

## Apresentação

### MC102 Algoritmos e Programação de Computadores

Aula de Laboratório 01  
Instituto de Computação  
Primeiro Semestre de 2012

6 de março de 2012



#### A aula de Laboratório

- Segunda feira de 10 às 12hs (SI03 e SI05);
- Aplicação da teoria;
- Utilização da linguagem de programação C;
- Frequência  $\geq 75\%$ .

#### Laboratórios (ou trabalhos)

- A cada 2 semanas com prazo de 1 semana para ser entregue;
- Utilizaremos o sistema **Susy** para a correção dos laboratórios;
- Laboratórios são **INDIVIDUAIS**;
- Com a complexidade crescente.



MC102 - Algoritmos e Programação de Computadores

3 / 15

## Conteúdo

- 1 Apresentação
- 2 Programação
- 3 Linguagem C
- 4 Ambiente Linux
- 5 Compilando Programas
- 6 Susy



MC102 - Algoritmos e Programação de Computadores

2 / 15

## Avaliação

#### Nota de Laboratório

- Aplicaremos  $n$  laboratórios (entre 5 a 7), cuja a nota será:

$$\begin{cases} 10.0 & \text{se o programa passar corretamente por todos os testes} \\ 0.0 & \text{caso contrário} \end{cases}$$

- A média será:  $L = \frac{L_1 + L_2 + \dots + L_n}{n}$ .

#### Nota do curso

- Nota Final:
  - Se  $P < 5.0$  ou  $L < 5.0$  então  $M \leftarrow \min\{P, L\}$ ;
  - Se  $P \geq 5.0$  e  $L \geq 5.0$  então  $M \leftarrow \frac{7P+3L}{10}$ ;
- Em caso de Exame:  $M \leftarrow \frac{M+E}{2}$ .



MC102 - Algoritmos e Programação de Computadores

4 / 15

# Lógica

## O que é Lógica?

- É a técnica de encadear pensamentos para atingir determinado objetivo;
- A lógica tem vista a ordem da razão, encadeando a ordem do pensamento.

## Sequência Lógica

- Passos executados até atingir a solução de um problema através de uma sequência de instruções ou passos.



# Lógica de Programação

## Instruções

- Na computação, porém, a instrução é a informação que indica ao computador uma ação a ser executada.

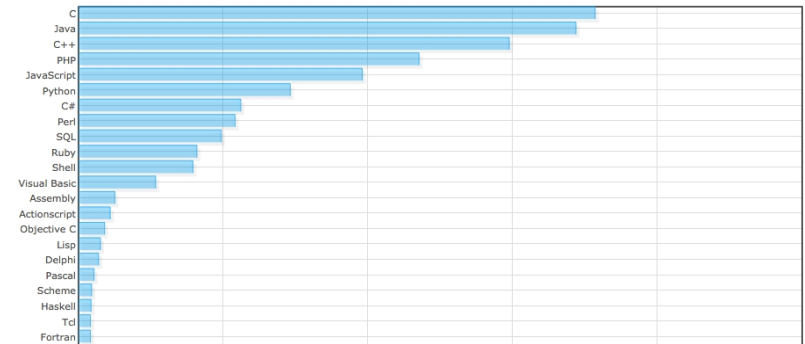
## Lógica de Programação

- Uso correto das leis do pensamento e de processos de raciocínio na programação de computadores;
- Procura representar o raciocínio utilizando Algoritmos (Pseudocódigos).



# História

- Criada na década de 70 por Dennis Ritchie;
- Muitas de suas características derivam da linguagem B (possível explicação do nome);
- Estruturada;
- É uma das linguagens mais populares do mundo;



fonte: <http://www.langpop.com>

# Ambiente

## Acessando os Computadores

- Para se logar nos computadores utilize **login** e **senha**;
- Para abrir um terminal Clique no **Menu: Aplicativos** → **Acessórios** → **Terminal**;
- O Terminal serve para digitar comandos para o sistema operacional;

## Acessando a Internet

- No Terminal digite: `firefox &` .
- Ou clicando no **Menu: Aplicativos** → **Internet** → **Firefox**;



## Ambiente

### Gerência de Diretórios

- **mkdir** <nome-do-diretorio>: cria um diretório.  
Ex.: mkdir programas;
- **rmdir** <nome-do-diretorio>: remove um diretório.  
Ex.: rmdir programas;
- **cd** <nome-do-diretorio>: muda de diretório.  
Ex.: cd programas: vai para a pasta programas;  
cd .. vai para a pasta pai;

## Editores

### Kate

- Terminal: kate &;
- Ou clicando no **Menu: Aplicativos** → **Acessórios** → **Kate**;

### Gedit

- Terminal: gedit &;
- Ou clicando no **Menu: Aplicativos** → **Acessórios** → **Gedit**;

## Ambiente

### Gerência de Arquivos

- **ls**: lista os arquivos e subdiretórios da pasta corrente;
- **mv** <origem> <destino>: move arquivo da orig. para o destino.  
Ex.: mv primeiro.c segundo.c: move o primeiro.c para segundo.c;  
mv primeiro.c programas/: move primeiro.c para a pasta programas.
- **cp** <origem> <destino>: copia arquivo da orig. para o dest.;  
Ex.: cp primeiro.c segundo.c: copia o primeiro.c para segundo.c;  
cp primeiro.c programas/: copia primeiro.c para a pasta programas.
- **rm** <arquivo>: apaga arquivo. Ex.: rm primeiro.c

## Compilação

### No Linux o gcc é o compilador utilizado

- **gcc** <nome do código fonte> -o <nome do executável>;  
Ex.: gcc exemplo.c -o exemplo
- **./<nome do executável>**. Ex.: ./exemplo

# Compilação

## Listing 1: Exemplo de Programa

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int main(){
5     int a;
6     int b;
7     int c;
8     char d;
9     printf("Digite um numero:");
10    scanf("%d",&a);
11    getchar();
12
13    printf("Digite outro numero:");
14    scanf("%d",&b);
15    getchar();
16
17    if(a>b){
18        c = a;
19        a = b;
20        b = c;
21    }
22    printf("Pense em um numero entre %d e %d e pressione enter",a,b);
23    getchar();
24    printf("O numero que voce pensou foi %d\n", a+(rand()%(b-a)));
25 }
```



# Susy

- O Susy (*Submission and Testing System for Student Programs*) é um sistema de submissão e teste;
  - Simplificar a tarefa de recebimento e de testes de programas de alunos;
  - Facilitar as atividades dos alunos através de comunicação pela Internet.
- As tarefas serão publicadas e submetidas pelo Susy;
- O Susy possui sérias restrições de limite de tempo.



# Questões?

Obrigado!

Para informação:

Página do Curso: <http://www.lrc.ic.unicamp.br/~geraldoms/mc102>

E-mail:

[geraldoms\[at\]lrc.ic.unicamp.br](mailto:geraldoms[at]lrc.ic.unicamp.br)

[brhenrique.fischer\[at\]gmail.com](mailto:brhenrique.fischer[at]gmail.com)

