

Unidade I - Introdução ao Linux e à Linguagem C

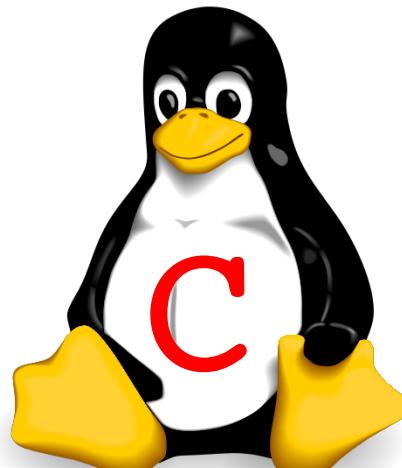
Disciplina Linguagens de Programação I
Bacharelado em Ciência da Computação da Uerj
Professores Guilherme Mota e Leandro Marzulo

ANSI C

```
#include <stdio.h>
int main()
{
    printf("Hello World!\n");
    return 0;
}
```

Que assuntos serão abordados nesta unidade?

- Sistema Linux:
 - arquitetura
 - sistema de arquivos
 - uso desktop
 - shell
 - instalação
- Conceitos básicos da linguagem C:
 - Função `main()`
 - Declaração de variáveis
 - Comando `if`
 - Comando `for`
 - `printf()`/`scanf()`
 - Funções
 - Primeiro programa
 - Compilação Básica
 - Execução do programa
 - Ambiente VPL (Moodle)



Introdução ao Linux

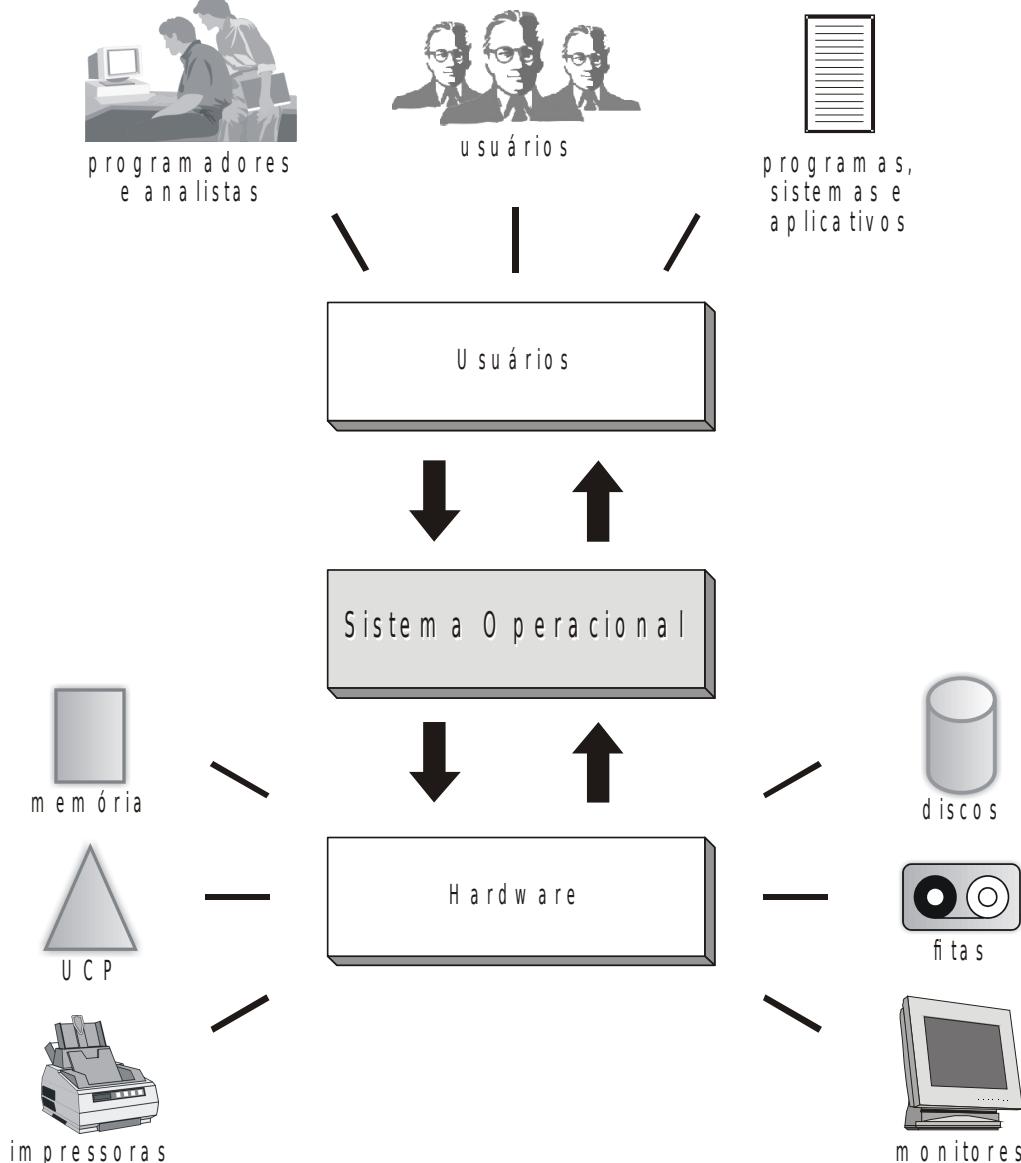
Sugestão de leitura:

