

Unidade II - Compilação de Programas C no Linux

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

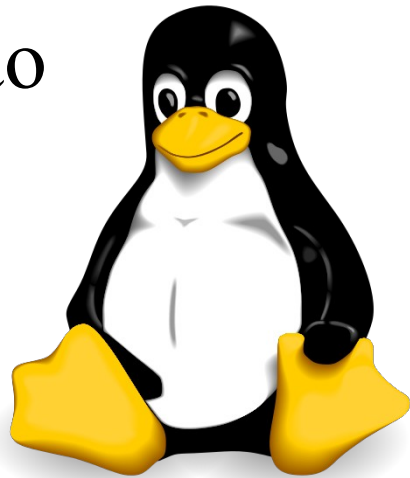
ANSI C

```
#include <stdio.h>
main ()
{
    printf("Hello World!");
}
```

Que assuntos serão abordados nesta unidade?

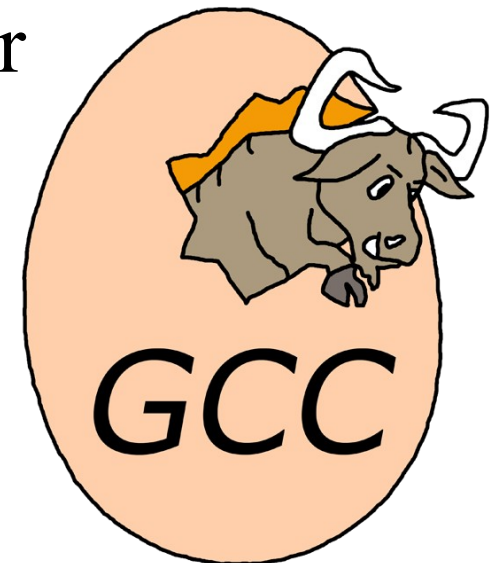
- Sistema Linux:

- arquitetura
- sistema de arquivos
- uso desktop
- shell
- instalação



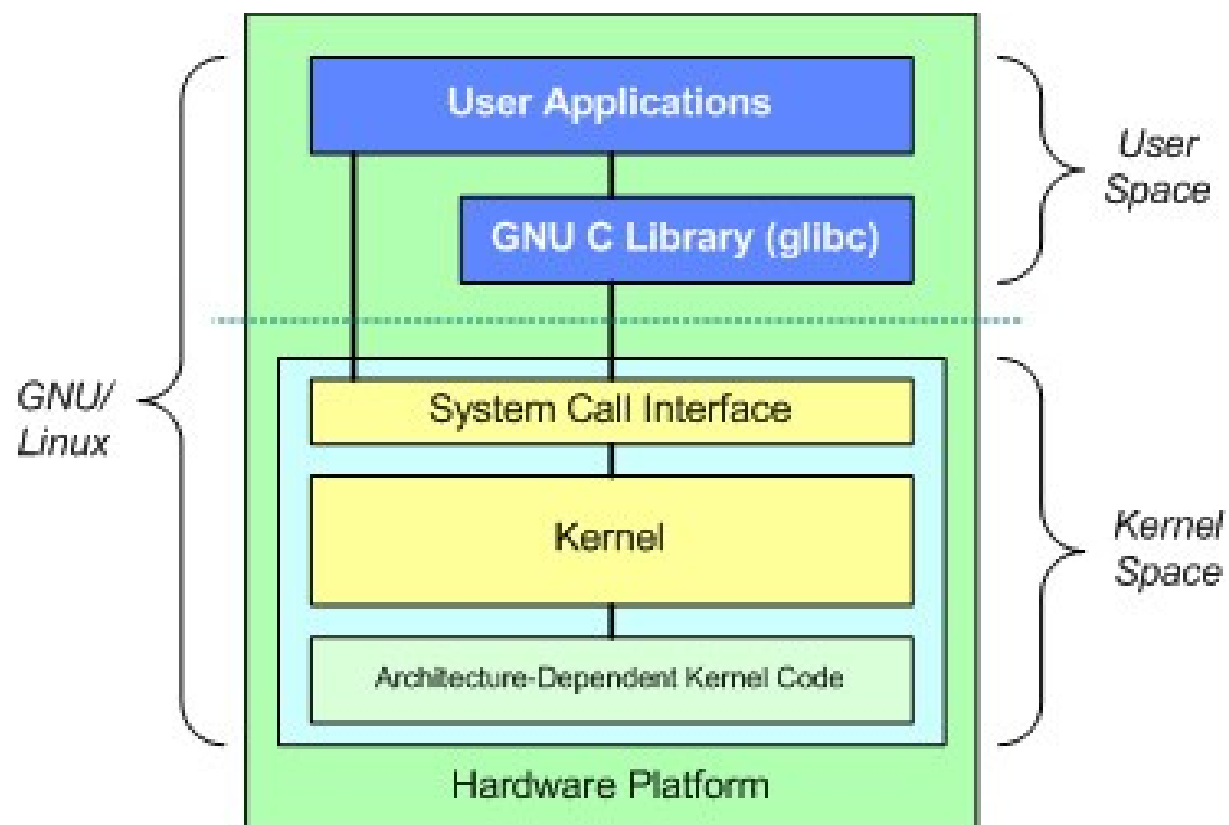
- GCC:

- arquitetura
- compilação
- linkagem
- makefile
- debugger

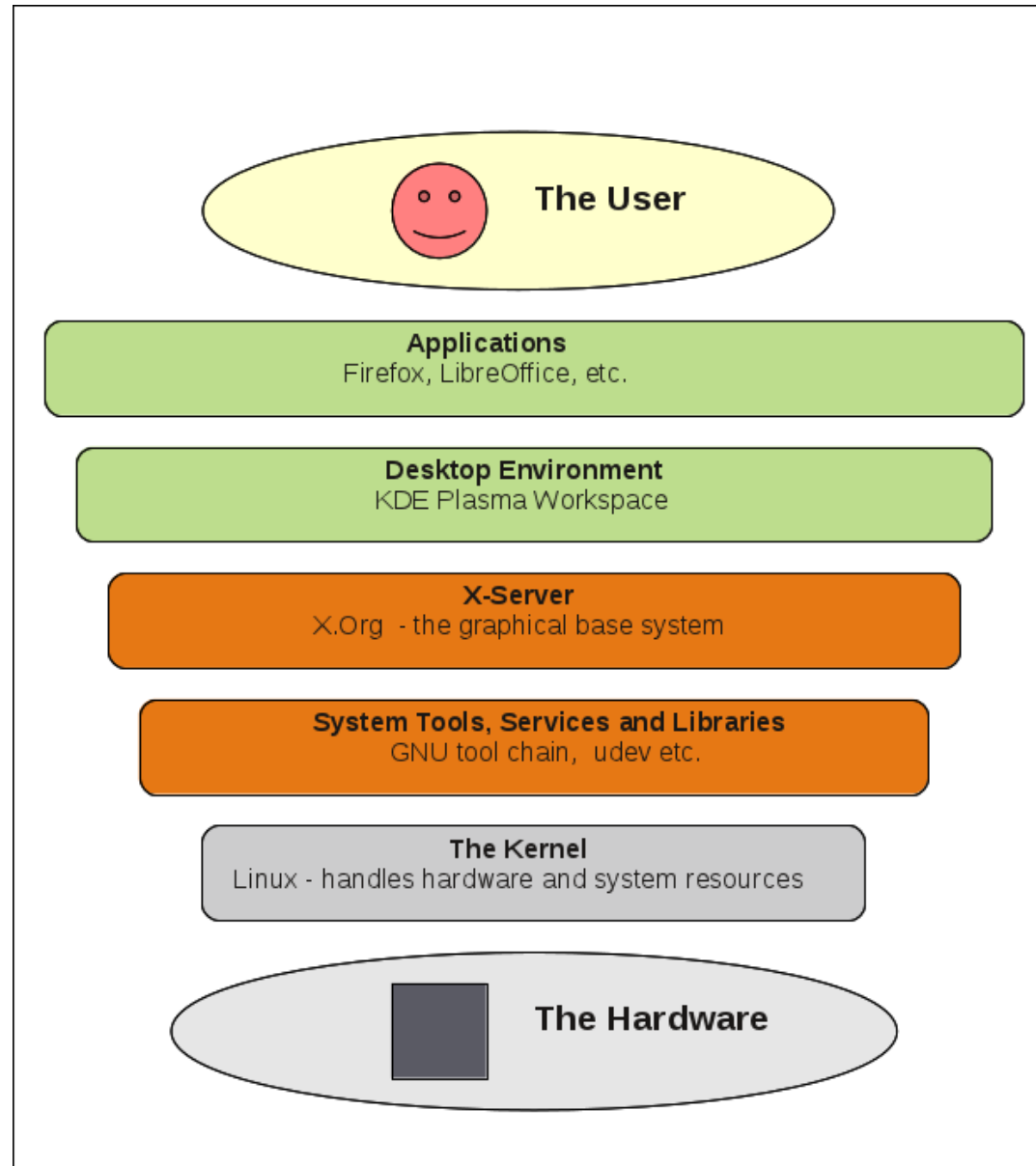


Introdução ao Linux

Visão Geral da Arquitetura do Sistema



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 |

Estrutura de Diretórios

`/usr` arquivos de uso comum

`/bin` aplicativos

`/include` headers de bibliotecas

`/lib` bibliotecas usadas pelos aplicativos

`/sbin` ferramentas pós-boot

`/share` arquivos independentes de plataforma

`/doc` documentação dos aplicativos

Ubuntu Linux no Desktop

O gerenciador de janelas gráficas Unity3D

Menu do aplicativo

Barra de Estado

painel

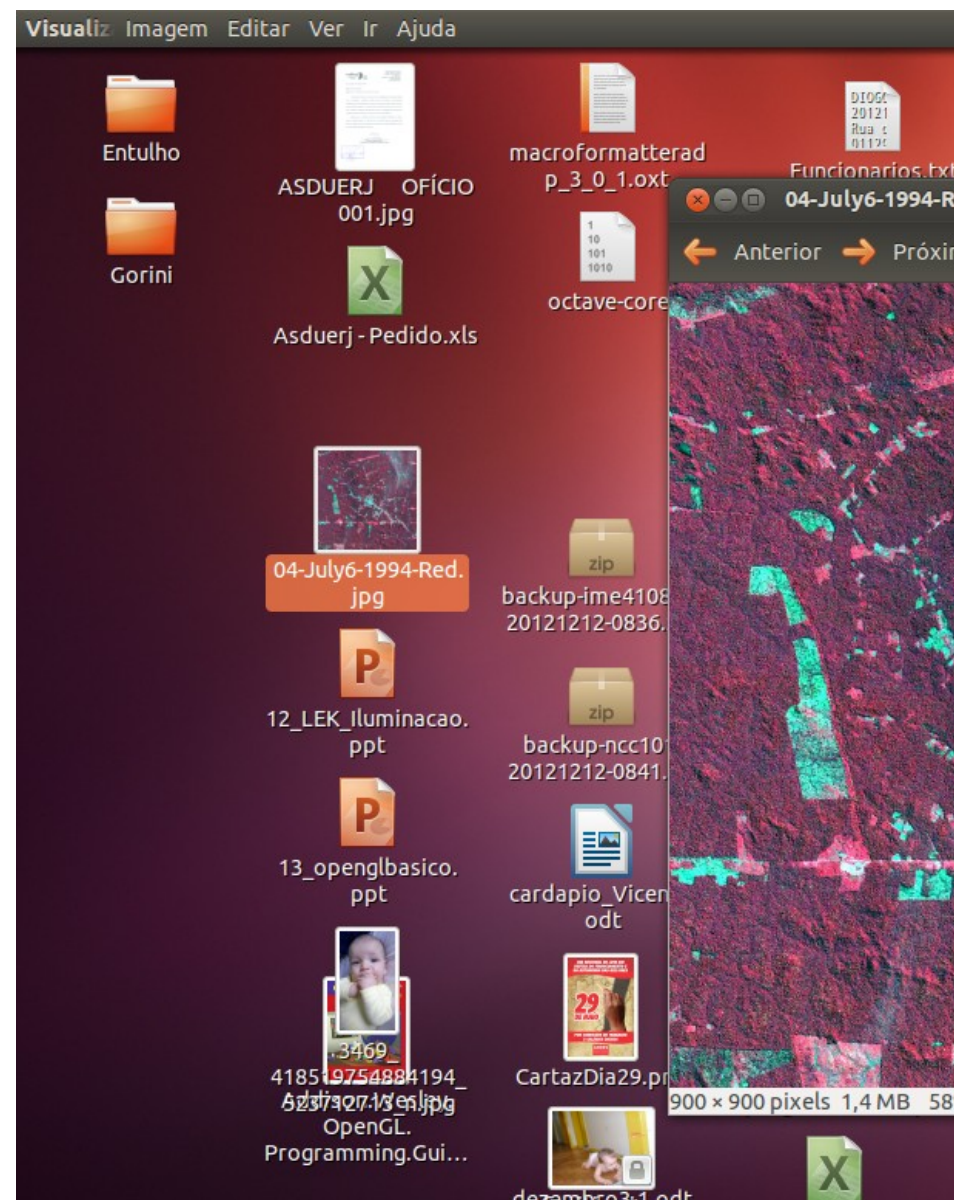
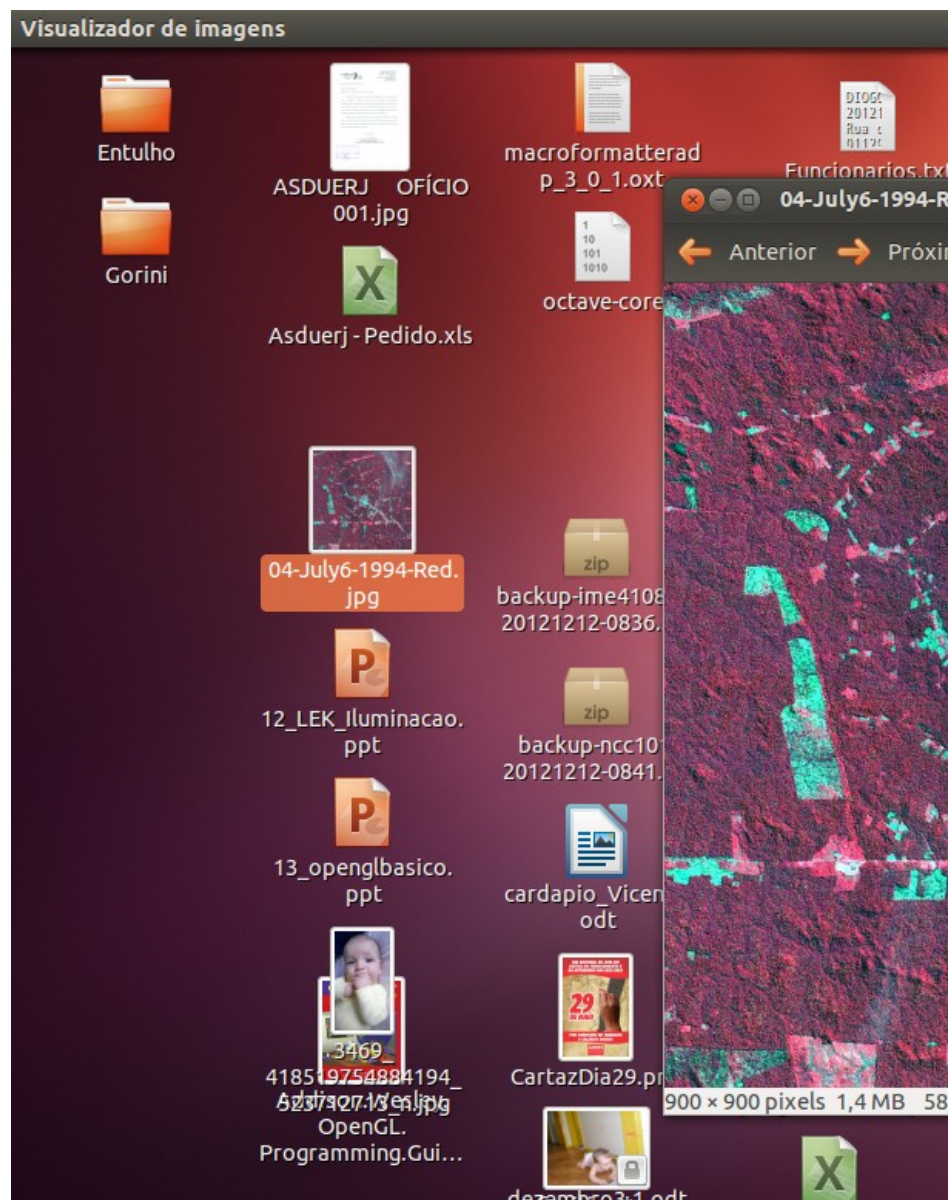
Configuração e desligamento

