

SCC0222 - Laboratório de Introdução à Ciência de Computação I

# Compilação e Saída de Dados

Prof.: Leonardo Tórtoro Pereira

[leonardop@usp.br](mailto:leonardop@usp.br)

O que preciso usar para programar em  
C?

# Como começar a programar em C

- C é uma linguagem de programação de “alto” nível
- Nós escrevemos um código nela (programa) e ele precisa ser lido e entendido pelo computador para que ele possa executar o que você pediu
- Mas o computador não “entende” C
  - ◆ Apenas binário
- Como fazer?

## Como começar a programar em C

- É preciso que um *compilador* leia o programa em C e *compile* essa informação em um código *assembly*
- Esse código *assembly* é um conjunto de instruções que o seu computador consegue processar e transformar nos binários para manipulação dos dados em si.
- Em C costuma-se usar o compilador [gcc](#)
  - ◆ Ou sua versão para Windows contida no [MinGW](#)

## Como começar a programar em C

- Ele transforma seu código C em um arquivo objeto (.o) e, depois, em um arquivo executável (.exe no Windows)
- Usando uma IDE (ambiente de desenvolvimento integrado), como o [Code Blocks](#), ou o [onlinegdb](#) é possível fazer isso através da interface do ambiente
  - ◆ Temos licença de estudante para a [CLion](#) também
- Mas também podemos usar comandos de texto para o computador, especialmente em ambientes Linux

## Como começar a programar em C

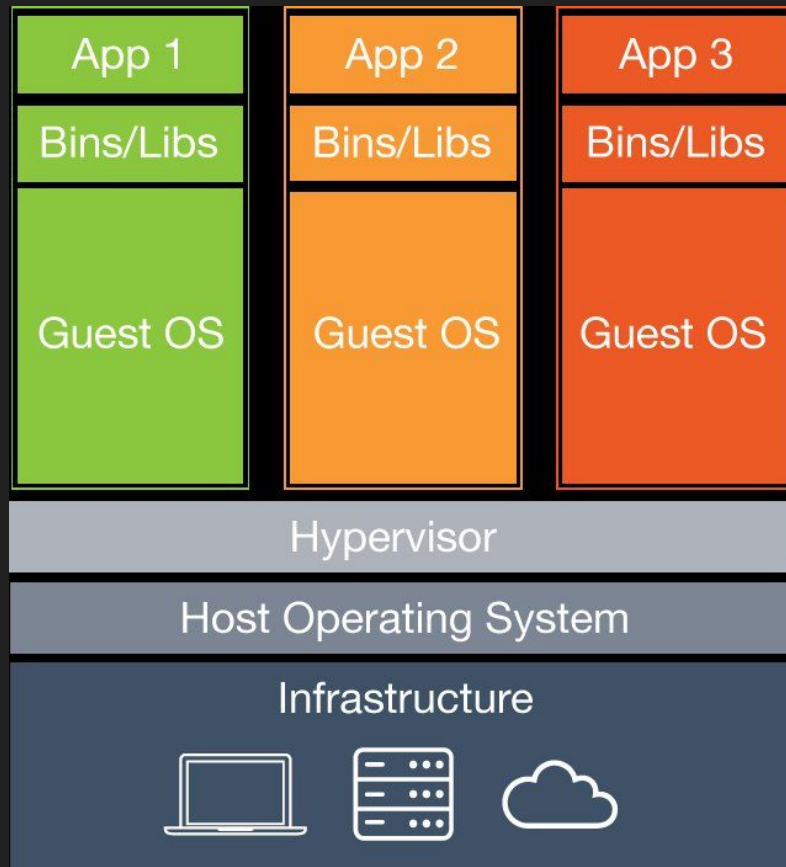
- Vamos ver mais para o fim da aula COMO fazer isso. A pergunta agora é:
  - ◆ “Ouvi falar que precisa ter um Linux para programar direito em C. É verdade?”
- Não exatamente...
- O Linux evita máscarar alguns erros que surgem do uso indevido de memória e ponteiros
- E nele podemos usar o [valgrind](#)

## Como começar a programar em C

- É recomendado usá-lo. Especialmente quando começarmos a manipular ponteiros e realizar alocação dinâmica
- Preciso fazer um *dual boot*?
- Não... você pode usar uma [Máquina Virtual](#)

# Máquina Virtual (VM)





Estrutura de uma VM. Fonte:

[https://www.researchgate.net/publication/317294656\\_A\\_prototype\\_Data\\_Processing\\_Center\\_for\\_LISA/](https://www.researchgate.net/publication/317294656_A_prototype_Data_Processing_Center_for_LISA/)