Manual Completo do Linux – Guia do Administrador

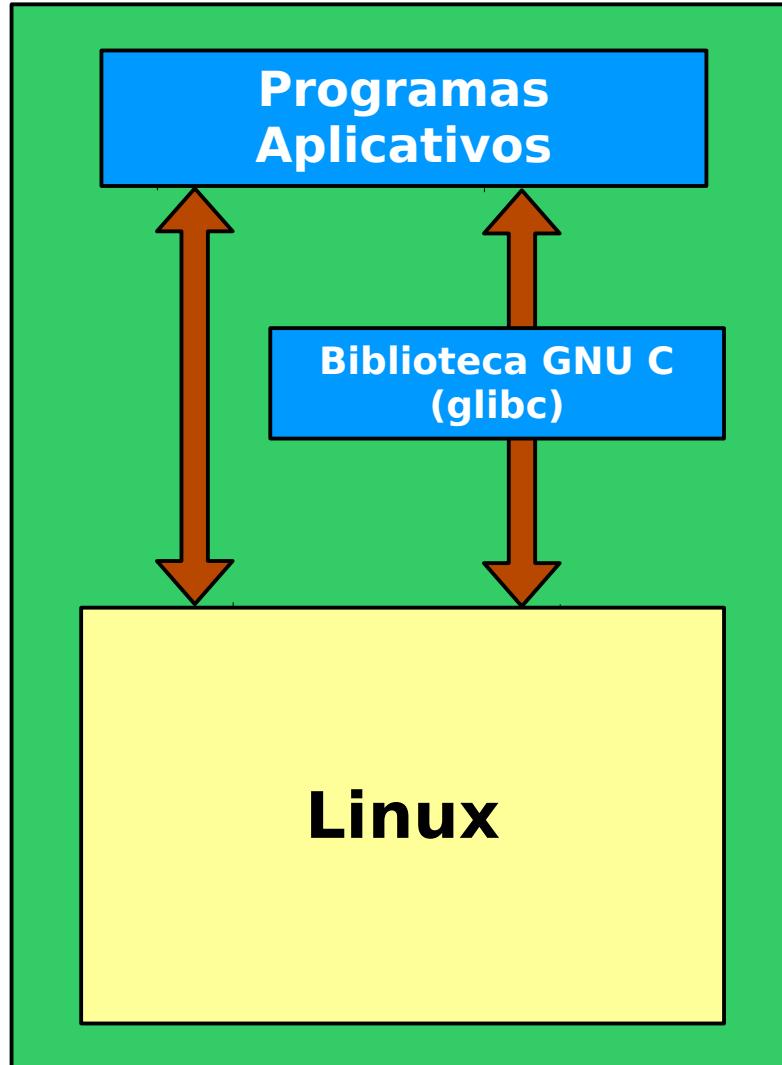
Evi Nemeth/ Garth Snyder/ Trent R. Hein

2^a Edição Pearson / Makron Books

Visão Geral de um Sistema Operacional



Visão Geral da Arquitetura do Sistema



Estrutura de Diretórios

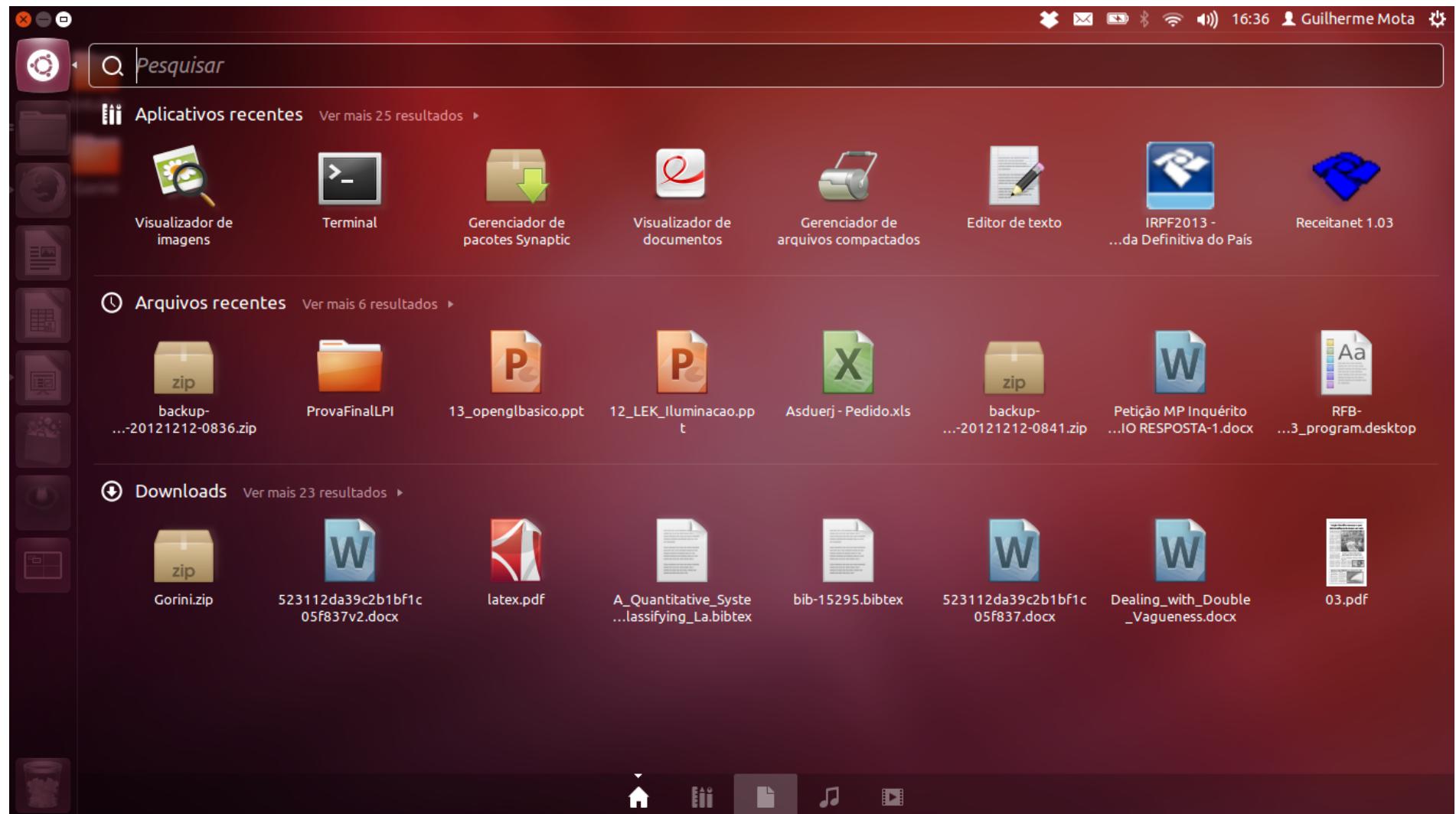
- / diretório raiz do sistema de arquivos
- /bin executáveis de programas básicos
- /etc arquivos de configuração
- /home arquivos pessoais dos usuários
- /lib bibliotecas de uso geral
- /sbin ferramentas administrativas básicas
- /usr arquivos de uso comum

Ubuntu Linux no Desktop

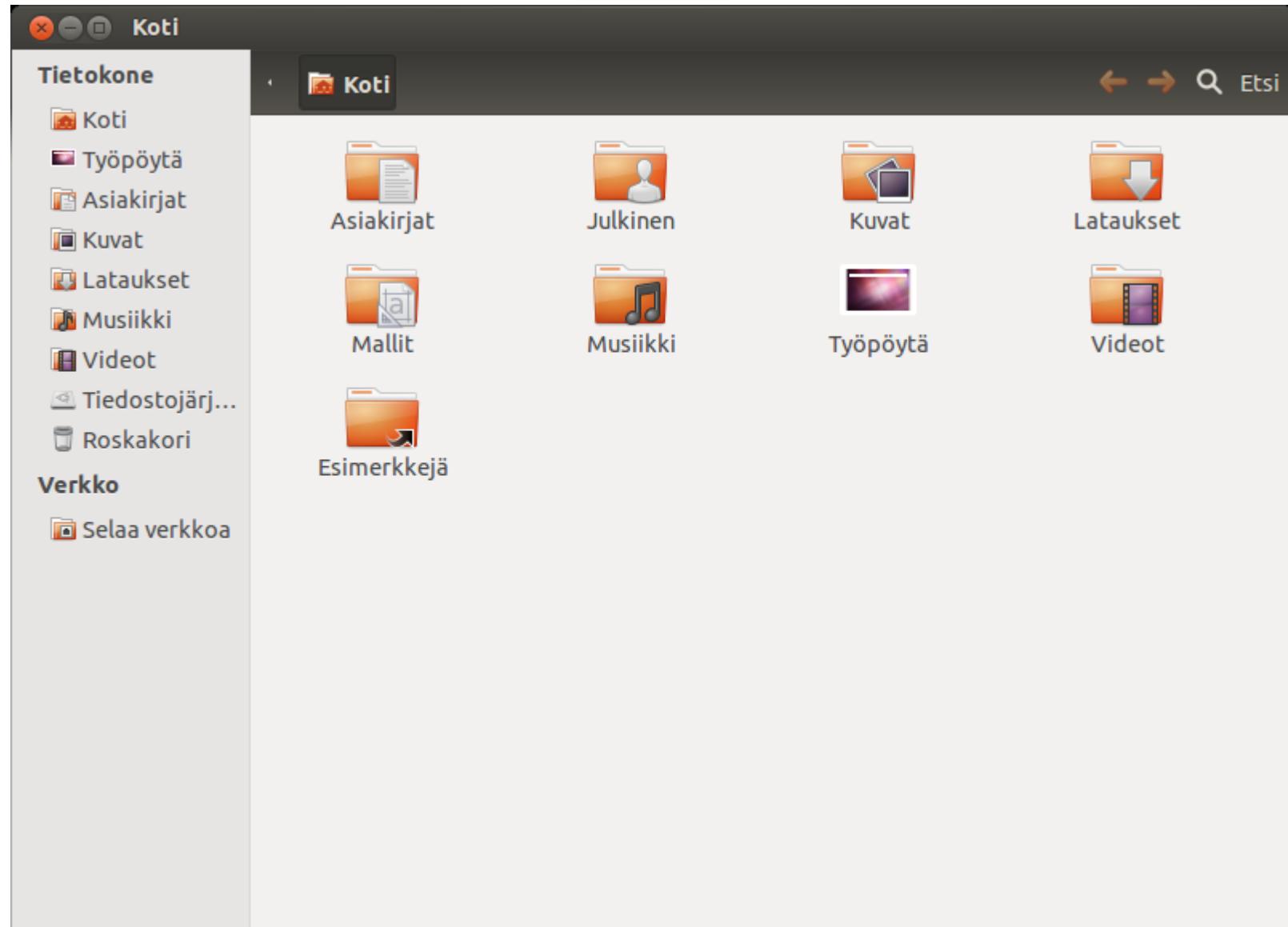
O gerenciador de janelas gráficas Unity3D