Área de Trabalho

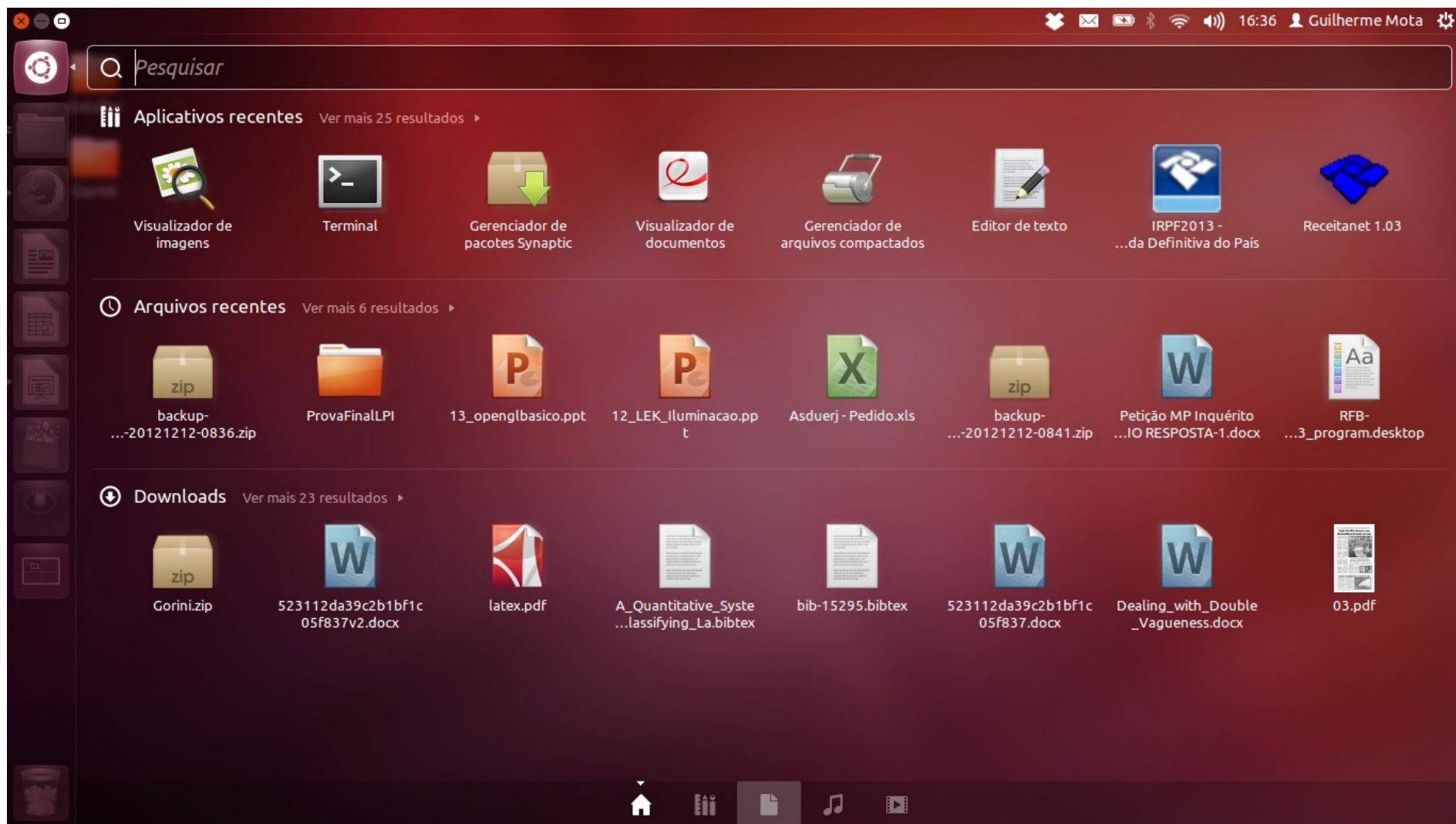
Lista de Acesso Rápido

Barra do Lançador

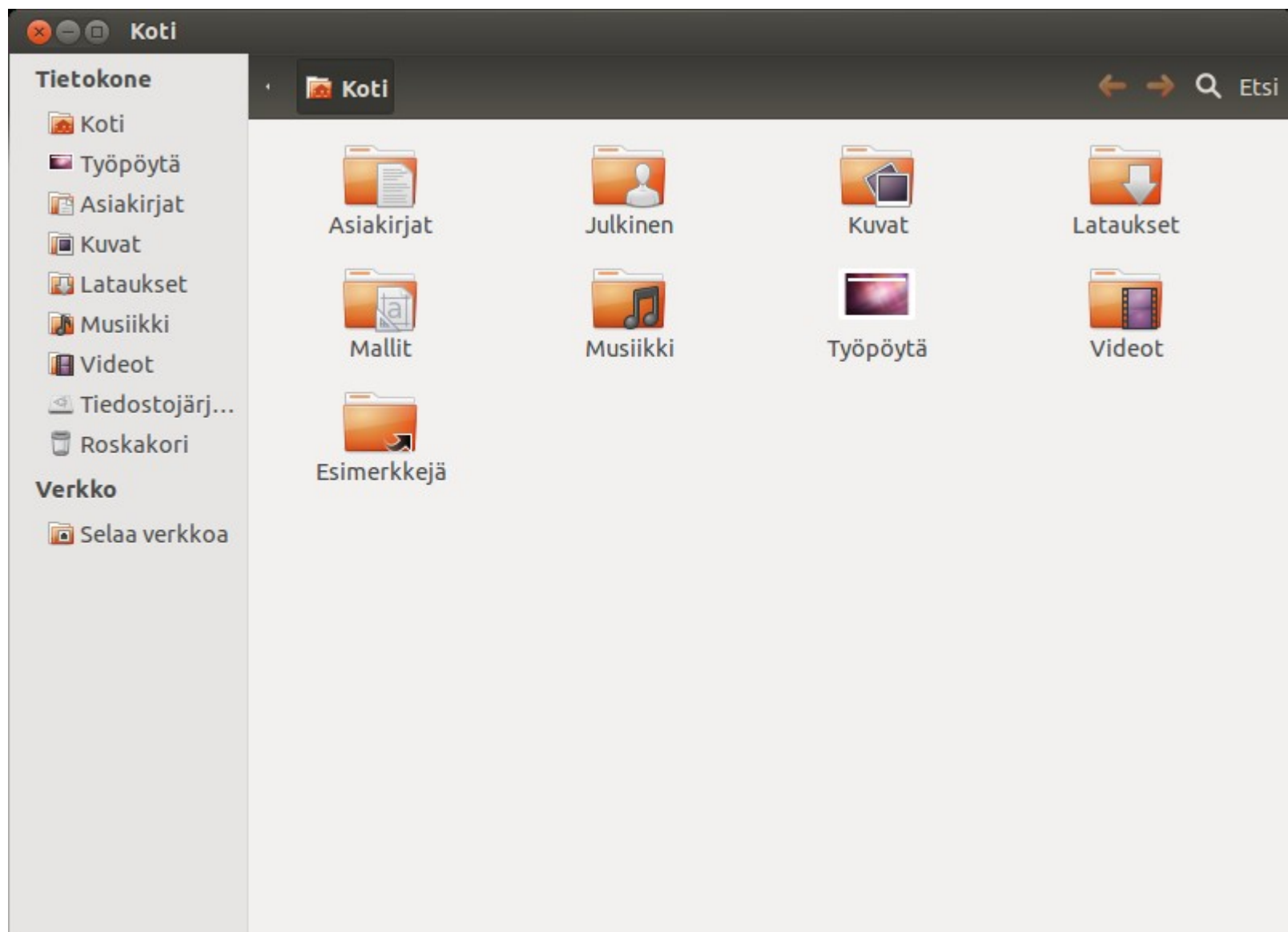
Menu do Aplicativo



Painel de Pesquisa de Aplicativos e Arquivos



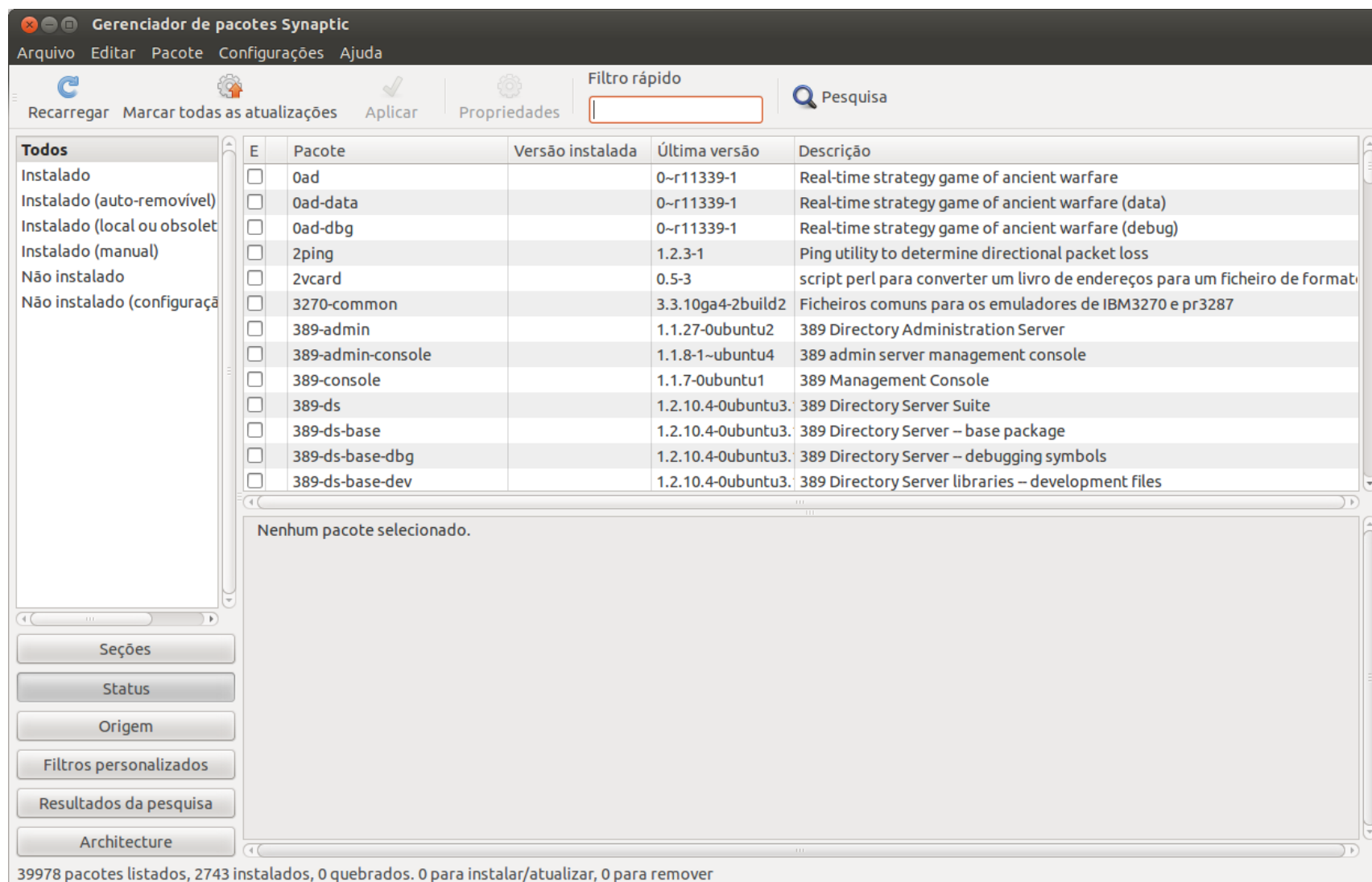
Nautilus - Gerenciador de Arquivos



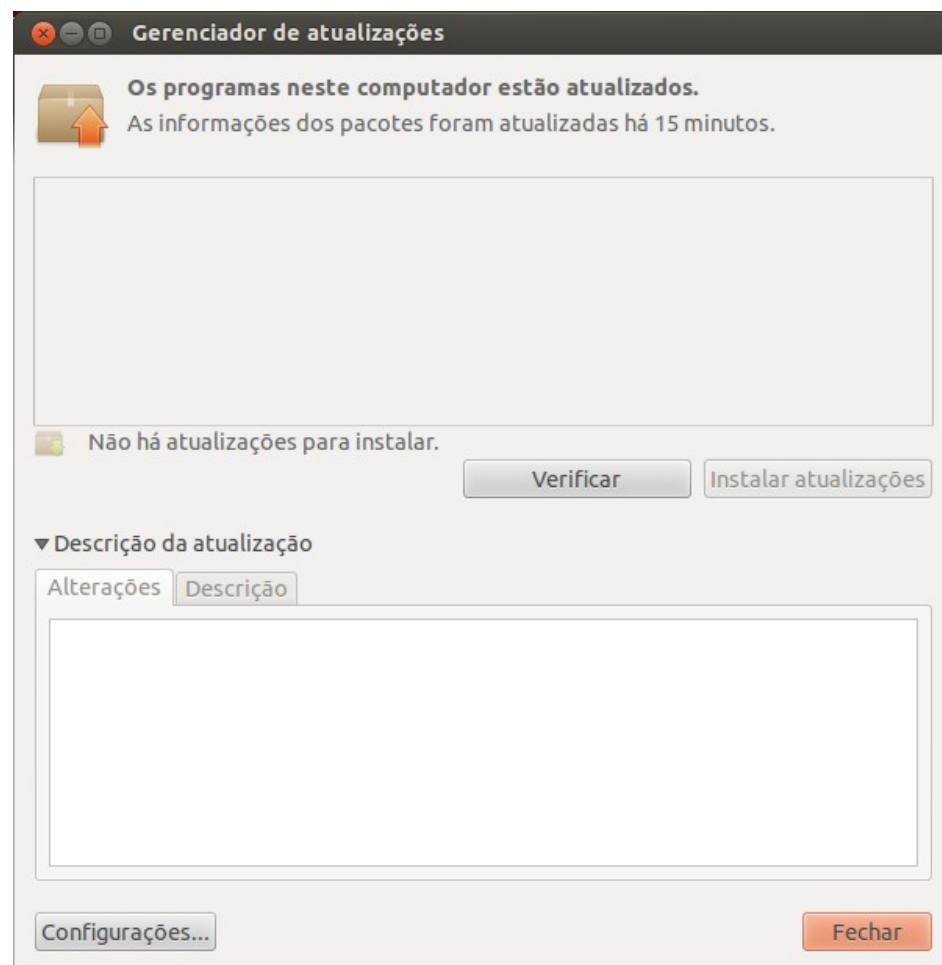
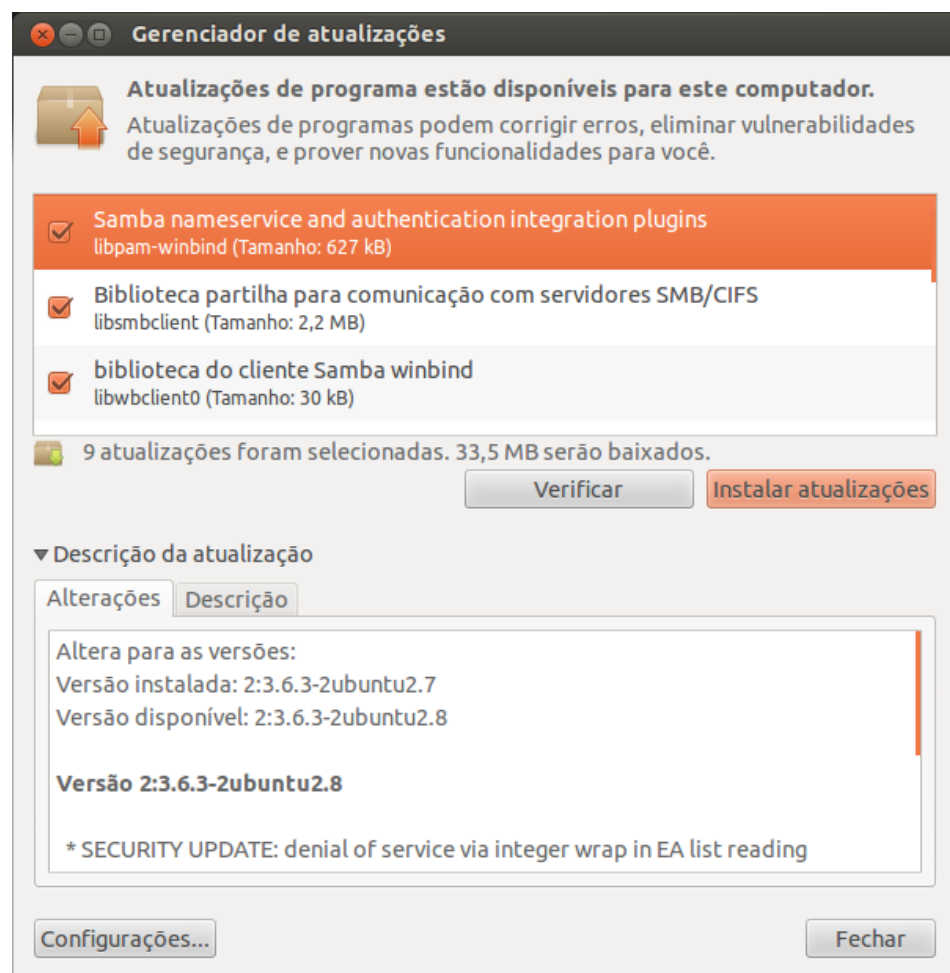
Gedit - editor de texto ascii

```
<!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>
<div class="toc"><dl>
<dt>
<span class="refentrytitle"><a href="libgpod-Tracks.html">Tracks</a></span><span class="refpurpose"> &#8212; Data structure to store
metadata about an iPod track</span>
</dt>
<dt>
<span class="refentrytitle"><a href="libgpod-Playlists.html">Playlists</a></span><span class="refpurpose"> &#8212; Data structure to
```

Synaptic - gerenciador de pacotes



update-manager - gerenciador de atualizações



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:/$ cd tmp
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

