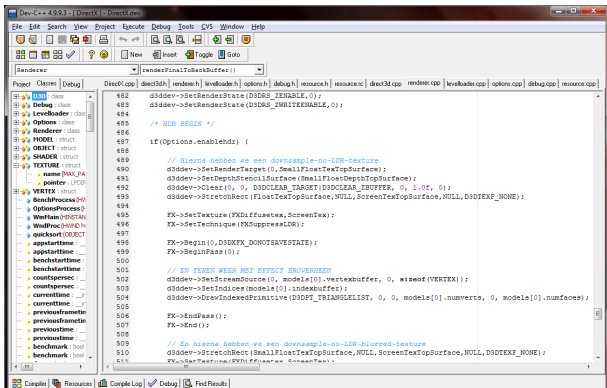
- Neste curso vocês aprenderão o básico para se criar programas para computador.
- Um programa de computador pode ser definido como uma coleção de instruções que descrevem uma tarefa a ser realizada por um computador.
- Tais instruções são escritas na linguagem de programação escolhida e depois são compiladas (ou interpretadas) para serem executadas.
- Exemplos de programas: Firefox , Quake, MatLab, Media Player.

O que será necessário

- Você deverá ter acesso a um computador.
- Para criar um programa, utilizamos um *editor de texto* (para escrever o código do programa) e um *compilador*.
- O compilador transforma o código em um programa executável. Para compilar os programas da linguagem C (os quais serão desenvolvidos no decorrer do curso) utilizaremos o compilador *GCC*.
- Pode ser utilizada uma IDE (como o *DEV C++* e o *CODEBLOCKS*) para editar os programas.
- Se você usa *linux* ou *MAC OS*, você poderá utilizar qualquer editor simples como *emacs*, *kyle* etc.

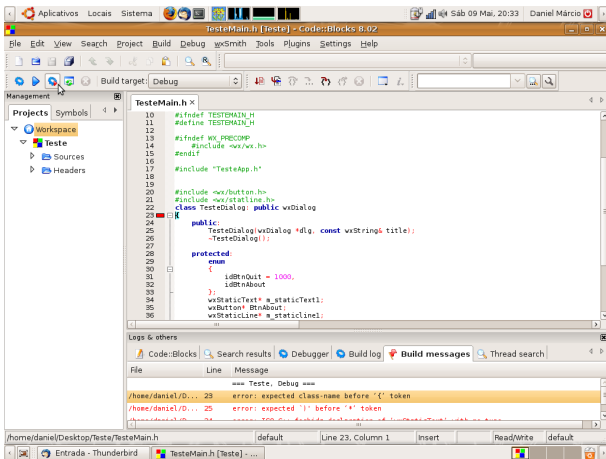
O que será necessário

- Se você usa o Windows ou o LINUX , instale o DEVCC++ baixando do site <http://www.bloodshed.net/dev/devcpp.html>
 - Este programa já tem integrado um editor, um compilador, um depurador, além de outras utilidades.



O que será necessário

- Se você usa o LINUX, instale o CODEBLOCKS e o GCC utilizando o gerenciador de pacotes Synaptic.



O que será necessário

Para ir bem neste curso:

- Faça todos os laboratórios.
- Faça e implemente as listas de exercícios.
- E finalmente faça e implemente as listas de exercícios.

O que será necessário

Para ir bem neste curso:

- Faça todos os laboratórios.
- Faça e implemente as listas de exercícios.
- E finalmente faça e implemente as listas de exercícios.

O que será necessário

Para ir bem neste curso:

- Faça todos os laboratórios.
- Faça e implemente as listas de exercícios.
- E finalmente faça e implemente as listas de exercícios.