Painel de Pesquisa de Aplicativos e Arquivos



Nautilus - Gerenciador de Arquivos



Gedit - editor de texto ascii

ch02.html (~/sbird/trunk/dependencies/vendor/libgpod/docs/reference/html) - gedit

File Edit View Search Tools Documents Help

New Open Save Print... Undo Redo Cut Copy Paste Find Replace

install-sh depcomp libgpod-overrides.txt ch02.html

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=US-ASCII">
<title>iPod database components</title>
<meta name="generator" content="DocBook XSL Stylesheets V1.70.1">
<link rel="start" href="index.html" title="libgpod Reference Manual">
<link rel="up" href="index.html" title="libgpod Reference Manual">
<link rel="prev" href="libgpod-Low-level-functions.html" title="Low-level functions">
<link rel="next" href="libgpod-Tracks.html" title="Tracks">
<meta name="generator" content="GTK-Doc V1.7 (XML mode)">
<link rel="stylesheet" href="style.css" type="text/css">
<link rel="chapter" href="ch01.html" title="iPod database">
<link rel="chapter" href="ch02.html" title="iPod database components">
</head>
<body bgcolor="white" text="black" link="#0000FF" vlink="#840084" alink="#0000FF">
<table class="navigation" id="top" width="100%" summary="Navigation header" cellpadding="2" cellspacing="2"><tr valign="middle">
<td><a accesskey="p" href="libgpod-Low-level-functions.html"></a></td>
<td>#160;</td>
<td><a accesskey="h" href="index.html"></a></td>
<th width="100%" align="center">libgpod Reference Manual</th>
<td><a accesskey="n" href="libgpod-Tracks.html"></a></td>
</tr></table>
<div class="chapter" lang="en">
<div class="titlepage"><div><div><h2 class="title">
<a name="id2510305"></a>iPod database components</h2></div></div>
<div class="toc"><dl>
<dt>
<span class="refentrytitle"><a href="libgpod-Tracks.html">Tracks</a></span><span class="repurpose"> &#8212; Data structure to store
metadata about an iPod track</span>
</dt>
<dt>
<span class="refentrytitle"><a href="libgpod-Playlist.html">Playlist</a></span><span class="repurpose"> &#8212; Data structure to
store a list of songs</span>
</dt>
<dt>
<span class="refentrytitle"><a href="libgpod-Bitrate.html">Bitrate</a></span><span class="repurpose"> &#8212; Data structure to
store a list of bitrates</span>
</dt>
<dt>
<span class="refentrytitle"><a href="libgpod-Song.html">Song</a></span><span class="repurpose"> &#8212; Data structure to
store song information</span>
</dt>
</dl>
</div>

```

Ln 8, Col 37 INS

terminal - comunicação direta com o sistema