Terminal do Linux - Miscelânea

echo escreve mensagem no monitor

more limita e controla exibição de texto

grep seleciona parte do texto

| conecta E/S de dois programas

> redefine canal de saída

>> redefine canal de saída - modo apensar

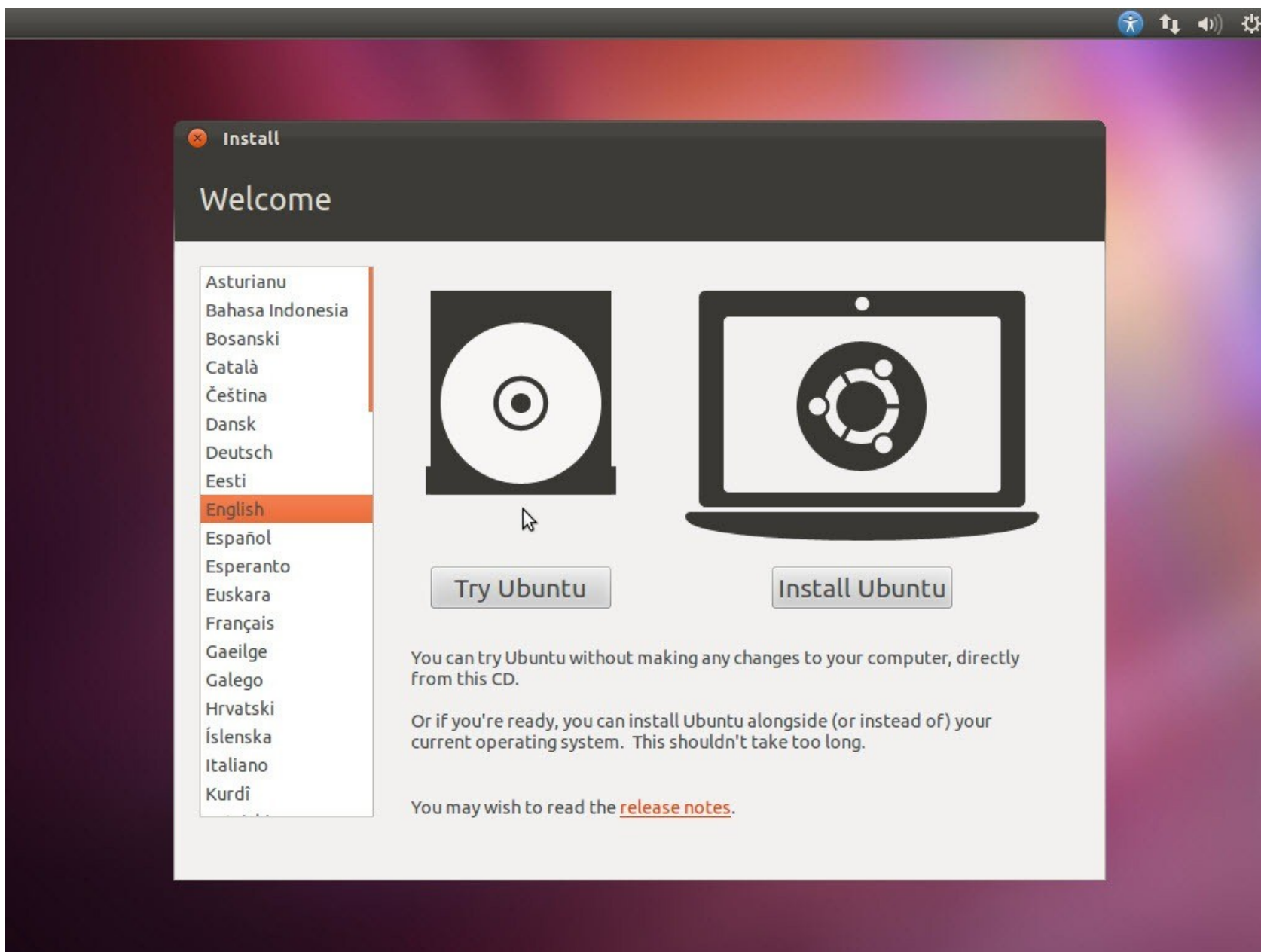
< redefine canal de entrada

gcc compila programas

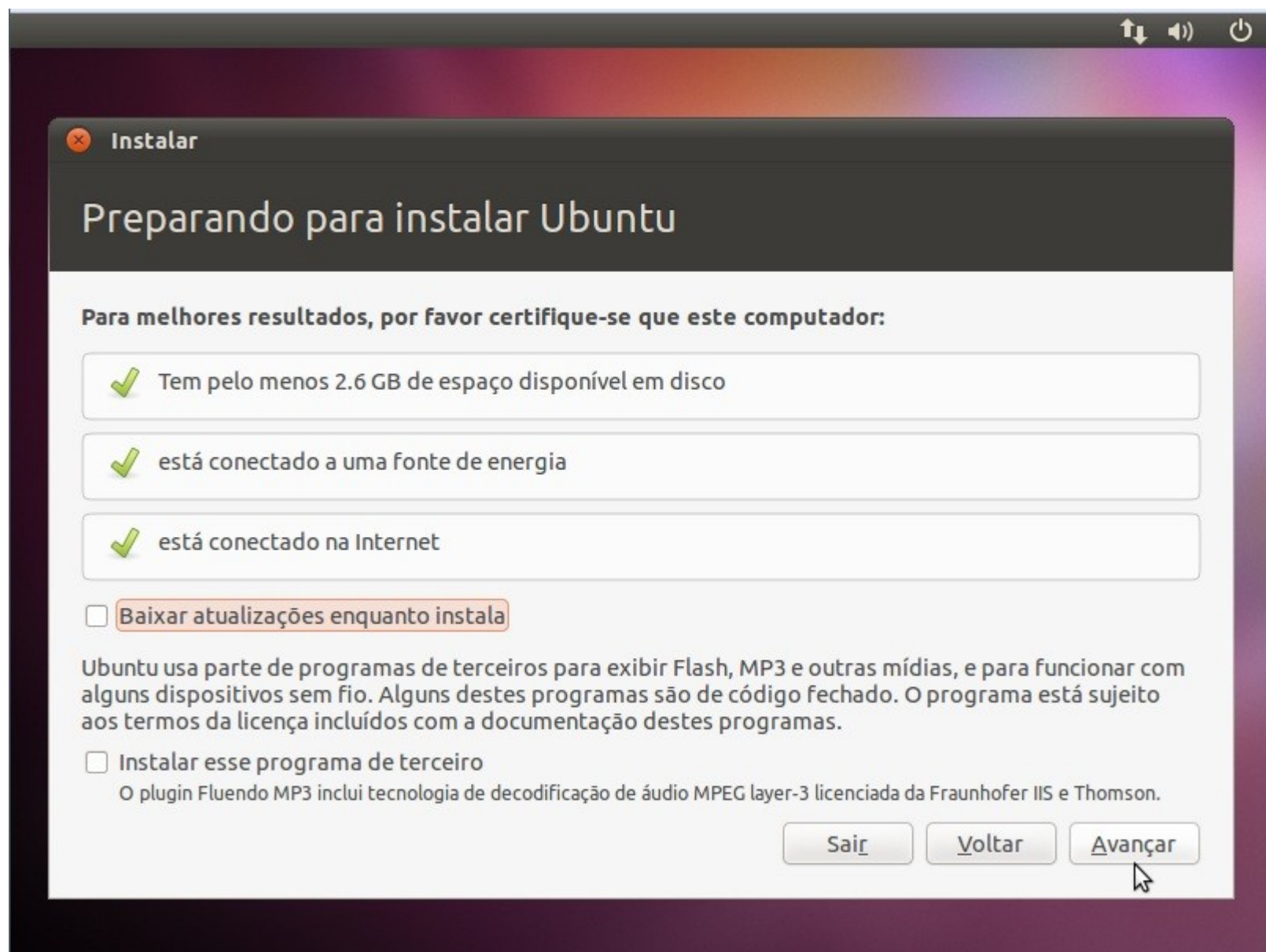