# Como começar a programar em C

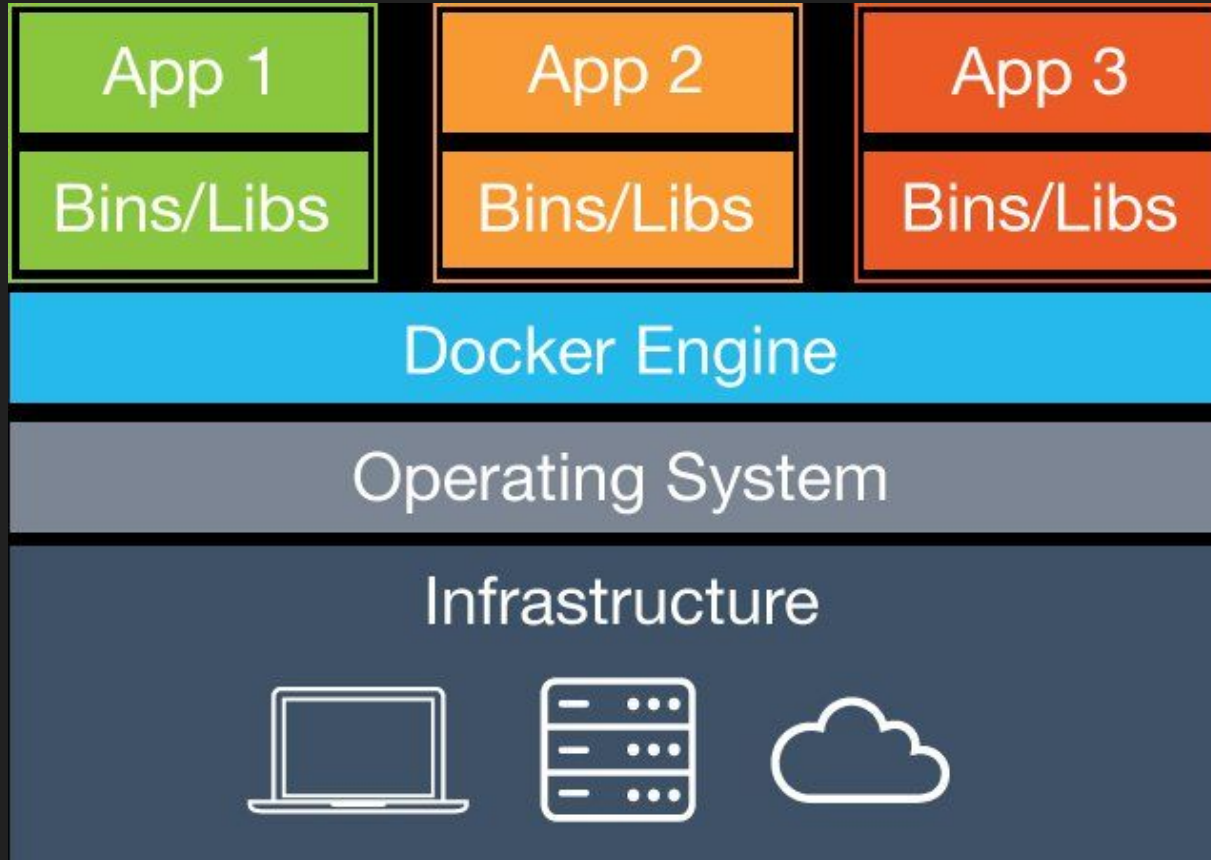
- Mas máquinas virtuais são pesadas!
- Hoje em dia temos 2 soluções MUITO melhores: WSL e Docker
- WSL é bem mais simples e nativa do Windows
- Mas Docker é uma ferramenta poderosa que vale a pena guardar em mente para uso futuro

# Windows Subsystem for Linux (WSL)

# WSL

- “Nova” (2016) tecnologia que permite rodar um SO Linux diretamente do seu Windows
- Otimiza (muito) uma VM de Linux
- <https://docs.microsoft.com/pt-br/windows/wsl/about>
- <https://docs.microsoft.com/pt-br/windows/wsl/install-win10>
- <https://www.techrepublic.com/article/wsl-2-where-is-it-and-where-is-it-going/>

Docker



Uso de Docker. Fonte:

[https://www.researchgate.net/publication/317294656\\_A\\_proto-Data\\_Processing\\_Center\\_for\\_LISA/](https://www.researchgate.net/publication/317294656_A_proto-Data_Processing_Center_for_LISA/)

# Docker

- Versão mais nova é compatível com Windows Home
- Também pode ser usado no Linux sem problemas
- Mais trabalhoso que usar o WSL, mas é uma tendência no mercado para testes de programar multiplataformas

Olá Mundo!



# Olá Mundo!

→ Vamos fazer um programa bem simples para mostrar como compilar e como é C

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    printf("Hello World\n");
```

```
    return 0;
```

```
}
```

# Olá Mundo!

```
#include <stdio.h>

int main()
{
    printf("Hello World\n");
    return 0;
}
```

# Olá Mundo!

- Para compilar em linha de comando:
- Windows:
  - ◆ `gcc -o hello.exe hello.c`
- Linux
  - ◆ `gcc -o hello hello.c`

Dúvidas?

# Bibliografia

- MEDINA, M.; FERTIG, C. Algoritmos e Programação: teoria e prática, Novatec, 2006.
- Ascencio, Ana Fernanda Gomes. Fundamentos da programação de computadores : algoritmos, pascal, C/C++ (padrão ANSI) e JAVA.
- Aguilar, Luis J. Fundamentos de Programação : Algoritmos, Estruturas de Dados e Objetos (tradução brasileira da 3a edição)
- SCHILDT, H. C completo e total, 3.ed Pearson, 1997.
- C++ Reference <http://www.cplusplus.com/reference/>
- The GNU C Reference Manual  
<http://www.gnu.org/software/gnu-c-manual/gnu-c-manual.html>
- Stack Overflow <http://www.stackoverflow.com>
- [https://www3.ntu.edu.sg/home/ehchua/programming/cpp/gcc\\_make.html](https://www3.ntu.edu.sg/home/ehchua/programming/cpp/gcc_make.html)
- <https://blog.geekhunter.com.br/docker-na-pratica-como-construir-uma-aplicacao/>