```
guimota@iobac-desktop-1: /tmp
lrwxrwxrwx  1 root root   36 Jan 14  2013 libnss3.so -> /usr/lib/x86_64-linux-
gnu/libnss3.so
drwx-----  4 root root 16384 Abr 11  2011 lost+found
drwxr-xr-x  2 root root  4096 Set 19 11:57 media
drwxr-xr-x  2 root root  4096 Out  7  2010 mnt
drwxr-xr-x  3 root root  4096 Out 17  2011 opt
dr-xr-xr-x 228 root root     0 Set 26 09:34 proc
drwx----- 17 root root  4096 Dez 12  2012 root
drwxr-xr-x 24 root root   960 Set 26 09:35 run
drwxr-xr-x  2 root root 12288 Set 19 09:18 sbin
drwxr-xr-x  2 root root  4096 Mai 10  2010 selinux
drwxr-xr-x  2 root root  4096 Out  7  2010 srv
drwxr-xr-x 13 root root     0 Set 26 09:34 sys
drwxrwxrwt 18 root root 12288 Set 26 10:59 tmp
drwxr-xr-x 11 root root  4096 Nov 21  2012 usr
drwxr-xr-x 13 root root  4096 Set 25 19:28 var
lrwxrwxrwx  1 root root   29 Set 11 11:32 vmlinuz -> boot/vmlinuz-3.2.0-53-gen
eric
lrwxrwxrwx  1 root root   29 Ago  6 12:10 vmlinuz.old -> boot/vmlinuz-3.2.0-51
-generic
guimota@iobac-desktop-1:~/tmp$ ls
4MKKhtiM.html.part
at-spi2
```

Terminal Linux

Terminal do Linux - Manipulação de Arquivos

clear limpa o terminal

> teste.txt cria arquivo vazio

ls lista o conteúdo de diretório

ls -la

rm <nome> remove arquivo

rm <nome> -R

mv move ou renomeia arquivo

cp copia arquivo

chmod <modo> <arq> muda permissões

Terminal do Linux - Manipulação de Diretórios

`pwd` informa o diretório atual

`cd` muda diretório atual

`cd ~ /`

`cd . . /`

`mkdir` cria diretório

`rmdir` remove diretório

Terminal do Linux - Manipulação de Programas

./progName executa programa

./progName &