Instalação do Linux

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

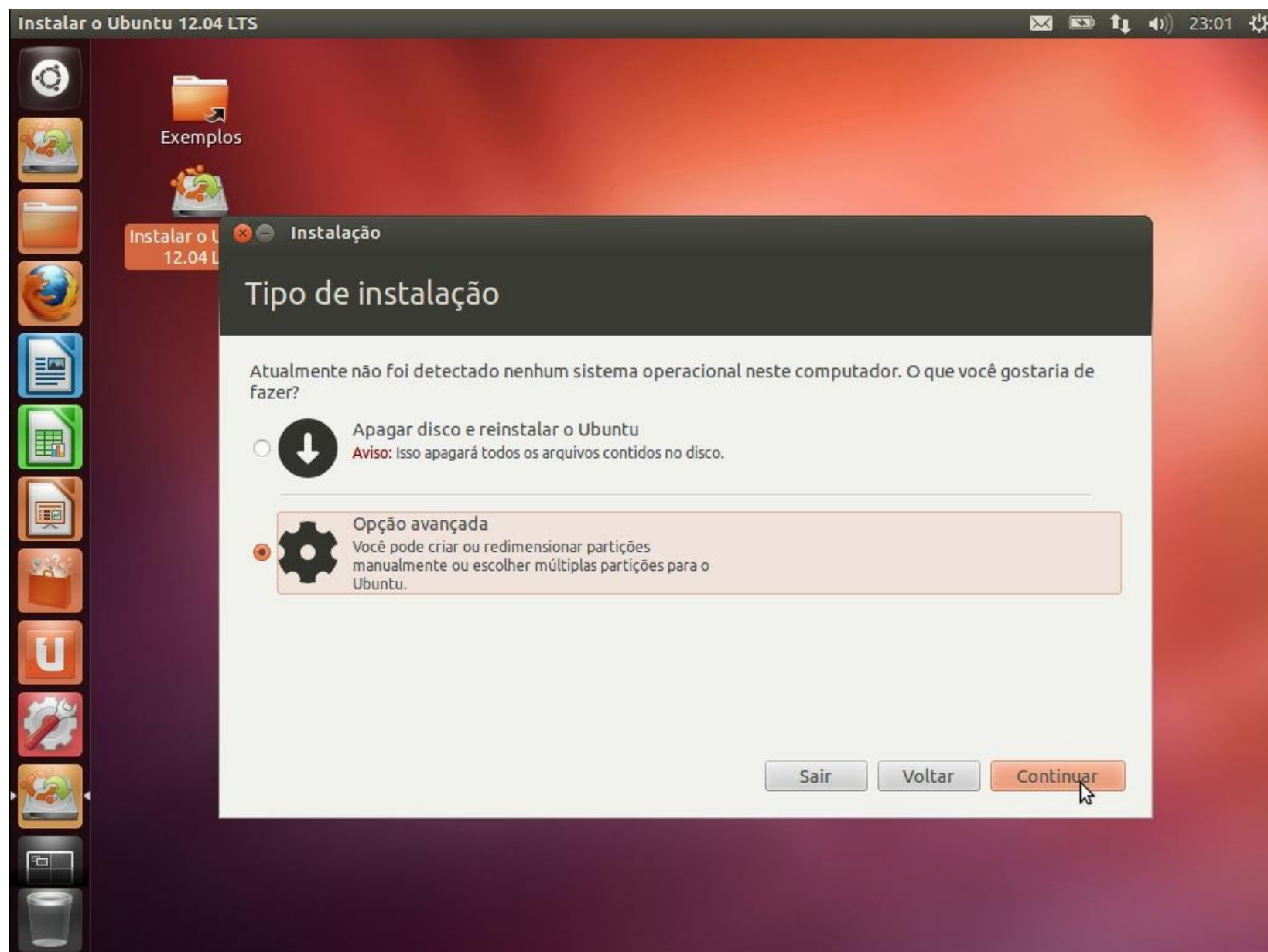
Instalação do Linux - Internacionalização



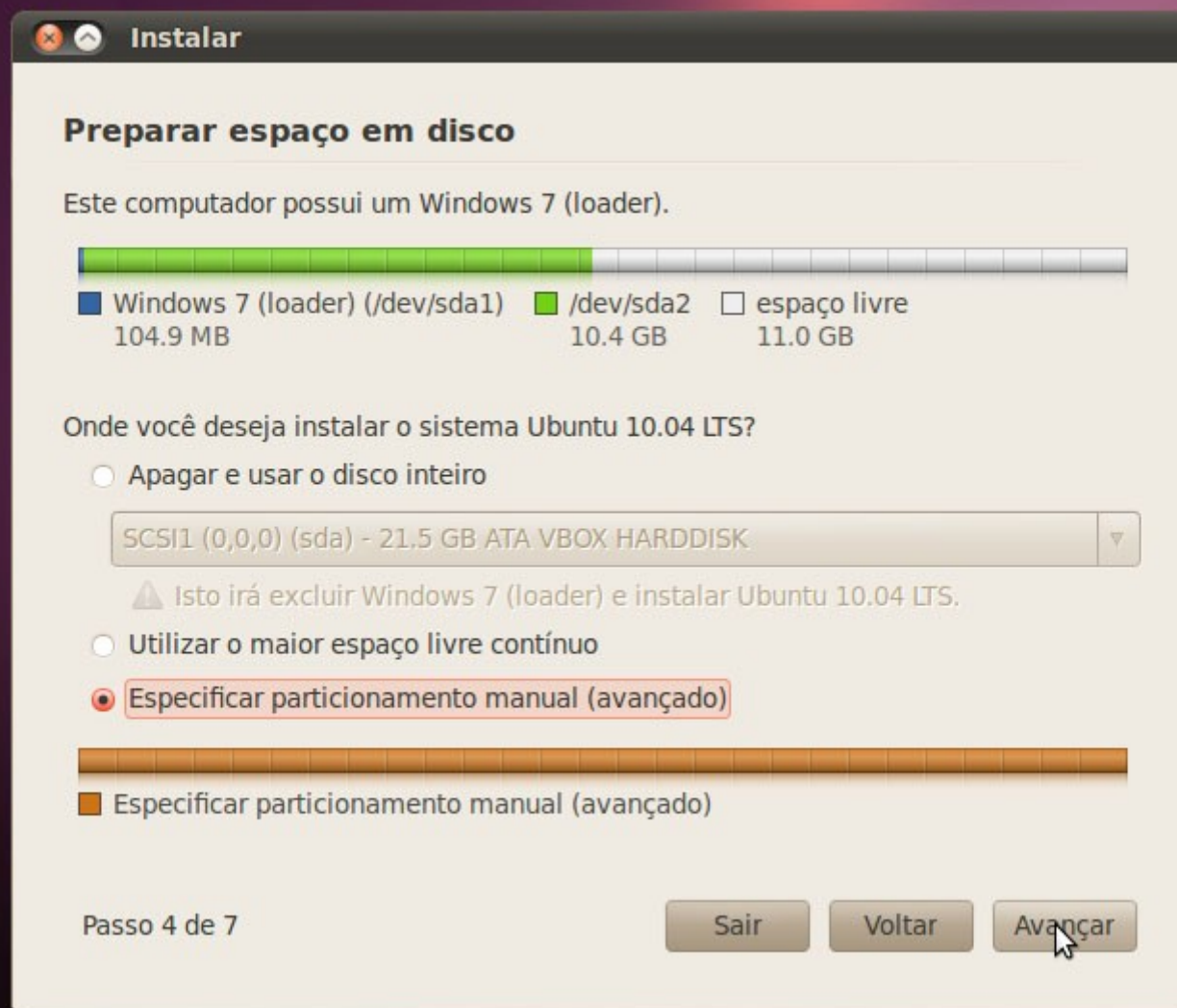
Instalação do Linux - Preparativos e Opções



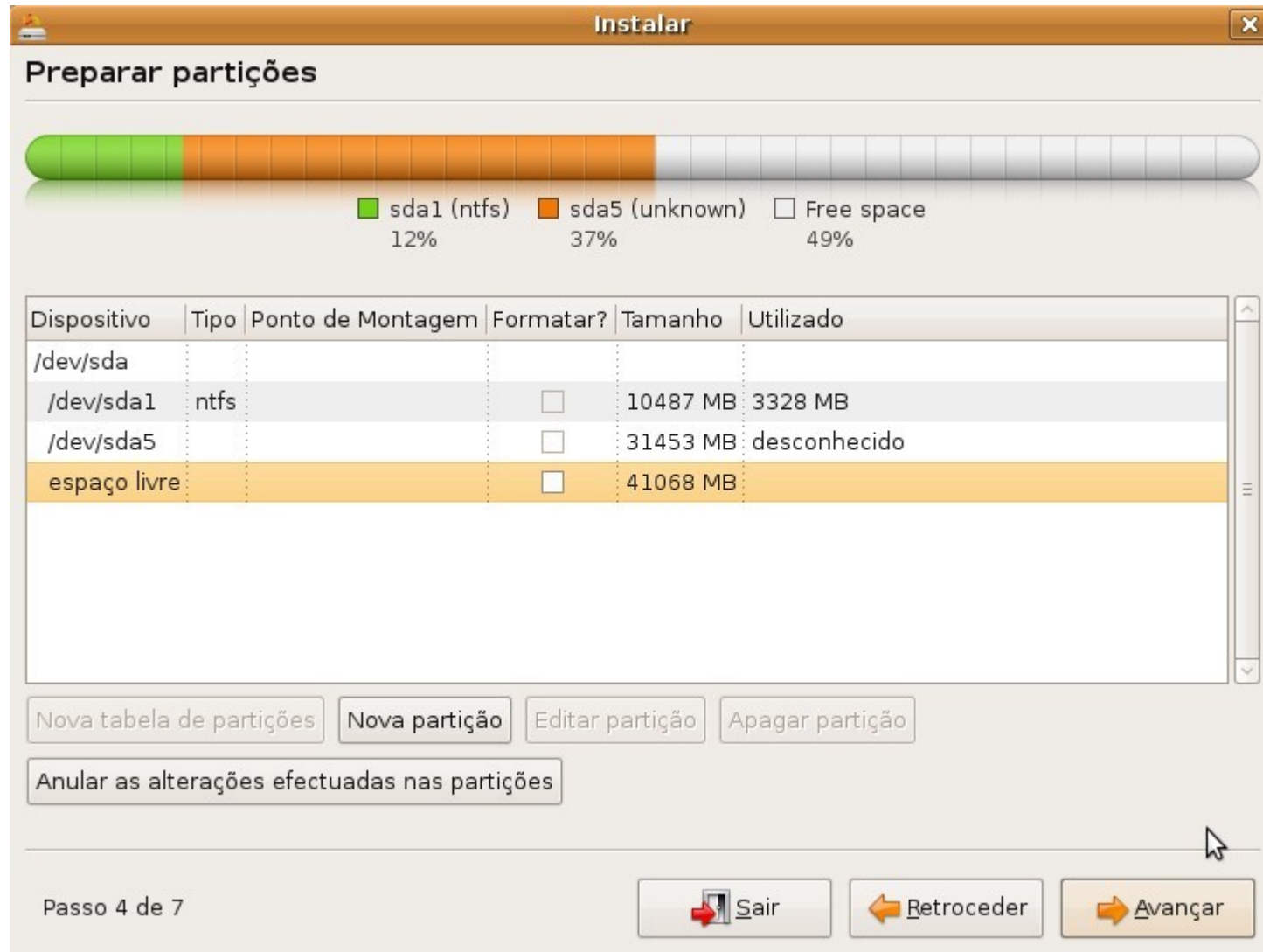
Instalação do Linux - Modos de Instalação



