

Noções de sistemas de computação

Software

Marco A L Barbosa
malbarbo.pro.br

Departamento de Informática
Universidade Estadual de Maringá



Este trabalho está licenciado com uma Licença Creative Commons - Atribuição-Compartilhual 4.0 Internacional.

Conteúdo

Introdução

Software de sistema

Softwares aplicativo

Desenvolvimento de software

Atividades

Introdução

- Coleção de dados e instruções que dizem o que o computador (hardware) deve fazer

- Todo o software é distribuído sobre uma licença de uso
- Proprietários
 - Foco nas restrições
 - Geralmente tem a execução restrita, não podem ser redistribuídos, modificados, etc
 - Suporte ao usuário geralmente fornecido pela empresa que vende/distribui o software
- Livres
 - Foco nas liberdades
 - Os usuários são livres para executar, redistribuir, modificar, estudar o código, etc
 - Suporte ao usuário geralmente fornecido pela comunidade de usuários e desenvolvedores

Tipos de software

- Software de sistema
- Software aplicativos



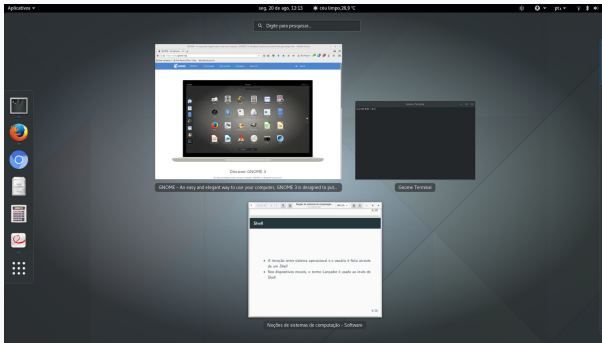
Camadas de um sistema computacional
Por Golftheman [CC BY-SA 3.0] from Wikimedia Commons

Software de sistema

- Sistema operacional
 - Gerencia os recursos do sistema
 - Provê serviços comuns para outros softwares
 - Exemplos
 - Linux (Android, Debian GNU/Linux, Ubuntu, etc)
 - MacOS (Ios)
 - Windows
- Utilitários
 - Ajudam a analisar, configurar e otimizar o sistema
 - Exemplos
 - Antivírus
 - Gerenciadores de pacotes
 - Desfragmentadores de disco

- A interação entre sistema operacional e o usuário é feita através de um *Shell*
- Nos dispositivos moveis, o termo Lançador é usado ao invés de Shell

- Gráficos
 - Manipulação de janelas
 - Geralmente utilizam a metáfora de área de trabalho
 - Facilidade de uso e aprendizado
 - Menos poderosos



- Textuais
 - Comandos descritos por texto
 - Mais difícil de aprender e usar
 - Mais agilidade
 - Mais poderoso

```
malbarbo@t430:/home/malbarbo
malbarbo ~$ fortune
fortune: command not found
malbarbo ~$ 127 sudo apt-get install -q fortune-mod
Lendo listas de pacotes...
Construindo árvore de dependências...
Lendo informação de estado...
Pacotes sugeridos:
  fortune
Os NOVOS pacotes a seguir serão instalados:
  fortune-mod
0 pacotes atualizados, 1 pacotes novos instalados, 0 a serem removidos e 0 não atualizados.
É preciso baixar 0 B/49,5 kB de arquivos.
Depois desta operação, 113 kB adicionais de espaço em disco serão usados.
A seleccionar pacote anteriormente não seleccionado fortune-mod.
(Lendo banco de dados ... 288931 ficheiros e directórios actualmente instalados.)
A preparar para descompactar .../fortune-mod_1%3al.99.1-7+b1_amd64.deb ...
A descompactar fortune-mod (1:1.99.1-7+b1) ...
Configurando fortune-mod (1:1.99.1-7+b1) ...
A processar 'triggers' para man-db (2.7.6-1-2) ...
malbarbo ~$ fortune
Make a wish, it might come true.
malbarbo ~$ |
```

```
$ python
The program 'python' is not installed. Install it by executing:
pkg install python
$ apt-get install -qq python > /dev/null
$ python
Python 3.6.6 (default, Jul 21 2018, 02:41:03)
[GCC 4.2.1 Compatible Android (4691093 based on r316199) Cl
lang 6.0.2 (https://a on linux
Type "help", "copyright", "credits" or "license" for more i
information.
>>> print("Olá mundo!")
Olá mundo!
>>> |
```

ESC / HOME END PGUP
TAB CTRL ALT _ | PGDN

1 2 3 4 5 6 7 8 9 0
q w e r t y u i o p
@ # \$ % ^ & * () _ + =
a s d f g h j k l ; ' : < > [] \ / | ~
↑ ↓ ← →
?123 , .

Softwares aplicativo

Software aplicativo (ou apenas aplicativo ou ainda app)

- Realiza um conjunto de funções e tarefas que atendem as necessidades do usuário
- Exemplos
 - Editores de texto
 - Planilhas eletrônicas
 - Navegadores da internet

Desenvolvimento de software

- Em geral os programas são descritos de forma textual usando uma linguagem de programação
- Uma linguagem de programação pode ser
 - Compilada
 - C/C++
 - Java
 - Pascal
 - Interpretada
 - Python
 - Javascript
 - Lua

Compilação

- Um compilador transforma o código (texto) do programa em um arquivo diretamente executável pelo processador
- Execução eficiente
- Ciclo de desenvolvimento lento (tem que esperar o compilador)
 - Código do programa

```
#include <stdio.h>
int main() {
    printf("Olá mundo!\n");
}
```

- Compilação seguida da execução

```
$ cc -o ola ola.c
$ ./ola
```


Interpretação pura

- Um interpretador lê o código do programa e executa diretamente as instruções descritas nele
- Execução menos eficiente
- Ciclo de desenvolvimento mais rápido (não precisa esperar o compilador)

- Código do programa

```
print("Olá mundo!")
```

- Execução direta pelo interpretador

```
$ python ola.py  
Olá mundo
```

Atividades

1. Identifique o sistema operacional do seu computador ou *smartphone*. Inclua a versão, o desenvolvedor e o tipo de licença.
2. Liste os softwares livres que você utiliza.
3. Experimente um shell textual no seu computador ou *smartphone* e relate como foi a sua experiência.
4. Experimente um shell gráfico alternativo no seu computador ou *smartphone* e relate como foi a sua experiência.

Atividades

5. Experimente algum software utilitário no seu computador ou *smartphone* e relate como foi a sua experiência. Inclua a identificação e o propósito do software.
6. Selecione três softwares que você utiliza e pesquise em quais linguagens de programação eles foram escritos.
7. Selecione três softwares proprietários que você utiliza e pesquise alternativas em software livre.
8. Experimente o interpretador interativo do Python disponível em <https://www.python.org/shell/>. Digite algumas expressões aritméticas seguida de enter. Digite `print("")` com algum texto entre as aspas. O que acontece?