ps lista processos em execução

kill derruba uma thread

sudo executa como se fosse o root

man <comando> exibe documentação

Terminal do Linux - Gerenciamento de Pacotes

apt-get gerencia a instalação de pacotes

update atualiza lista pacotes

upgrade atualiza pacotes

dist-upgrade atualiza distribuição

install baixa e instala pacotes

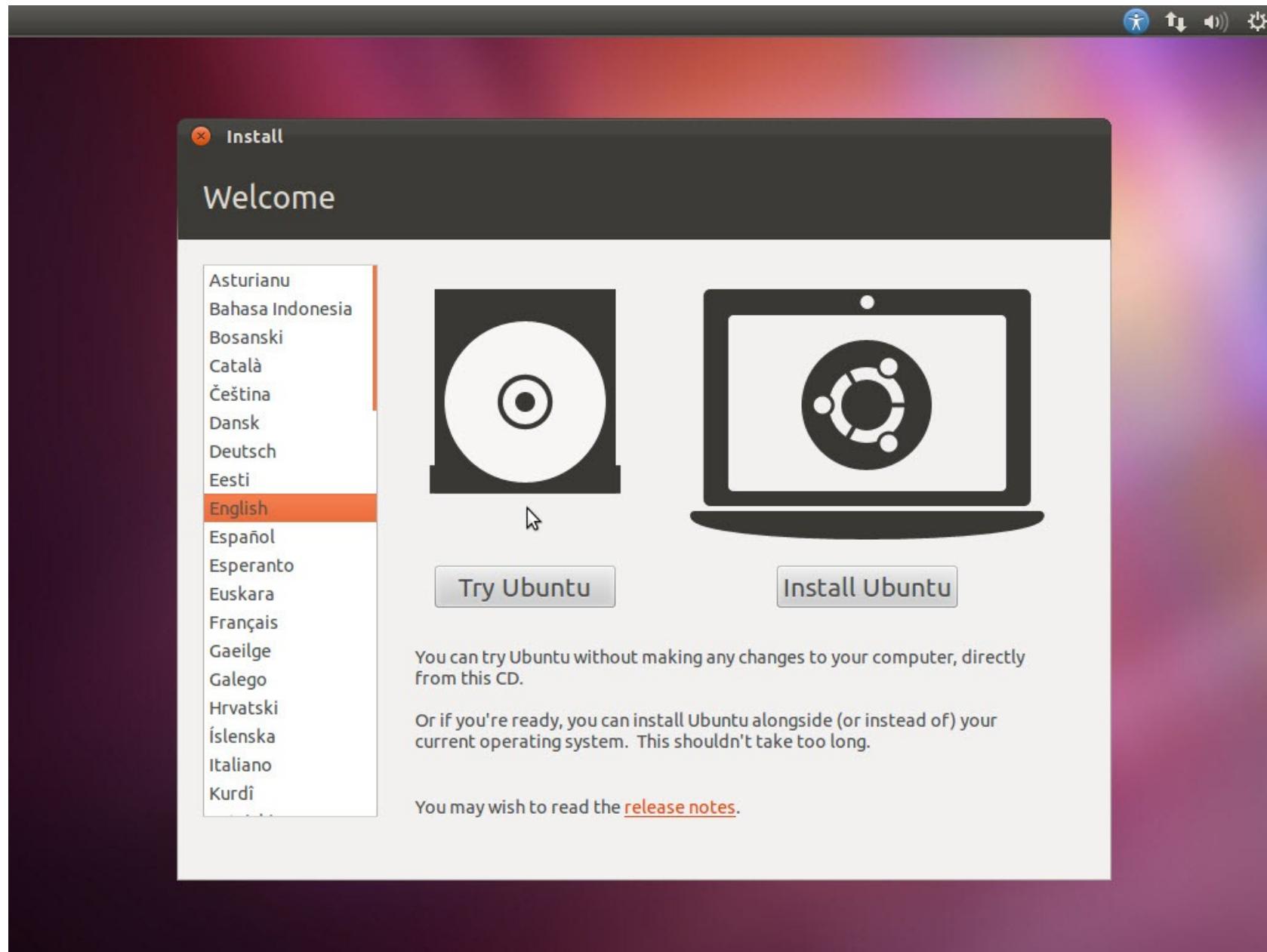
remove desinstala pacotes

purge purga pacotes

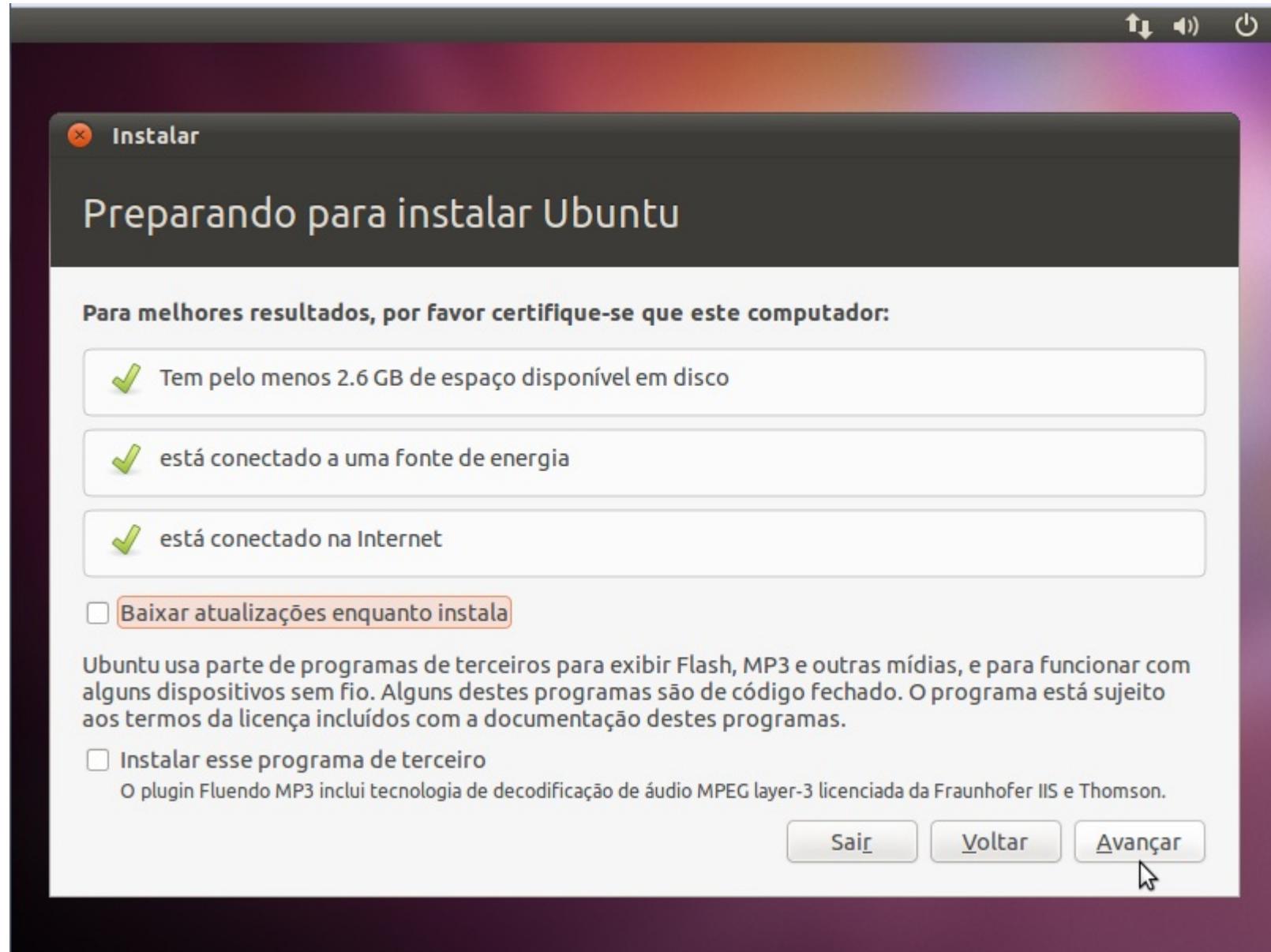
Instalação do Linux

<http://ubuntued.info/como-instalar-o-ubuntu-12-04-precise-pangolin-e-primeiros-passos>

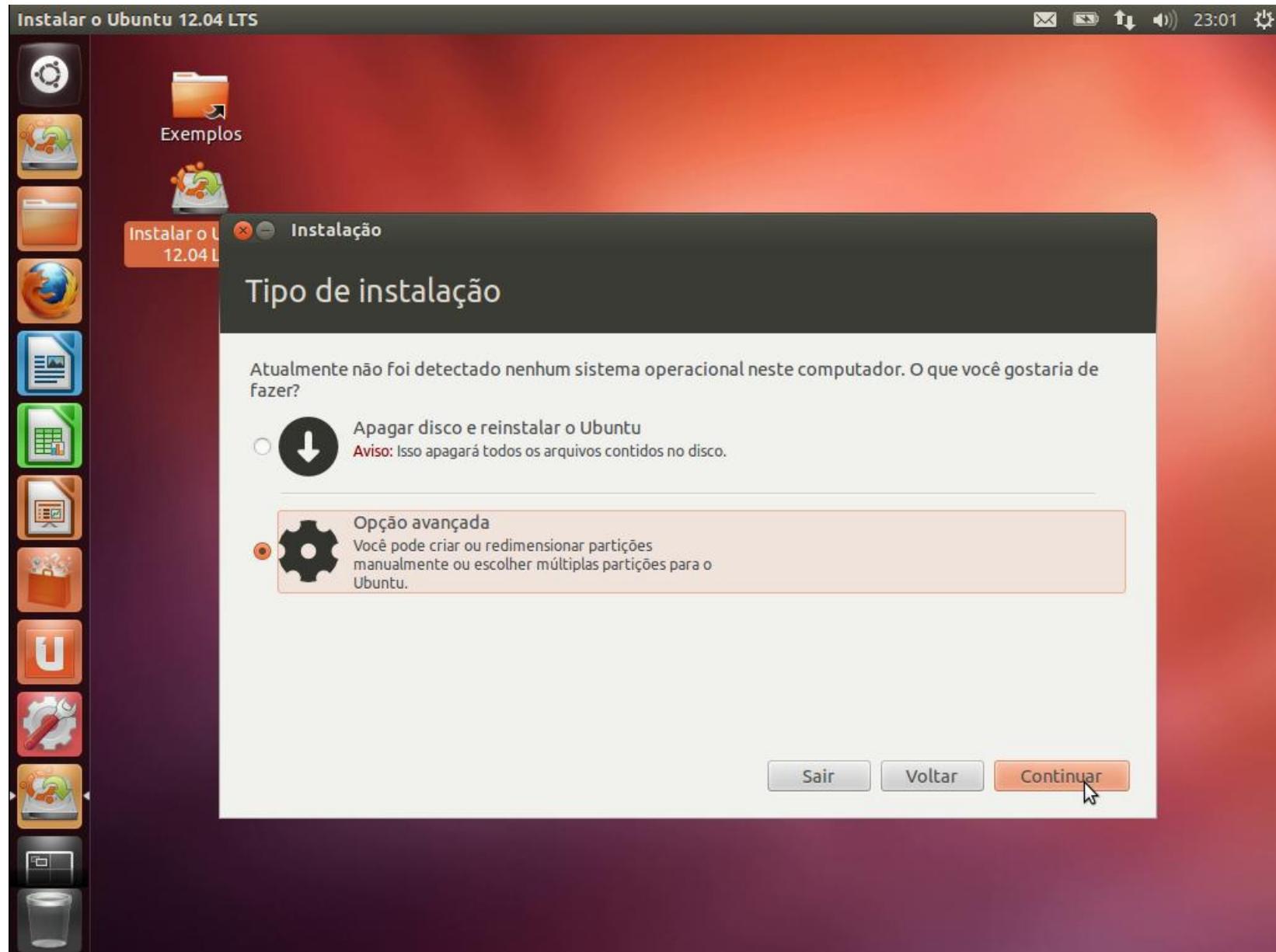
Instalação do Linux - Internationalização



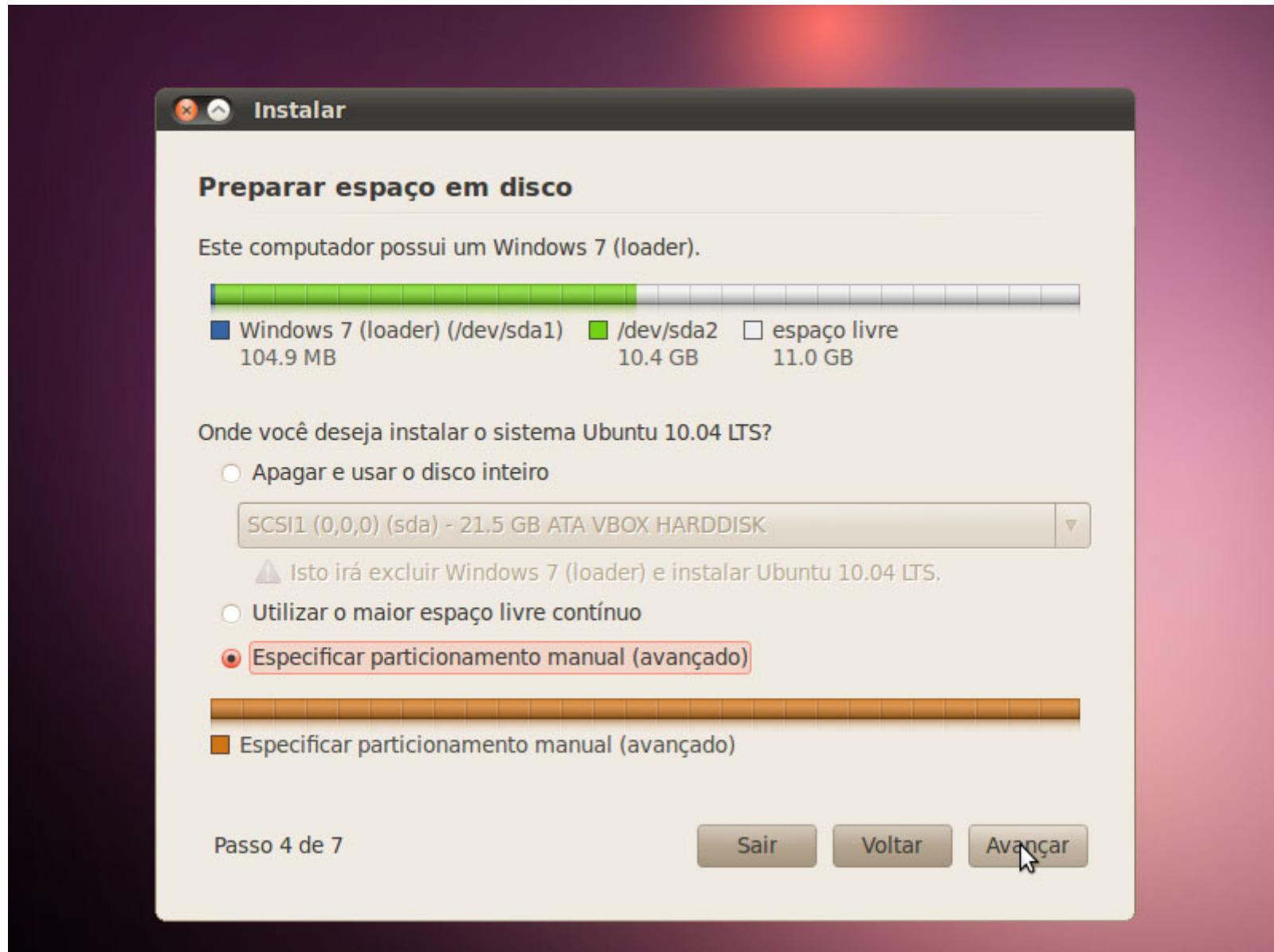
Instalação do Linux - Preparativos e Opções



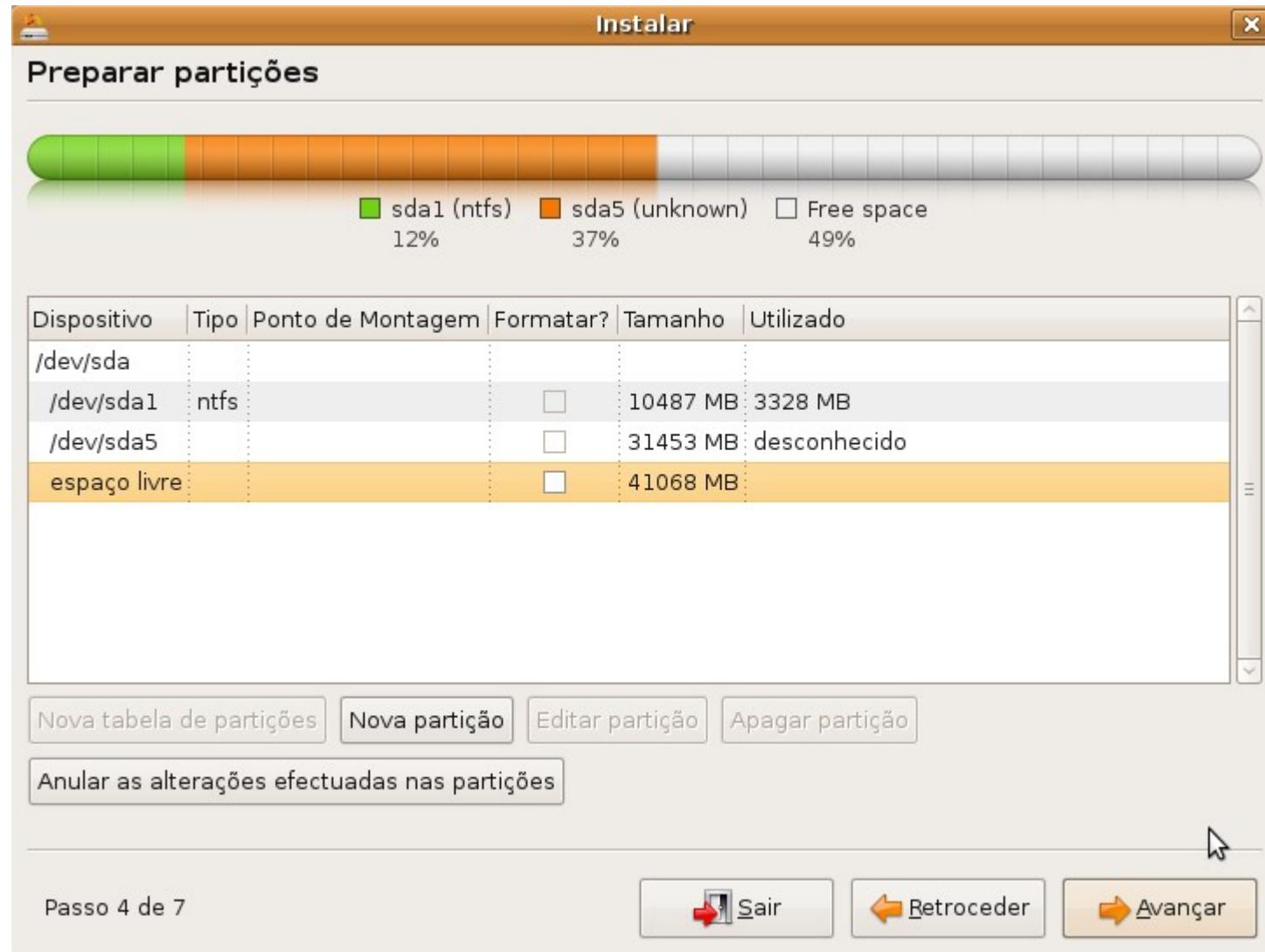