Instalação do Linux - Estado do HD



Instalação do Linux - gparted particionador de HDs



Linux no Desktop

```
GNU GRUB versão 1.99-12ubuntu5

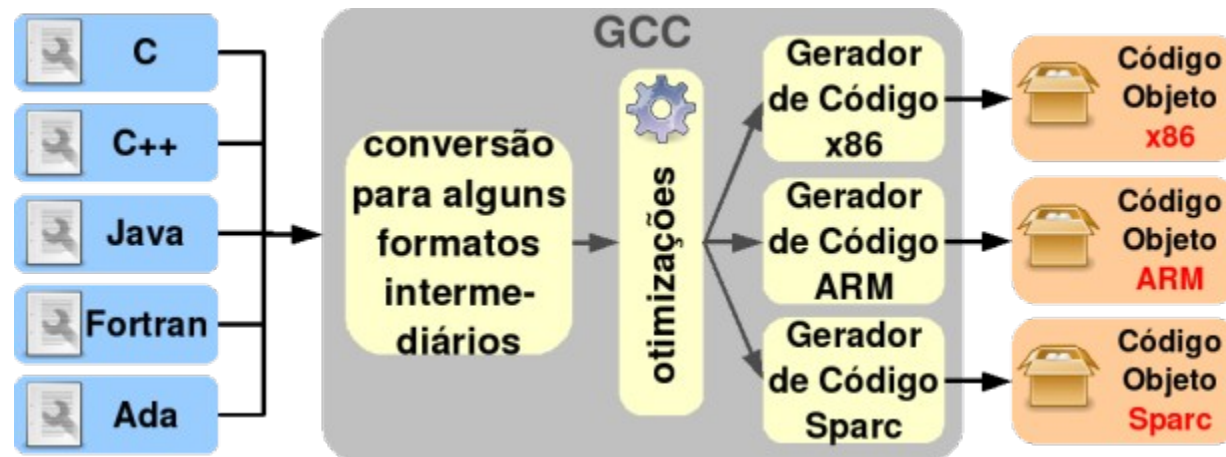
Ubuntu, com Linux 3.0.0-12-generic
Ubuntu, com Linux 3.0.0-12-generic (modo de recuperação)
Memory test (memtest86+)
Memory test (memtest86+, serial console 115200)
Linux Mint 12 32-bit, 3.0.0-12-generic (/dev/sda1) (on /dev/sda1)
Linux Mint 12 32-bit, 3.0.0-12-generic (/dev/sda1) -- recovery mode (+)

Use as teclas + e + para selecionar qual a entrada está
destacada.
Pressione enter para iniciar o OS selecionado, 'e' para editar
os comandos antes da inicialização or 'c' para linha de
```

Introdução ao GCC

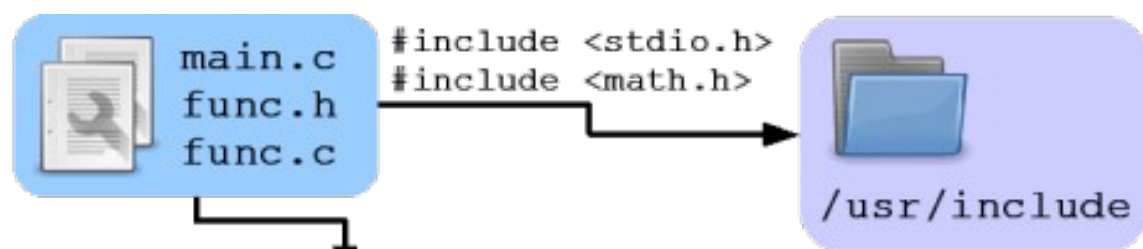
O que é o GCC

- GNU Compiler Collection



Visão Geral do Desenvolvimento com GCC

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



Fontes e APIs

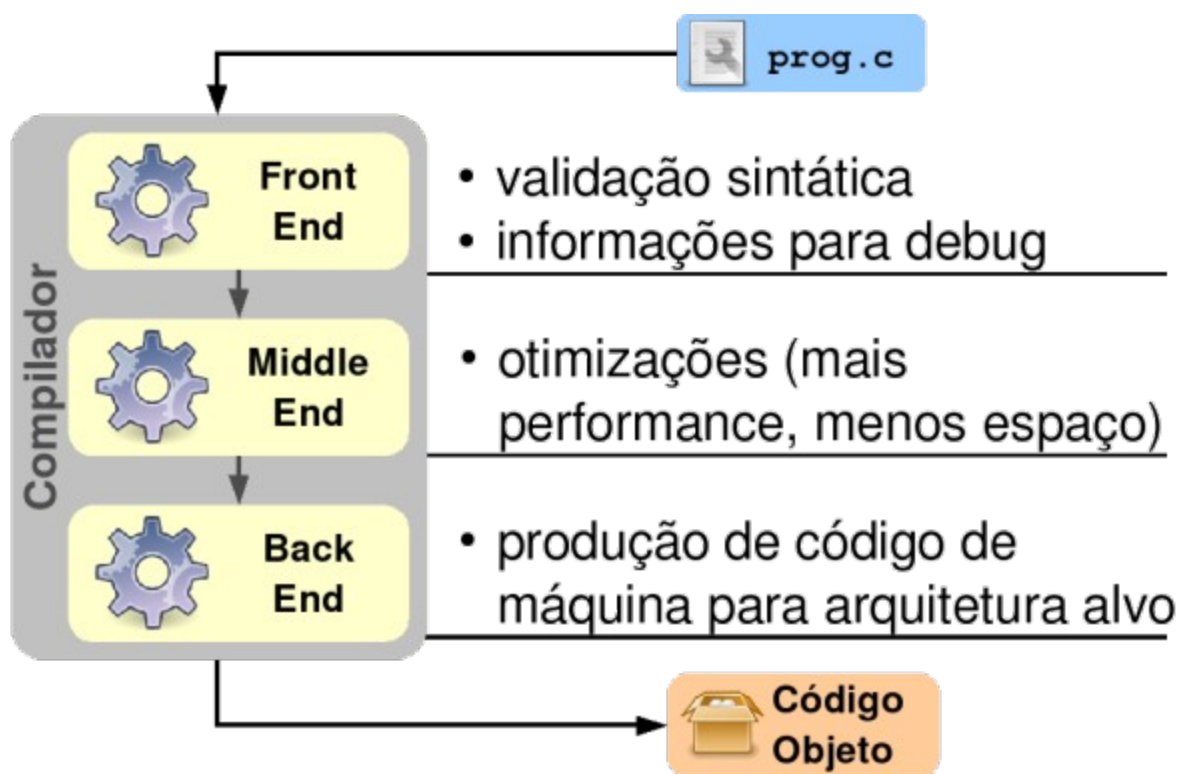


Arquivos objeto e bibliotecas



Programa executável

Estrutura de um Copilador do GCC



```
while (a!=EOF)
    i++;
```

```
a = 3
print a + 5 ↔ print 8
```

Compilação

Passo a passo

Compilação Passo a Passo

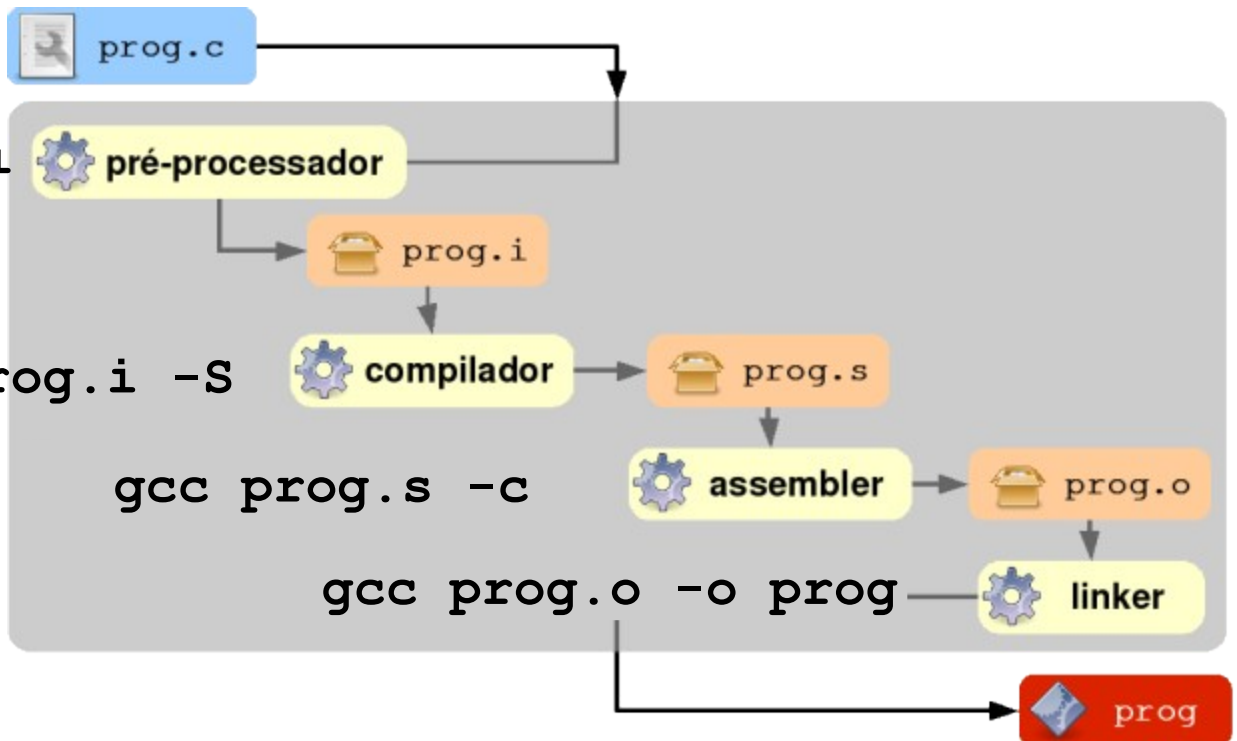
```
#include <stdio.h>
main(void)
{
    printf("Hello World!\n");
}
```

`gcc prog.c -E -o prog.i`

`gcc prog.i -S`

`gcc prog.s -c`

`gcc prog.o -o prog`



Hello World Assembly

```
.file    "hello.c"
.section          .rodata
.LC0:
.string "Hello World!"
.text
.globl main
.type    main, @function
main:
    pushl    %ebp
    movl     %esp, %ebp
    subl     $8, %esp
    andl     $-16, %esp
    movl     $0, %eax
    addl     $15, %eax
    addl     $15, %eax
    shrl     $4, %eax
    sall     $4, %eax
    subl     %eax, %esp
    movl     $.LC0, (%esp)
    call     puts
    movl     $0, %eax
    leave
    ret
    .size    main, .-main
    .ident   "GCC: (GNU) 4.0.3 (Ubuntu 4.0.3-
1ubuntu5)"
    .section          .note.GNU-stack,"",@progbits
```

Opções de Parada da Compilação

| Parâmetro do GCC | Para após | Saída |
|------------------|-------------------|----------------------------|
| -E | pré-processamento | código pré-processado (.i) |
| -S | compilação | código assembly AT&T (.s) |
| -c | assembler | código objeto (.o) |

Exercício 1 - Etapas da Compilação

1. Baixe o arquivo UD2.tar.gz
2. Descompacte-o em sua área de trabalho
3. Entre no diretório EX1
4. A cada comando digitado use o comando `ls -la`
5. `gcc hello.c -E -o hello.pre`
6. `gcc hello.c -S`
7. `gcc hello.c -c`
8. `gcc hello.o -o hello1`
9. `gcc hello.c -o hello2`

Forçando a ligação de bibliotecas

Forçando a ligação de bibliotecas

```
#include <stdio.h>
#include <math.h>
main(int argc, char *argv[])
{
    double valor = 37.0;
    printf("O seno de %3.2f eh %3.2f\n", valor, sin(valor));
}
```

