

C++

Básico ao Avançado

Introducto

Heitor Rodrigues Savegnago

UFABC Rocket Design

2017.3

- 1 Sobre o curso
- 2 Um pouco sobre C++
- 3 Um pouco de código
- 4 GNU/Linux e o GCC
- 5 Hora de brincar

Sobre o professor

Sobre o professor

- Técnico de mecatrônica - ETEC Jorge Street - 2014

Sobre o professor

- Técnico de mecatrônica - ETEC Jorge Street - 2014
- Ingressante de 2015

Sobre o professor

- Técnico de mecatrônica - ETEC Jorge Street - 2014
- Ingressante de 2015
- Programador na equipe UFABC Rocket Design

Sobre o professor

- Técnico de mecatrônica - ETEC Jorge Street - 2014
- Ingressante de 2015
- Programador na equipe UFABC Rocket Design
- Ciência da Computação

Sobre o professor

- Técnico de mecatrônica - ETEC Jorge Street - 2014
- Ingressante de 2015
- Programador na equipe UFABC Rocket Design
- Ciência da Computação
- Engenharia de Informação

Sobre o professor

- Técnico de mecatrônica - ETEC Jorge Street - 2014
- Ingressante de 2015
- Programador na equipe UFABC Rocket Design
- Ciência da Computação
- Engenharia de Informação
- Entusiasta de tecnologia

Sobre o curso

Sobre o curso

- Essencialmente presencial

Sobre o curso

- Essencialmente presencial
- Temos um site! <sites.google.com/view/heckrodsav>

Sobre o curso

- Essencialmente presencial
- Temos um site! <sites.google.com/view/heckrodsav>
- Presença: chamada toda aula

Sobre o curso

- Essencialmente presencial
- Temos um site! <sites.google.com/view/heckrodsav>
- Presença: chamada toda aula
- Avaliação: pequenas atividades ao longo do curso

Sobre o curso

- Essencialmente presencial
- Temos um site! <sites.google.com/view/heckrodsav>
- Presença: chamada toda aula
- Avaliação: pequenas atividades ao longo do curso
- Aulas em horários *relativamente* flexíveis

Sobre o curso

- Essencialmente presencial
- Temos um site! <sites.google.com/view/heckrodsav>
- Presença: chamada toda aula
- Avaliação: pequenas atividades ao longo do curso
- Aulas em horários *relativamente* flexíveis
- Temos um servidor! <c9.io/heck0cpp0course>

Sobre o curso

- Essencialmente presencial
- Temos um site! <sites.google.com/view/heckrodsav>
- Presença: chamada toda aula
- Avaliação: pequenas atividades ao longo do curso
- Aulas em horários *relativamente* flexíveis
- Temos um servidor! <c9.io/heck0cpp0course>
- Ao fim de cada aula, um momento para praticar o conteúdo

Sobre o curso

- Essencialmente presencial
- Temos um site! <sites.google.com/view/heckrodsav>
- Presença: chamada toda aula
- Avaliação: pequenas atividades ao longo do curso
- Aulas em horários *relativamente* flexíveis
- Temos um servidor! <c9.io/heck0cpp0course>
- Ao fim de cada aula, um momento para praticar o conteúdo
- Esquema de cores para indicar nível de dificuldade



Breve história

Breve história

- Surgiu em 1979

Breve história

- Surgiu em 1979
- Deriva diretamente da linguagem C

Breve história

- Surgiu em 1979
- Deriva diretamente da linguagem C
- Primeira padronização em 1998, depois 2003, 2011 e 2014

Breve história

- Surgiu em 1979
- Deriva diretamente da linguagem C
- Primeira padronização em 1998, depois 2003, 2011 e 2014
- Próxima atualização em 2017

Breve história

- Surgiu em 1979
- Deriva diretamente da linguagem C
- Primeira padronização em 1998, depois 2003, 2011 e 2014
- Próxima atualização em 2017
- Multiparadigma!

Porque usar C++?

Porque usar C++?

- Realmente rápido, perdendo para linguagens de máquina, como Assembly

Porque usar C++?

- Realmente rápido, perdendo para linguagens de máquina, como Assembly
- Amplamente difundida, utilizada e referenciada

Porque usar C++?

- Realmente rápido, perdendo para linguagens de máquina, como Assembly
- Amplamente difundida, utilizada e referenciada
- Acesso a hardware

Porque usar C++?

- Realmente rápido, perdendo para linguagens de máquina, como Assembly
- Amplamente difundida, utilizada e referenciada
- Acesso a hardware
- Respeita o Programador

Porque usar C++?