Instalação do Linux - Modos de Instalação



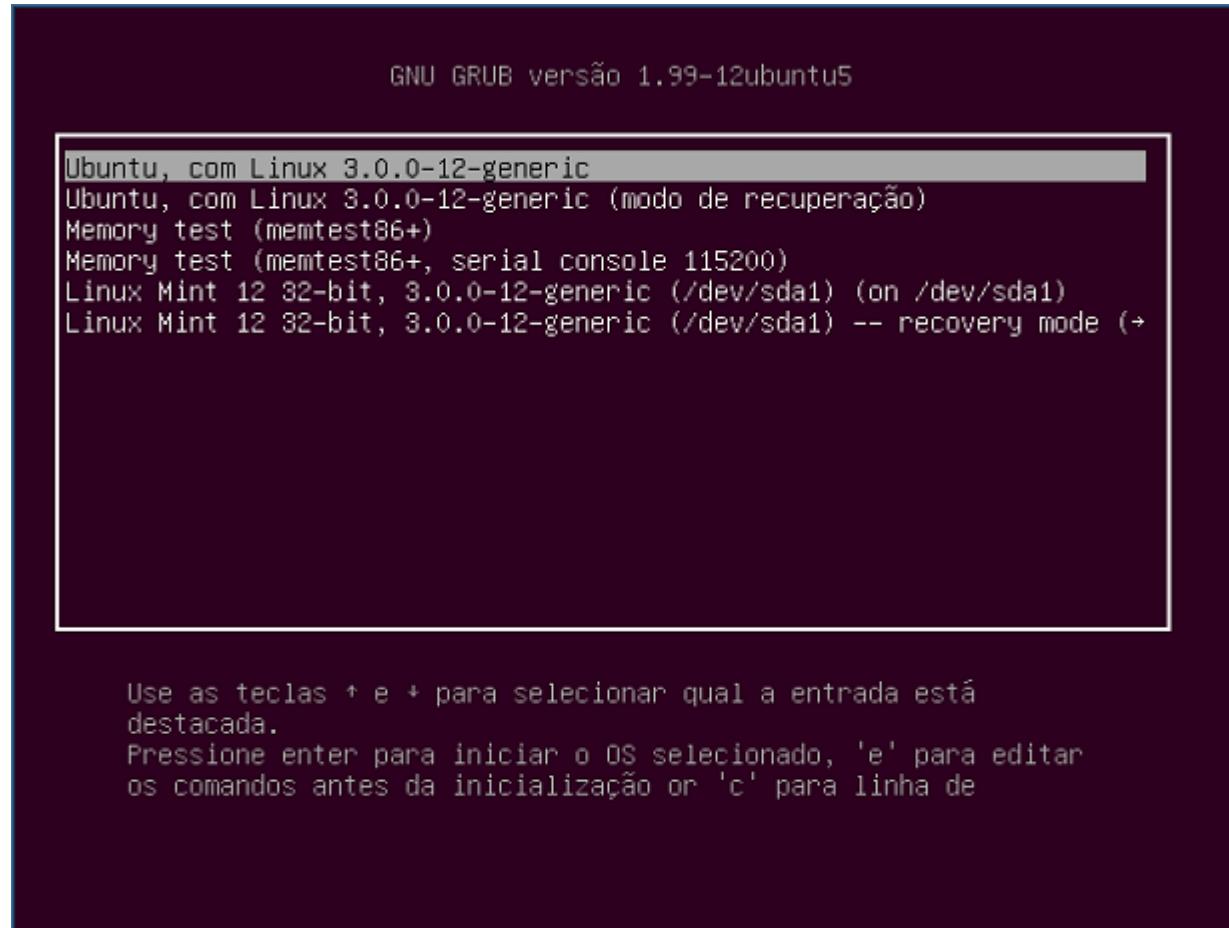
Instalação do Linux - Estado do HD



Instalação do Linux - gparted particionador de HDs

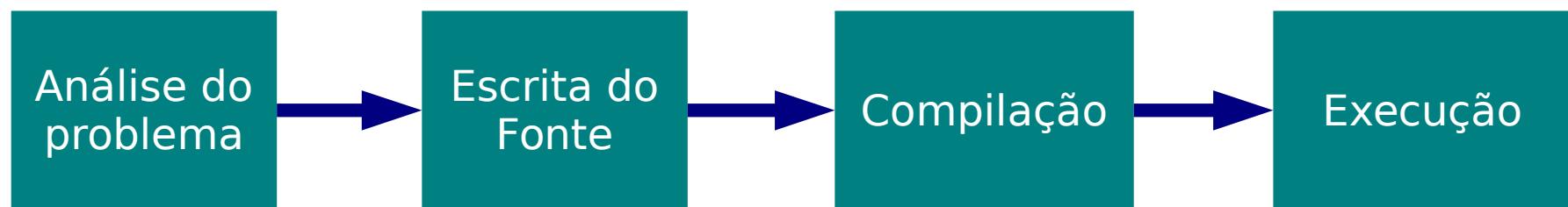


Linux no Desktop

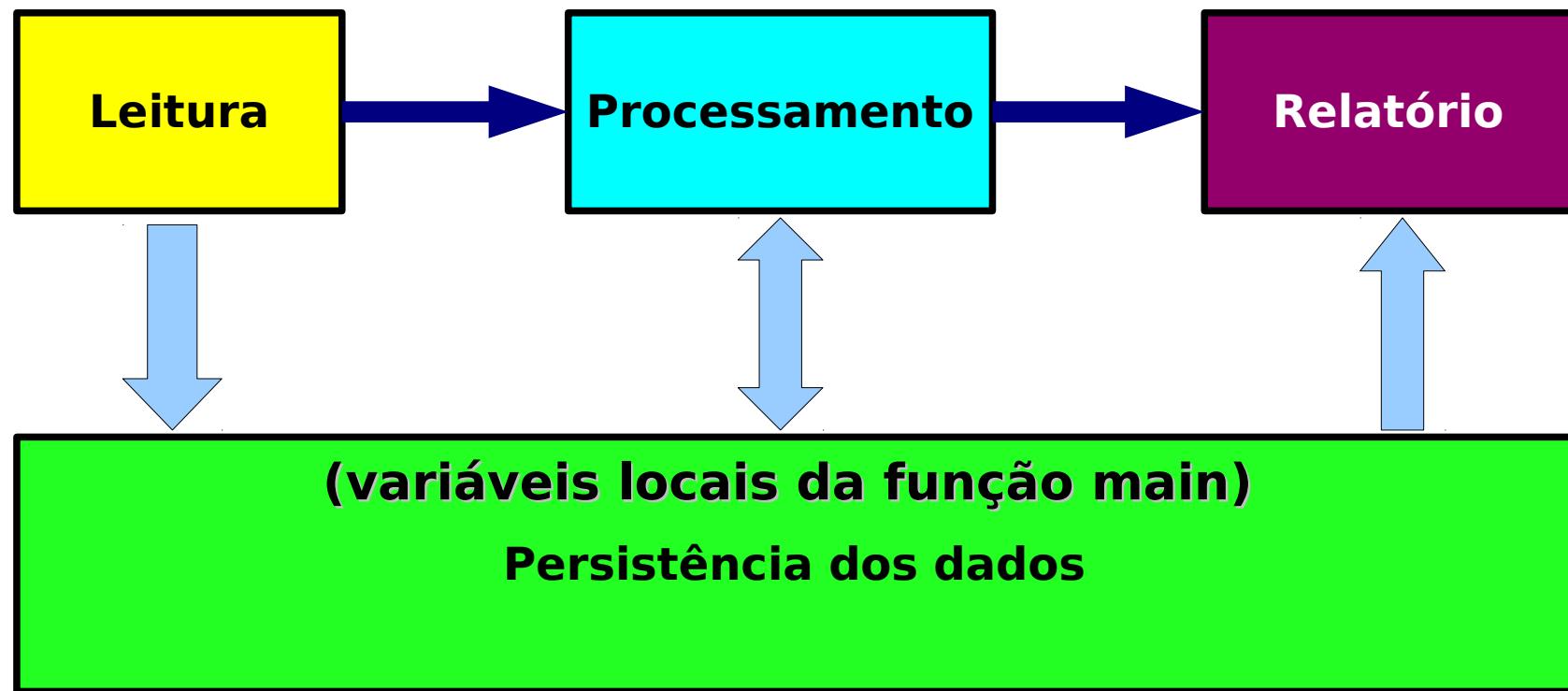


Conceitos básicos da linguagem C

Etapas da elaboração de um programa C



Subdivisão da Computação



Primeiro Programa

Primeiro Programa

- Escreva um programa que leia dois números inteiros, calcule a divisão e imprima o resultado na tela do terminal.
- É preciso testar se o segundo número é zero.
 - Padronização do programa principal
 - Declaração e uso de variáveis
 - Operações aritméticas
 - Seleção de fluxo
 - Operação de leitura do teclado
 - Operação de escrita na tela

Primeiro Programa

primeiroProg.c