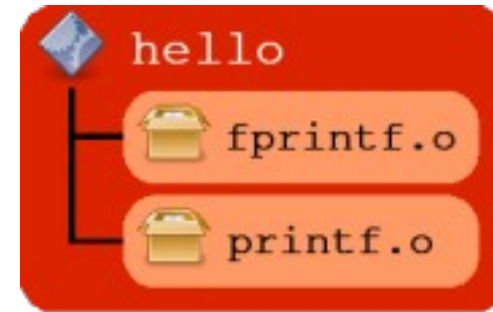
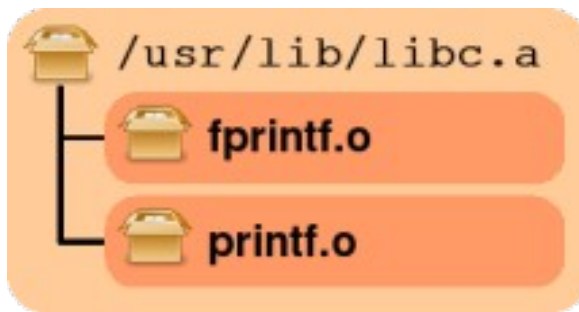
> gcc main.c -o contador

/tmp/cc8MZpoP.o: In function `main':main.c:(.text+0x5a): undefined
reference to `sin'
collect2: ld returned 1 exit status

> gcc main.c -o contador -lm

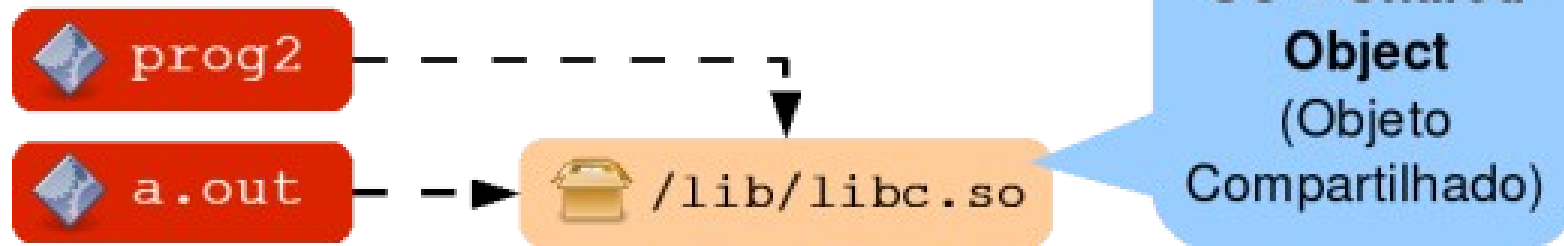
Forçando a ligação de bibliotecas

- Ligação estática



```
> gcc hello.c -o hello /MyLibPath/libc.a  
> gcc hello.c -o hello --static -L/MyLibPath/ -lc  
490K
```

- Ligação dinâmica



```
> gcc hello.c -o hello /MyLibPath/libc.so  
> gcc hello.c -o hello -L/MyLibPath/ -lc  
6,8K
```

necessita mudança da variável de ambiente `LD_LIBRARY_PATH` ou
linux.die.net/man/8/ldconfig

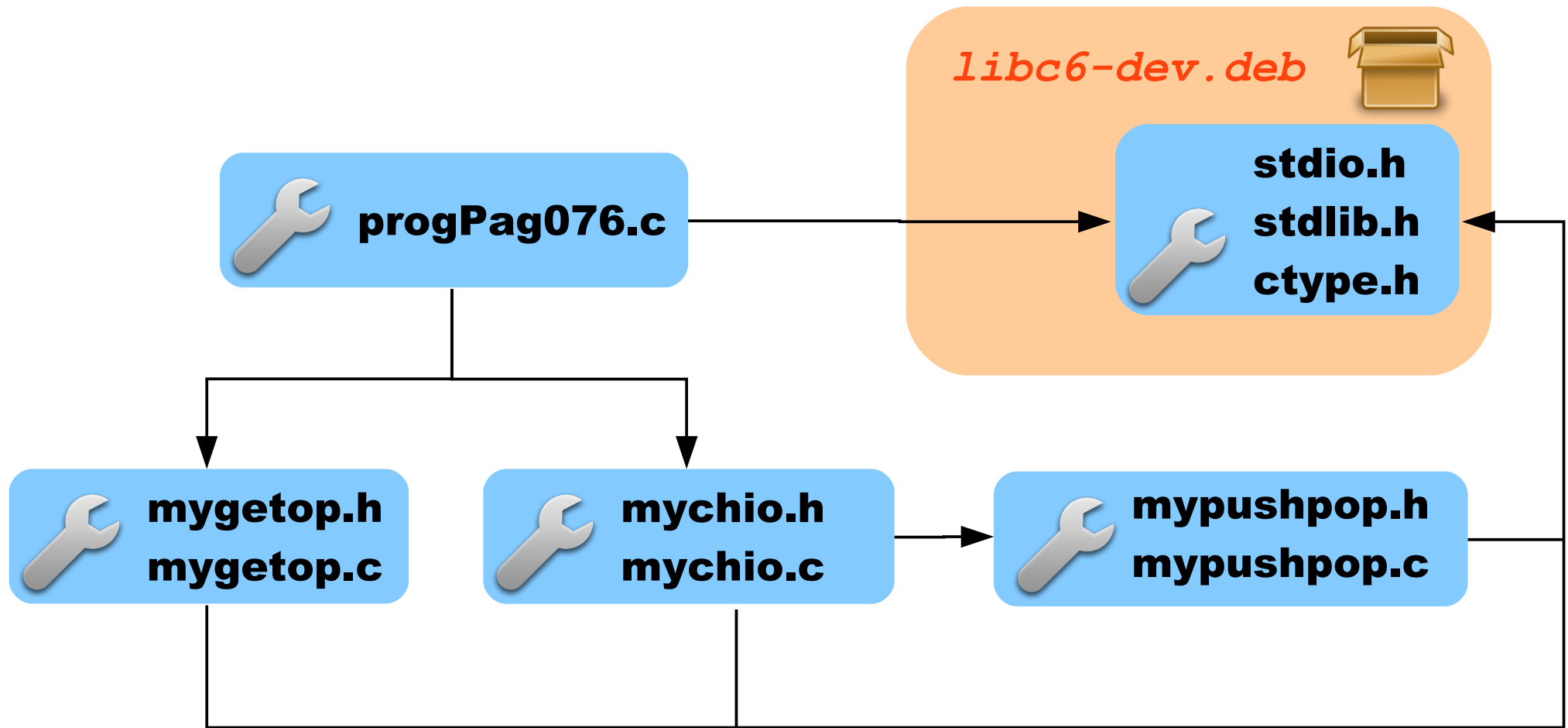
Exercício 2 - Forçando a linkagem

1. Entre no diretório EX2
2. Após digitar os comando a seguir use o comando `ls -la` e compare o tamanho dos executáveis.
3. `gcc seno.c -o seno`
4. `gcc seno.c -o senoDin1 -lm`
5. `gcc seno.c -o senoDin2 /usr/lib/x86_64-linux-gnu/libm.so`
6. `gcc seno.c -lm --static -o senoStat`
7. `gcc seno.c /usr/lib/x86_64-linux-gnu/libm.a -o senoStaDin`

Compilando Múltiplos Arquivos

Compilando Vários Arquivos Fonte de um Programa

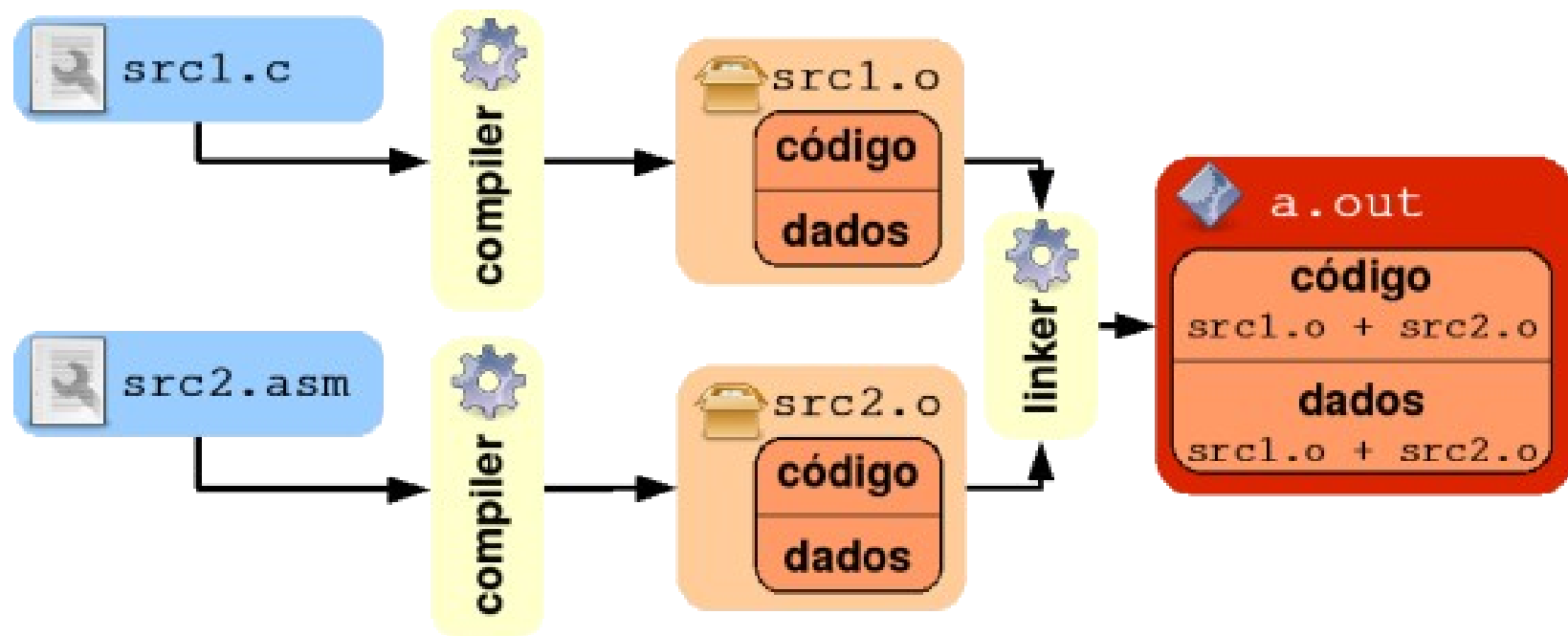
- Múltiplos arquivos fonte e compilação num só passo



```
> gcc mygetop.c mychio.c mypushpop.c progPag076.c -o progPag076
```

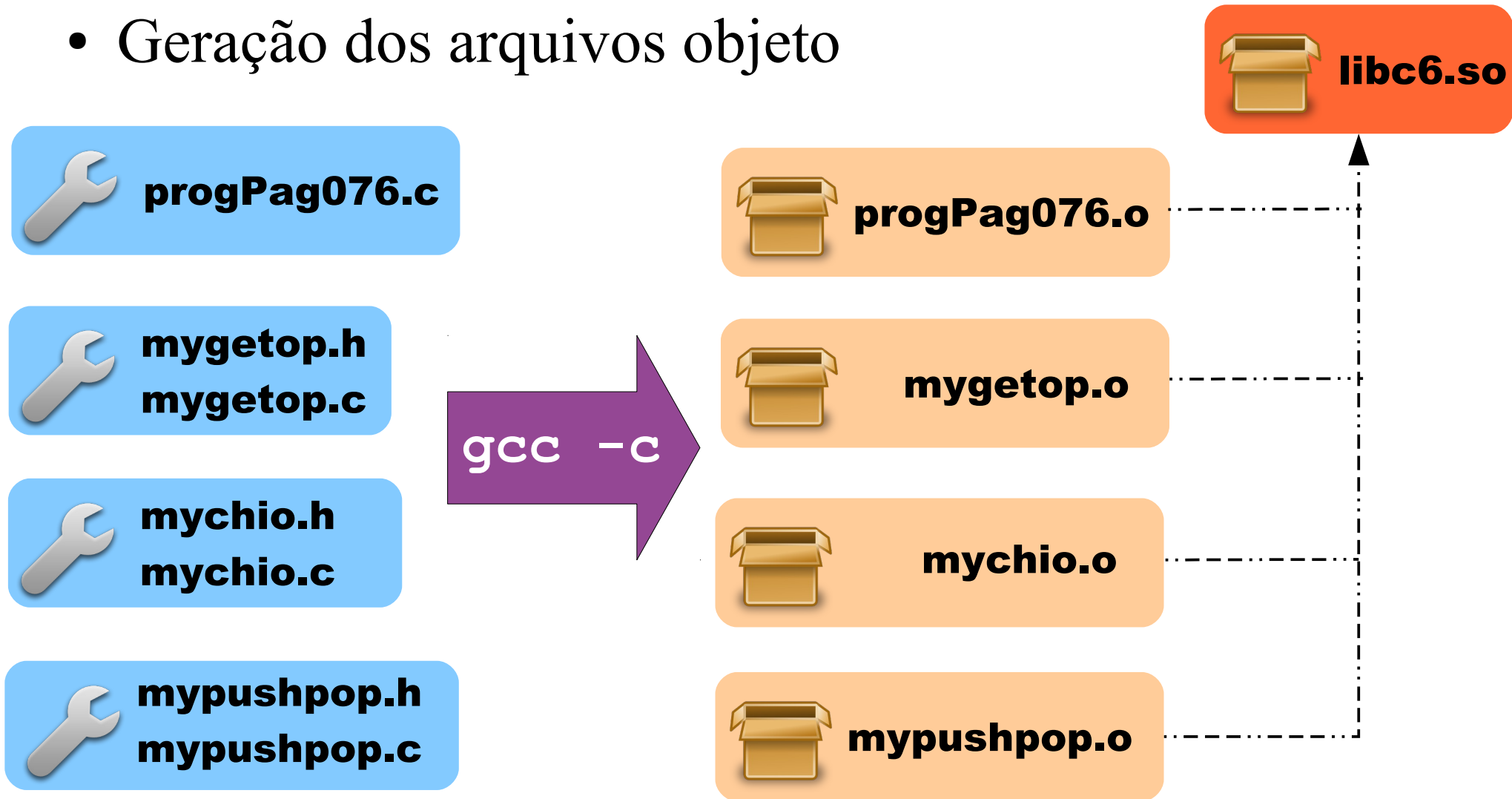
Compilando Vários Arquivos Fonte de um Programa