- Realmente rápido, perdendo para linguagens de máquina, como Assembly
- Amplamente difundida, utilizada e referenciada
- Acesso a hardware
- Respeita o Programador
- Falar mais vantagens é *spoiler*

O que é feito em C++?

O que é feito em C++?

- Adobe Photoshop

O que é feito em C++?

- Adobe Photoshop
- Arduino

O que é feito em C++?

- Adobe Photoshop
- Arduino
- Counter-Strike

O que é feito em C++?

- Adobe Photoshop
- Arduino
- Counter-Strike
- GTA

O que é feito em C++?

- Adobe Photoshop
- Arduino
- Counter-Strike
- GTA
- Half-Life

O que é feito em C++?

- Adobe Photoshop
- Arduino
- Counter-Strike
- GTA
- Half-Life
- JVM - Java

O que é feito em C++?

- Adobe Photoshop
- Arduino
- Counter-Strike
- GTA
- Half-Life
- JVM - Java
- Linux

O que é feito em C++?

- Adobe Photoshop
- Arduino
- Counter-Strike
- GTA
- Half-Life
- JVM - Java
- Linux
- Maya

O que é feito em C++?

- Adobe Photoshop
- Arduino
- Counter-Strike
- GTA
- Half-Life
- JVM - Java
- Linux
- Maya
- Microsoft Office

O que é feito em C++?

- Adobe Photoshop
- Arduino
- Counter-Strike
- GTA
- Half-Life
- JVM - Java
- Linux
- Maya
- Microsoft Office
- Mozilla Firefox

O que é feito em C++?

- Adobe Photoshop
- Arduino
- Counter-Strike
- GTA
- Half-Life
- JVM - Java
- Linux
- Maya
- Microsoft Office
- Mozilla Firefox
- The Sims

O que é feito em C++?

- Adobe Photoshop
- Arduino
- Counter-Strike
- GTA
- Half-Life
- JVM - Java
- Linux
- Maya
- Microsoft Office
- Mozilla Firefox
- The Sims
- Tibia

O que é feito em C++?

- Adobe Photoshop
- Arduino
- Counter-Strike
- GTA
- Half-Life
- JVM - Java
- Linux
- Maya
- Microsoft Office
- Mozilla Firefox
- The Sims
- Tibia
- Unreal Engine 4

O que é feito em C++?

- Adobe Photoshop
- Arduino
- Counter-Strike
- GTA
- Half-Life
- JVM - Java
- Linux
- Maya
- Microsoft Office
- Mozilla Firefox
- The Sims
- Tibia
- Unreal Engine 4
- Windows

O código mínimo

O código mínimo é base pra entendimento da linguagem, porém sua compreensão requer uma base sólida de conceitos que serão abordados posteriormente.

Ele faz absolutamente nada. 😊

```
1 int main()  
2 {  
3     return 0;  
4 }
```

Sintaxe básica

Sintaxe básica

{ ; }

Sintaxe básica

{;}

A≠a

Sintaxe básica

{;}

A≠a

//Linear

Sintaxe básica

{;}

A≠a

//Linear

/*Blocular*/

Onde programar?

Onde programar?

- Code::Blocks

Onde programar?

- Code::Blocks
- O GNU/Linux

Onde programar?

- Code::Blocks
- O GNU/Linux
- *Open source*

Onde programar?

- Code::Blocks
- O GNU/Linux
- *Open source*
- O GCC

Onde programar?

- Code::Blocks
- O GNU/Linux
- *Open source*
- O GCC

```
user@pc:~$ g++ code.cpp -o app.run
```

Hello Wolrd!

O programa mais comum para iniciar é o *Hello Wolrd*, que consiste em uma saída de texto simples. O C++ permite sua escrita de várias formas diferentes.

```
1 #include <iostream>
2 int main()
3 {
4     puts("Hellow World!");
5     return 0;
6 }
```

É a forma mais arcaica de apresentar uma informação, a mais complicada.

Hello Wolrd!

O programa mais comum para iniciar é o *Hello Wolrd*, que consiste em uma saída de texto simples. O C++ permite sua escrita de várias formas diferentes.

```
1 #include <iostream>
2 int main()
3 {
4     printf("%s\n", "Hellow World!");
5     return 0;
6 }
```

É a forma utilizada no C, que permite formatação de dados de maneira mais intuitiva, porém ultrapassada.

Hello Wolrd!

O programa mais comum para iniciar é o *Hello Wolrd*, que consiste em uma saída de texto simples. O C++ permite sua escrita de várias formas diferentes.

```
1 #include <iostream>
2 int main()
3 {
4     std::cout << "Hello World!" << std::endl;
5     return 0;
6 }
```

É a forma moderna do C++, utiliza o conceito de transmissão de dados, a mais simples.

Vamos testar!