



## *Software* Livre

Prof. Edwar Saliba Jr.  
Março de 2009



# O que é *software* livre?

"*Software* Livre" é uma questão de liberdade, não de preço. Para entender o conceito, você deve pensar em "liberdade de expressão", não em "cerveja grátis".



# Software Livre

- "*Software* livre" se refere à liberdade dos usuários executarem, copiarem, distribuírem, estudarem, modificarem e aperfeiçoarem o *software*.
- Mais precisamente, ele se refere a quatro tipos de liberdade, para os usuários do *software*:
  - Liberdade 1,
  - Liberdade 2,
  - Liberdade 3 e
  - Liberdade 4.



## Liberdade 1

- A liberdade de executar o programa, para qualquer propósito.



## Liberdade 2

- A liberdade de estudar como o programa funciona, e adaptá-lo para as suas necessidades.
- Acesso ao código-fonte é um pré-requisito para esta liberdade.



# Liberdade 3

- A liberdade de redistribuir cópias de modo que você possa ajudar ao seu próximo.



# Liberdade 4

- A liberdade de aperfeiçoar o programa, e liberar os seus aperfeiçoamentos, de modo que toda a comunidade se beneficie;
- Acesso ao código-fonte é um pré-requisito para esta liberdade.



# Software Livre

- Um programa é *software* livre se os usuários têm todas essas liberdades;
- Portanto, você deve ser livre para redistribuir cópias, seja com ou sem modificações, seja de graça ou cobrando uma taxa pela distribuição, para qualquer um em qualquer lugar.





# Software Livre

- Para que a liberdade de fazer modificações, e de publicar versões aperfeiçoadas do *software*, tenha algum significado, deve-se ter acesso ao código-fonte do programa;
- Portanto, acesso ao código-fonte é uma condição imprescindível ao *software* livre.



# Liberdade Irrevogável

- Para que essas liberdades sejam reais, elas têm que ser irrevogáveis desde que você não faça nada errado;
- Caso o desenvolvedor do *software* tenha o poder de revogar a licença, mesmo que você não tenha dado motivo, **o *software* não é livre.**



# *Software* Comercial

- "*Software* Livre" Não significa "não-comercial";
- Um programa livre deve estar disponível para uso comercial, desenvolvimento comercial, e distribuição comercial.



# Algumas Regras Aceitáveis

- Regras sobre como empacotar uma versão modificada são aceitáveis, se elas não bloquearem a sua liberdade de liberar versões modificadas;
- Regras como "se você tornou o programa disponível deste modo, você também tem que torná-lo disponível deste outro modo", também podem ser aceitas.



# Regras Internacionais

- Às vezes regras de controle de exportação e sanções de comércio, podem limitar a sua liberdade de distribuir cópias de programas internacionalmente.



# Precaução

- Quando falando sobre o *software* livre, é melhor evitar o uso de termos como "dado" ou "de graça", porque estes termos implicam que a questão é de preço, não de liberdade;
- Alguns termos comuns como "pirataria" engloba opiniões que nós esperamos que você não endossará.



# Considerações Finais

- Finalmente, note que critérios como os estabelecidos nesta definição do *software* livre requerem cuidadosa deliberação quanto à sua interpretação;
- Para decidir se uma licença se qualifica como de *software* livre, nós a julgamos baseados nestes critérios para determinar se ela segue este espírito assim como as palavras exatas.



# Considerações Finais

- Quando nós chegamos a uma conclusão sobre uma nova questão, nós frequentemente atualizamos estes critérios para tornar mais fácil determinar porque certas licenças se qualificam ou não.





# Fundamentos de Sistemas Operacionais

# GNU



- **Projeto GNU**, em computação, é um projeto iniciado por Richard Stallman em 1984, com o objetivo de criar um sistema operacional totalmente livre.
- Este sistema operacional GNU deveria ser compatível com o sistema operacional UNIX, porém não deveria utilizar-se do código fonte do mesmo. Stallman escolheu o nome **GNU** porque este nome, além do significado original do mamífero Gnu, é um acrônimo recursivo de: **GNU is Not Unix** (em português: GNU não é Unix).
- Em 1991 o sistema operacional já estava quase pronto, mas faltava o principal, que é o *kernel* do sistema operacional. O grupo liderado por Stallman estava desenvolvendo um *kernel* chamado Hurd. Porém, em 1991, um jovem filandês chamado Linus Torvalds havia criado um *kernel* que poderia usar todas as peças do sistema operacional GNU. Este *kernel* ficou conhecido como Linux, contração de Linus e Unix.
- Atualmente, o sistema operacional GNU com o *kernel* Linux é conhecido como **GNU/Linux**, que é como o projeto solicita aos utilizadores que se refiram ao sistema completo, embora a maioria das pessoas se referem ao sistema apenas como Linux por uma questão de comodidade.

(Fonte: <[http://pt.wikipedia.org/wiki/Projeto\\_GNU](http://pt.wikipedia.org/wiki/Projeto_GNU)> Acesso em: 12 mar. 2009)



## *Software* Livre





## Bibliografia

- MACHADO, F. B.; MAIA, L. P. **Arquitetura de Sistemas Operacionais**, 3ª Ed., Rio de Janeiro: LTC Editora, 2002.
- MARTINS, Thiago M. *Slides da disciplina de Sistemas Operacionais de Arquitetura Aberta*. Faculdade Pitágoras, 2009.