- Visão geral



Compilando Vários Arquivos Fonte de um Programa

- Geração dos arquivos objeto



```
> gcc mygetop.c mychio.c mypushpop.c progPag076.c -c
```

Compilando Vários Arquivos Fonte de um Programa

- Arquivos objeto e bibliotecas compartilhadas



progPag076.o

**main()
push()
pop()
printf()
getop()
atof()**

libc6-dev.deb



libc6.so

**printf()
atof()
getchar()
isdigit()**



mygetop.o

**getop()
isdigit()
getch()
ungetch()**



mypushpop.o

**push()
pop()
printf()**



mychio.o

**getch()
ungetch()
getchar()
printf()**

Compilando Vários Arquivos Fonte de um Programa

- Linking



progPag076.o



mygetop.o



mychio.o



mypushpop.o



gcc linker



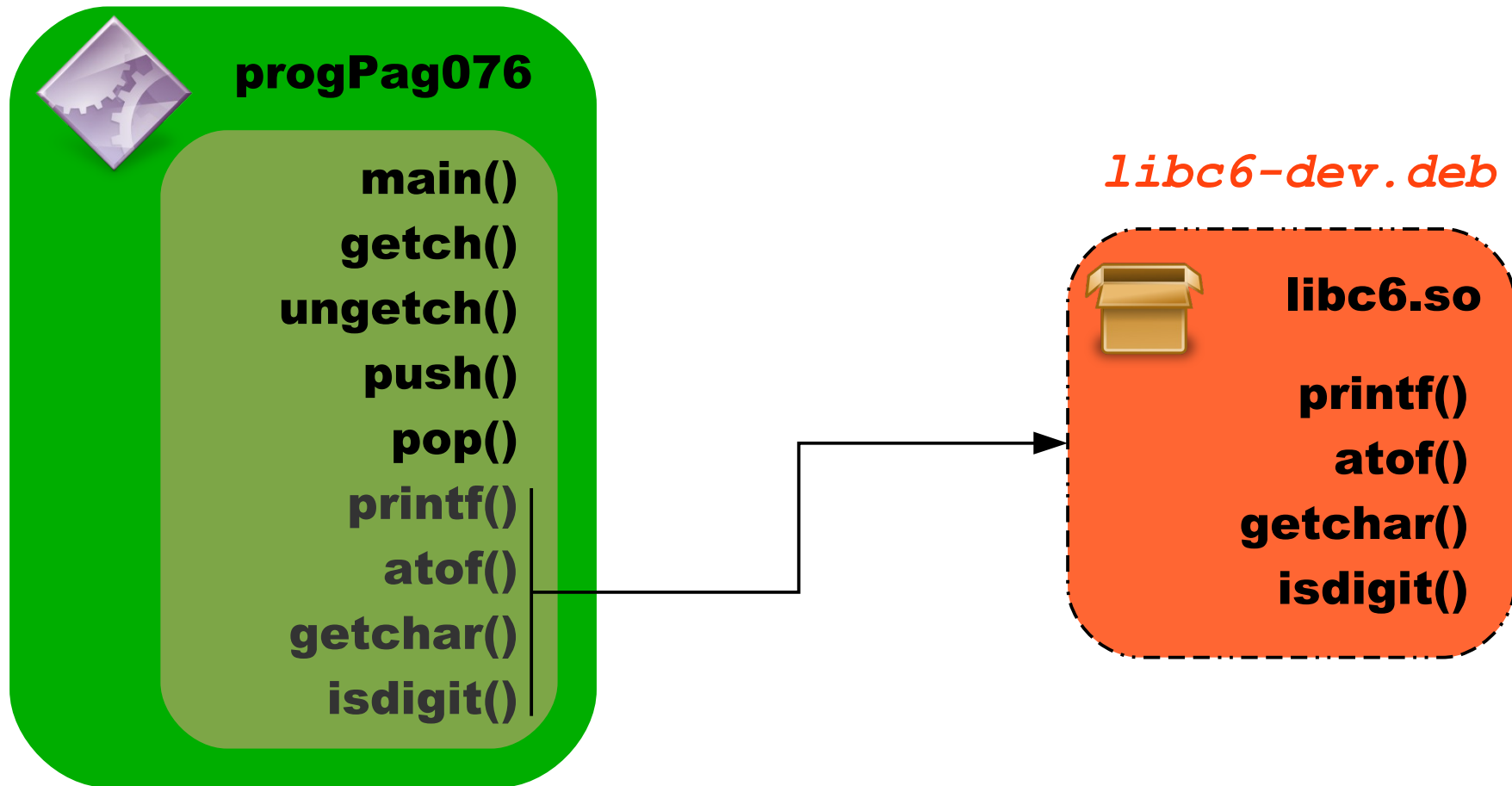
progPag076

```
main()
getch()
ungetch()
push()
pop()
printf()
atof()
getchar()
isdigit()
```

```
> gcc mygetop.o mychio.o mypushpop.o progPag076.o -o progPag076
```


Compilando Vários Arquivos Fonte de um Programa

- Execução e ligação dinâmica



> ./progPag076

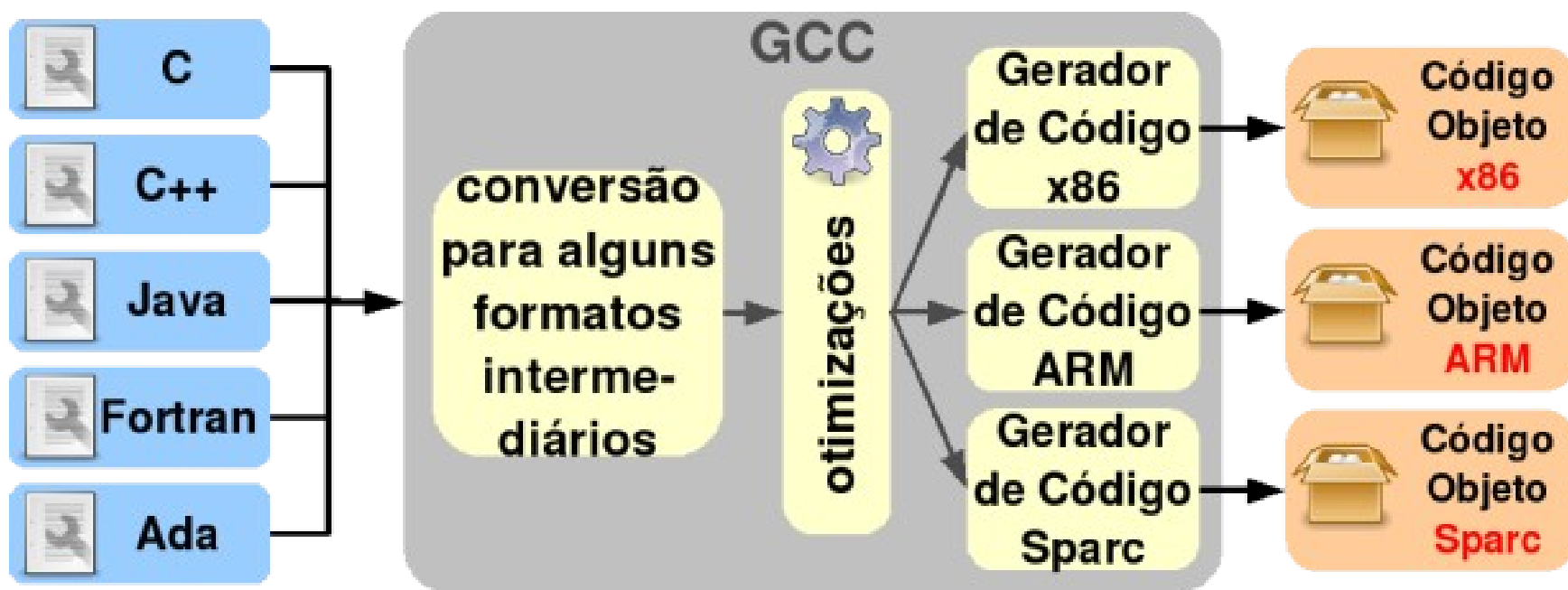
Exercício 3 - Compilando Múltiplos Arquivos

1. Entre no diretório EX3
2. Após digitar os comando a seguir use o comando `ls -la` e analise os arquivos gerados
3. `gcc mygetop.c -c`
4. `gcc mychio.c -c`
5. `gcc mypushpop.c -c`
6. `gcc progPag076.c -c`
7. `gcc mygetop.o mychio.o mypushpop.o
progPag076.o -o progPag076`

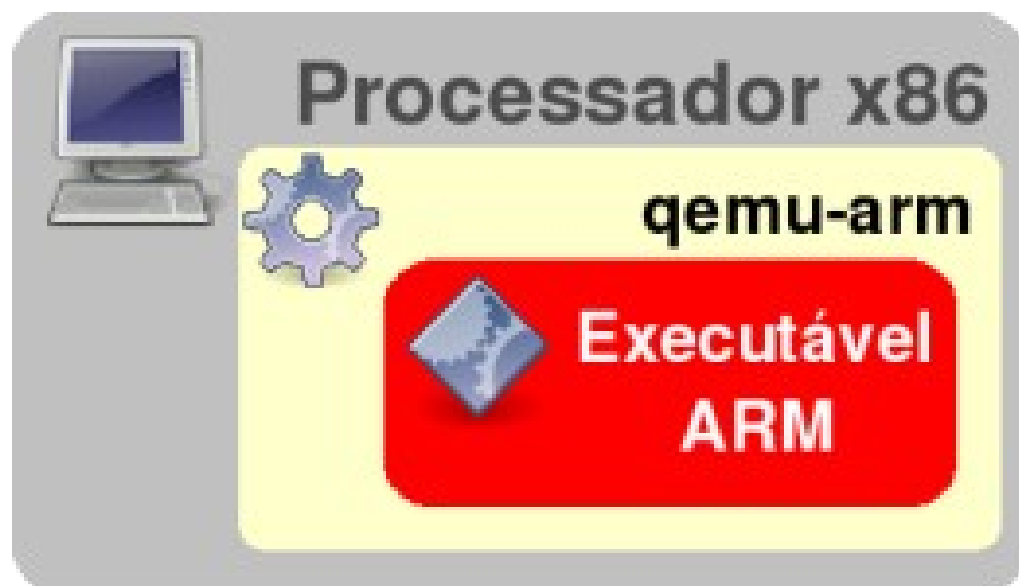
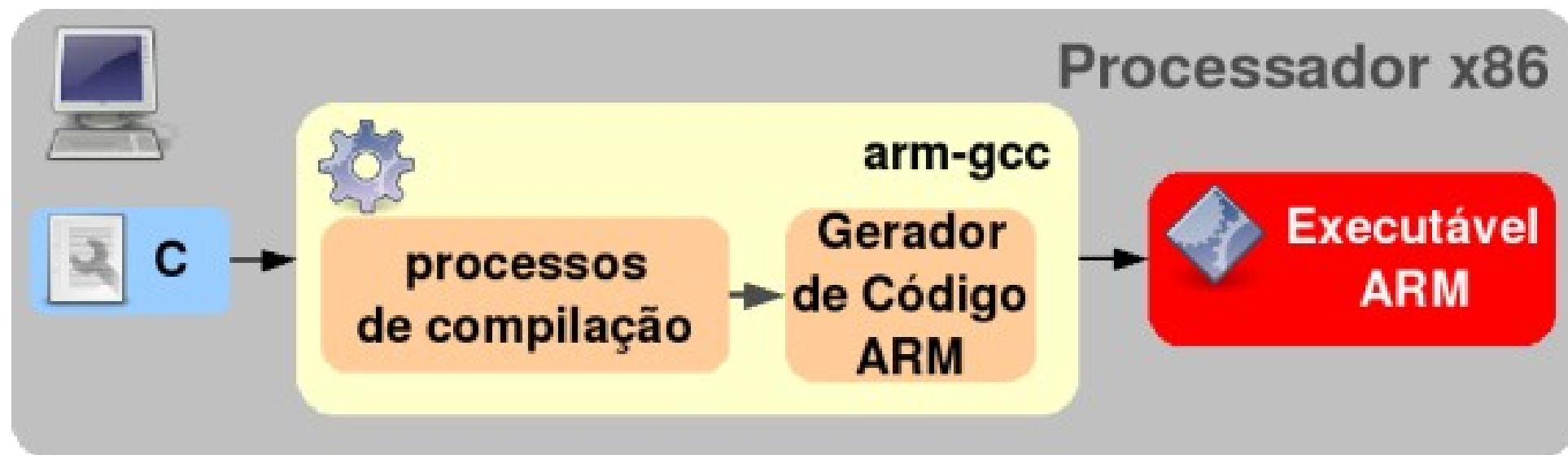
Compilação Cruzada