```
#include <stdio.h>
int main ()
{
    int a, b, c;
    scanf ("%d", &a);
    scanf ("%d", &b);
    if (b!=0)
    {
        c = a / b;
        printf ("%d\n", c);
    }
    else
        printf ("Divisão por zero");
    return 0;
}
```

Primeiro Programa

Inclusão de
biblioteca

primeiroProg.c

```
#include <stdio.h>
int main ()
{
    int a, b, c;
    scanf ("%d", &a);
    scanf ("%d", &b);
    if (b!=0)
    {
        c = a / b;
        printf ("%d\n", c);
    }
    else
        printf ("Divisão por zero");
    return 0;
}
```

Primeiro Programa

primeiroProg.c

Programa principal

```
#include <stdio.h>
int main ()
{
    int a, b, c;
    scanf ("%d", &a);
    scanf ("%d", &b);
    if (b!=0)
    {
        c = a / b;
        printf ("%d\n", c);
    }
    else
        printf ("Divisão por zero");
    return 0;
}
```

Primeiro Programa

primeiroProg.c

Declaração de variáveis

```
#include <stdio.h>
int main ()
{
    int a, b, c;
    scanf ("%d", &a);
    scanf ("%d", &b);
    if (b!=0)
    {
        c = a / b;
        printf ("%d\n", c);
    }
    else
        printf ("Divisão por zero");
    return 0;
}
```

Primeiro Programa

primeiroProg.c

Operações de
entrada

Operações de
saída

```
#include <stdio.h>
int main ()
{
    int a, b, c;
    scanf ("%d", &a);
    scanf ("%d", &b);
    if (b!=0)
    {
        c = a / b;
        printf ("%d\n", c);
    }
    else
        printf ("Divisão por zero");
    return 0;
}
```

`printf()` e `scanf()`

```
int printf(char* format, arg1, arg2, ...);
```

```
printf("%d\n", c);
```

```
int scanf(char* format, arg1, arg2, ...);
```

```
scanf("%d", &b);
```

| Caractere | Tipo do argumento |
|-----------|-------------------|
| %d | int |
| %c | int |
| %s | string |
| %f | float |

<http://www.cplusplus.com/reference/cstdio/printf/>

Primeiro Programa

primeiroProg.c

expressão

```
#include <stdio.h>
int main ()
{
    int a, b, c;
    scanf ("%d", &a);
    scanf ("%d", &b);
    if (b!=0)
    {
        c = a / b;
        printf ("%d\n", c);
    }
    else
        printf ("Divisão por zero");
    return 0;
}
```

Primeiro Programa

primeiroProg.c

```
#include <stdio.h>
int main ()
{
    int a, b, c;
    scanf ("%d", &a);
    scanf ("%d", &b);
    if (b != 0)
    {
        c = a / b;
        printf ("%d\n", c);
    }
    else
        printf ("Divisão por zero");
    return 0;
}
```

Seleção de fluxo

if - else

```
/* expressao != 0 ? */
```

```
if (expressao)
```

```
    comando-1;
```

```
else
```

```
    comando-2;
```

```
/* note que é expressão e  
não condição */
```

Compilando e executando o primeiro programa

primeiroProg.c

```
#include <stdio.h>
int main ()
{
    int a, b, c;
    scanf ("%d", &a);
    scanf ("%d", &b);
    if (b!=0)
    {
        c = a / b;
        printf ("%d\n", c);
    }
    else
        printf ("Divisão por zero");
    return 0;
}
```

```
> gcc primeiroProg.c
> ./a.out
```

Compilando e executando o primeiro programa

primeiroProg.c

```
#include <stdio.h>
int main ()
{
    int a, b, c;
    scanf ("%d", &a);
    scanf ("%d", &b);
    if (b!=0)
    {
        c = a / b;
        printf ("%d\n", c);
    }
    else
        printf ("Divisão por zero");
    return 0;
}
```

```
> gcc primeiroProg.c -o primeiro
> ./primeiro
```

Primeiro Problema

Primeiro Problema

- Escreva um programa que imprima uma tabela dos valores de temperatura em Fahrenheit e Celsius.
 - Valor inicial: 0 °F
 - Valor Final: 300 °F
 - Incremento: 20 °F

$$C = \frac{5}{9} \times (F - 32)$$

| Fahrenheit | Celsius |
|------------|---------|
| 0 | -17.8 |
| 20 | -6.7 |
| 40 | 4.4 |
| 60 | 15.6 |
| ... | ... |
| 300 | 148.9 |

Resultado do Programa Fahrenheit Celsius

```
#include <stdio.h>

// protótipo da função
float fahr2cels (int);

int main()
{
    int fahr;
    for (fahr = 0; fahr <= 300; fahr = fahr + 20)
        printf("%d %f\n", fahr, fahr2cels(fahr));
    return 0;
}

// implementação da função
float fahr2celc (int fahr)
{
    return (5.0 / 9)* (fahr-32);
```

```
$ ./progPag026
0 -17.8
20 -6.7
40 4.4
60 15.6
80 26.7
100 37.8
120 48.9
140 60.0
160 71.1
180 82.2
200 93.3
220 104.4
240 115.6
260 126.7
280 137.8
300 148.9
$
```

O comando for

```
for (pré-execução; condição; pós-iteração)
{
    . . . ;
}
```

Declaração de uma função

- Protótipo da função

```
Tipo_Retorno nome_func(Tipos_Args);
```

- Implementação da função

```
Tipo_Retorno nome_func(argumentos)
{
    Declarações e instruções;
    return expressão;
}
```

Resultado do Programa Fahrenheit Celsius

```
#include <stdio.h>

// protótipo da função
float fahr2cels (int);

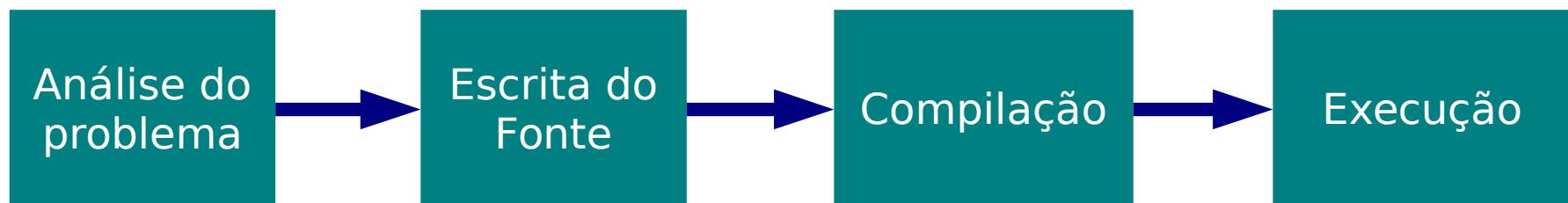
int main()
{
    int fahr;
    for (fahr = 0; fahr <= 300; fahr = fahr + 20)
        printf("%d %f\n", fahr, fahr2cels(fahr));
    return 0;
}

// implementação da função
float fahr2celc (int fahr)
{
    return (5.0 / 9)* (fahr-32);
```

```
$ ./progPag026
0 -17.8
20 -6.7
40 4.4
60 15.6
80 26.7
100 37.8
120 48.9
140 60.0
160 71.1
180 82.2
200 93.3
220 104.4
240 115.6
260 126.7
280 137.8
300 148.9
$
```

Exercício U1.1

- Escreva, utilizando o comando `for`, um programa que imprima a tabela de temperatura de Celsius para Fahrenheit
 - Valor inicial: -10 °C
 - Valor Final: 50 °C
 - Incremento: 10 °C
- Submeta sua solução no moodle



FIM