Compilação Cruzada

- Plataformas alvo



Compilação Cruzada



Bibliotecas pré-compiladas

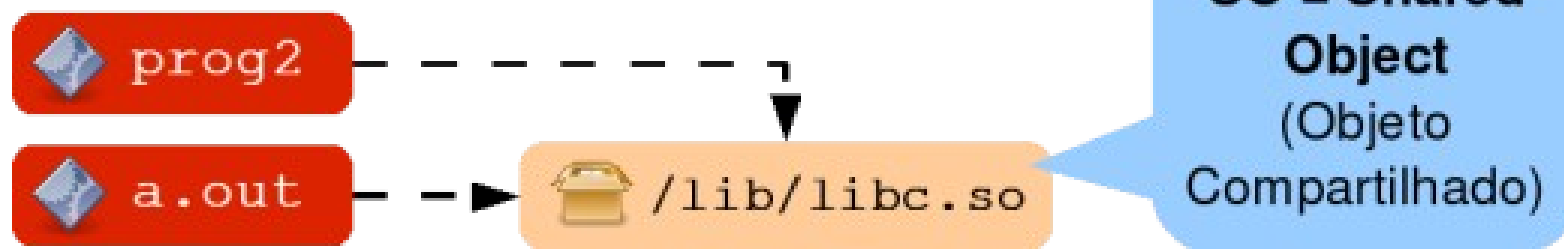
Usando Bibliotecas Pré-compiladas

- Ligação estática



```
> gcc hello.c -o hello --static /usr/lib/x86_64-linux-gnu/libc.a  
490K
```

- Ligação dinâmica



```
> gcc hello.c -o hello /usr/lib/x86_64-linux-gnu/libc.so  
6,8K
```

Criando Bibliotecas Pré-compiladas

- Ligação estática



```
> ar rcs libProgPag076.a mygetop.o mypushpop.o mychio.o
```

- Ligação dinâmica



```
> gcc mychio.c mypushpop.c mygetop.c -shared -fPIC -o libProgPag076.so
```


Examinando Bibliotecas Pré-compiladas

- Ligação estática



libProgPag076.a

mychio.o

mypushpop.o

mygetop.o

```
> objdump -a libProgPag076.a | grep mychio.o
```

- Ligação dinâmica



libProgPag076.so

mychio.c

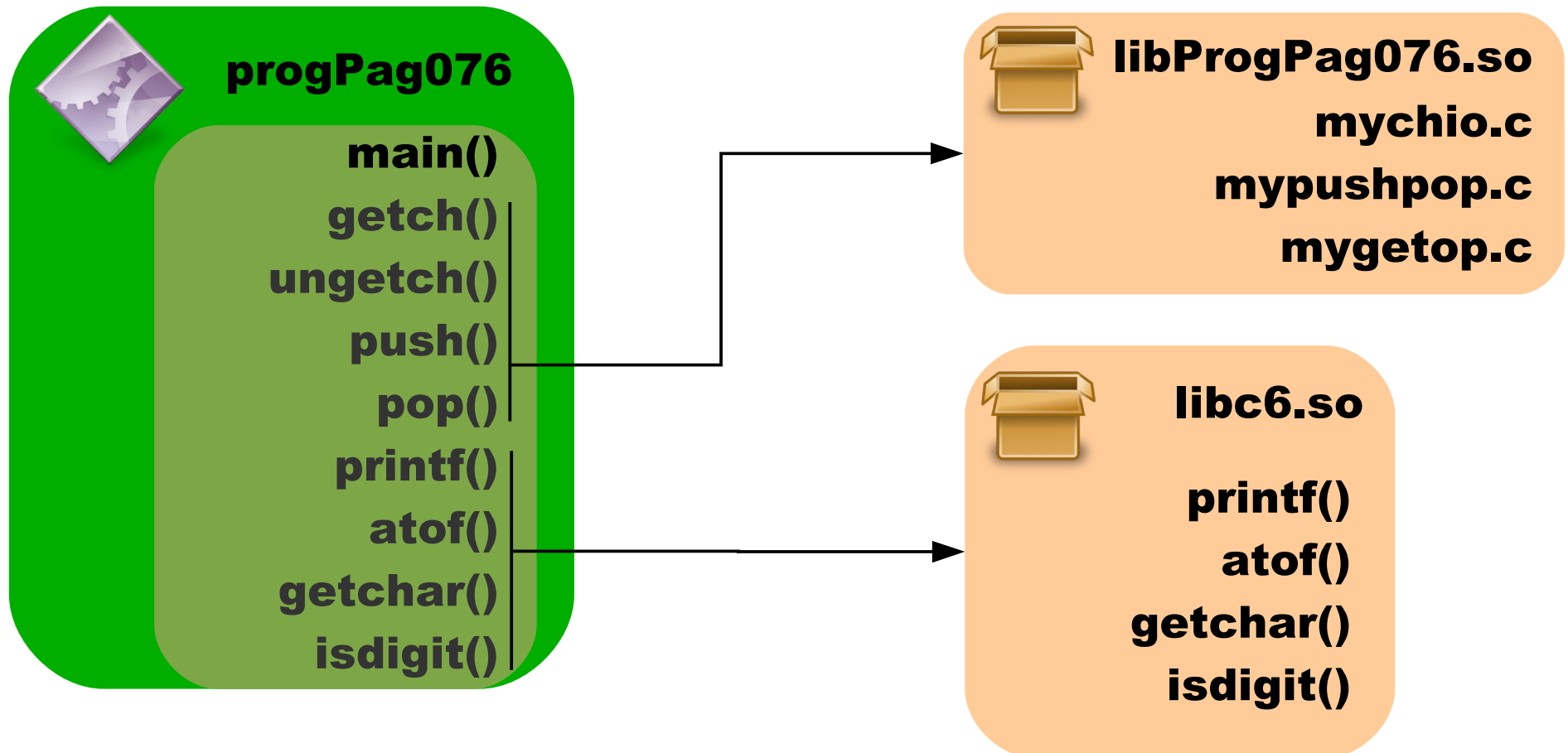
mypushpop.c

mygetop.c

```
> objdump -tT libProgPag076.so
```

Dependências dinâmicas de um executável

- Varrendo o executável



```
> ldd progPag076
./libProgPag076.so (0x00007fbe968d6000)
libc.so.6 => /lib/x86_64-linux-gnu/libc.so.6 (0x00007fbe964f4000)
/lib64/ld-linux-x86-64.so.2 (0x00007fbe96ada000)
```

Exercício 4 - Compilando bibliotecas .so e .a

1. Entre no diretório EX4
2. Após digitar os comando a seguir use o comando `ls -la` e analise os arquivos gerados quanto ao tipo e tamanho
3. `gcc mygetop.c mypushpop.c mychio.c -c`
4. `ar rcs libProgPag076.a mygetop.o
mypushpop.o mychio.o`
5. `gcc mychio.c mypushpop.c mygetop.c
-shared -fPIC -o libProgPag076.so`
6. `gcc progPag076.c libProgPag076.a -o
progPag076Stat --static`
7. `./progPag076Stat`

Exercício 4 - Compilando bibliotecas .so e .a

8. `gcc progPag076.c libProgPag076.so -o progPag076Dina`
9. `./progPag076Dina`
10. `objdump -a libProgPag076.a`
11. `objdump -tT libProgPag076.so`
12. `ldd progPag076Dina`

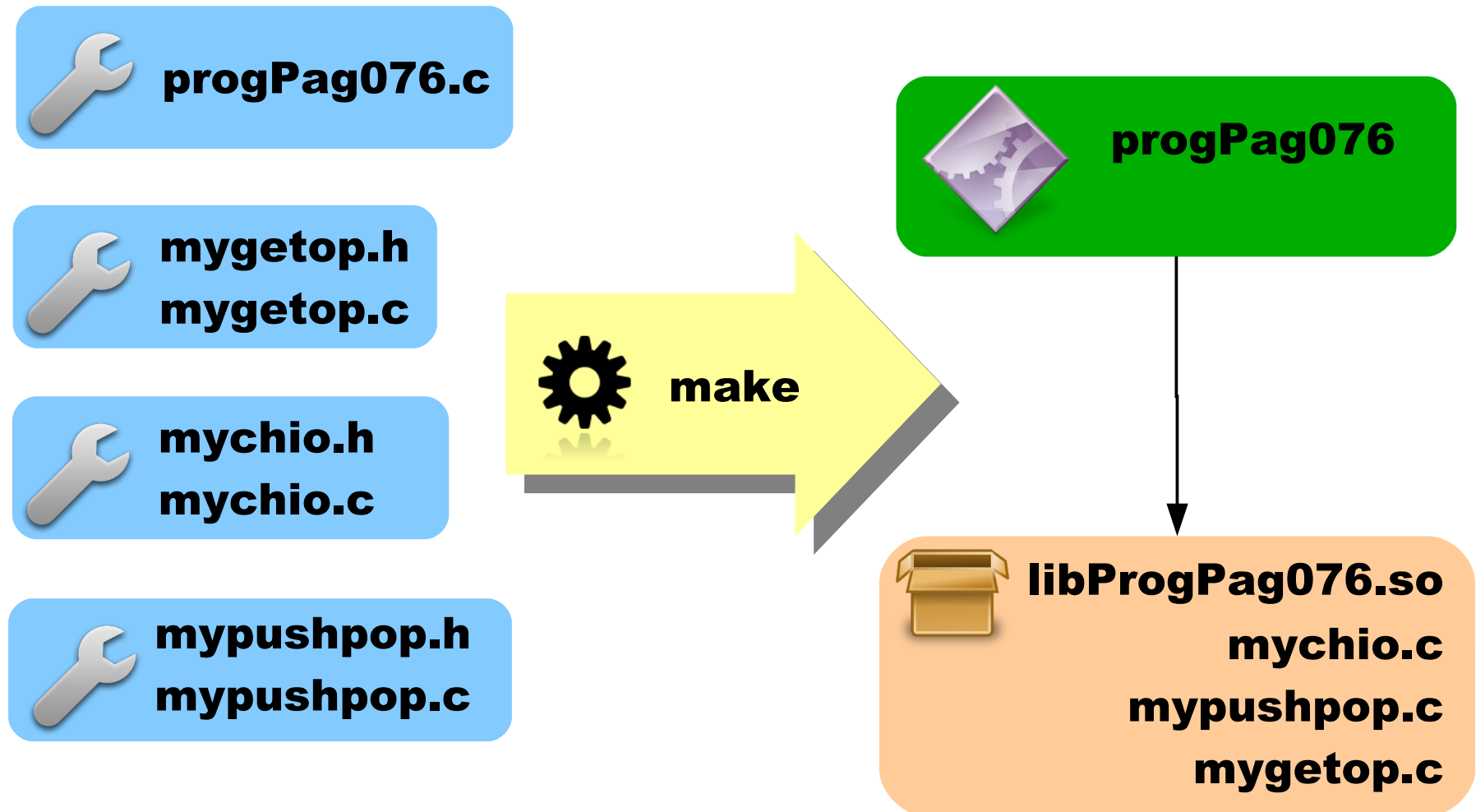
Makefile

Makefile

- Aplicativo make automatiza o processo de construção do software, respeitando a ordem de dependências entre os arquivos.
- O comando make procura no mesmo diretório o arquivo `Makefile` que contém as regras de construção para o código fonte.

```
# Comentários  
VARIABEL=valor  
target: dependencias  
    comando 1  
    comando n
```

Exemplo Makefile



Exemplo Makefile

```
LIBS=./libProgPag076.so
PROG=progPag076

progPag076: progPag076.o libProgPag076.so
    gcc progPag076.o ${LIBS} -o ${PROG}

progPag076.o: progPag076.c
    gcc progPag076.c -c

libProgPag076.so: mychio.h mychio.c mypushpop.c mypushpop.h mygetop.c
mygetop.h
    gcc mychio.c mypushpop.c mygetop.c -shared -fPIC -o libProgPag076.so

clean:
    rm *.o *.so ${PROG}
```



make



progPag076



libProgPag076.so

Exercício 5 - Adaptando um Makefile

1. Entre no diretório EX5
2. Crie a seção `libProgPag076.a` que ao ser evocada gere uma versão estática da biblioteca `libProgPag076`
3. Crie a seção `progPag076Stat` que ao evocada gere uma versão do executável original chamada `progPag076Stat`

OBS.: A linkagem deve ser estática e feita entre `libProgPag076.a` e `progPag076.o`

4. Modifique a seção `clean` para que sejam removidos também os arquivos `.a`
5. Após completar a tarefa use o comando `ls -la` e analise os resultados

Debugger

Debugger

- Compilando modo debugger

```
> gcc mygetop.c mychio.c mypushpop.c progPag076.c -o progPag076 -ggdb
```

- Executando o debugger

```
> gdb progPag076
```

- Criando um breakpoint

```
> (gdb) break progPag076.c:13
```

- Iniciando a execução

```
> (gdb) run
```

Debugger

- Exibindo o conteúdo de uma variável

```
> (gdb) print type  
> (gdb) printf "%s", s
```

- Continuando

```
> (gdb) cont
```

- Saindo

```
> (gdb) quit
```

Exercício 6 - Debugando com gdb

1. Entre no diretório EX6
2. Compile o programa seno com opção de debug
3. Rode o programa através do gdb
4. Crie um breakpoint na linha 7
5. Verifique os valores impressos a cada passo do laço

Fim