

Aula 01 – Aula Inaugural. Começando a
programar.

Algoritmos e Lógica de Programação



Quem sou eu

- Professor Marcos Nava
- E-mail: marcos.nava@fatec.sp.gov.br

Ementa

- Tipos de Dados
- Testes e condições
- Laços
- Muitos exercícios

Avaliações

- 3 Listas de Exercícios:
 - LE1 – 1 pontos
 - LE2 – 1 pontos
 - LE3 – 2 pontos
- 2 Avaliações em sala de aula
 - P1 – 8 pontos
 - P2 – 8 pontos
- 2 Exercícios de Desafio (2 pontos extras sendo um ponto em cada)
- Média final = $(((((L1 * 0.1) + (L2 * 0.1) + (P1 * 0.8)) + D1) + (((L3 * 0.2) + (P2 * 0.8)) + D2)) / 2) * 0.8) + (IN * 0.2)$
- Perceba que o total de pontos compõem 80% da nota, isto porque ainda existe o Projeto Integrador.
- As datas estão no siga

Notas e Faltas

- Acompanhe suas faltas no siga.
- Médias seguem o padrão da faculdade.
- Não deixe de estudar e vir nas aulas de monitoria.
- Em cada bimestre existem 11 pontos em jogo.
- Comece já a formar sua equipe para o projeto integrador, não deixe para a última hora.
- Nossas provas possuem peso 8 e o Integrador peso 2.

Breve Histórico da Linguagem C

- A Linguagem C foi criada em 1972 por Dennis Ritchie no Bell Telephone Laboratories.
- Era para ser usada na criação do Sistema Operacional Unix, precursor do Linux.
- Diversas empresas criaram suas versões de C o que tornava a programação caótica.
- Em 1983 a ANSI (America National Standards Institute) padronizou a linguagem, criando o que chamamos de ANSI C.

Mil e uma razões para programar em C

- A maioria das linguagens possuem um objetivo específico:
 - COBOL – Linguagem comercial, geralmente de computadores de grande porte.
 - PASCAL – Ensino de programação.
 - FORTRAN – Cálculo científico e engenharia.
 - LISP e PROLOG – Inteligência Artificial
 - E a linguagem C? – Uso geral.

Mil e uma razões para programar em C

- Ela pode ser utilizada em qualquer situação, desde um sistema até na criação de Sistemas Operacionais (tanto o Linux quanto o Windows possuem muito código em C)
- Ela também é usada para criar compiladores de outras linguagens.
- Ela é rápida, quase alcança a velocidade da linguagem Assembly
- É simples pois possui um conjunto pequeno de instruções

Mil e uma razões para programar em C

- É portátil para outros sistemas operacionais e processadores, desde que siga o ANSI C.
- É popular, tem uma grande rede de programadores que a utilizam.
- Possibilita a criação de sistemas modulares.
- É de alto nível, sendo desnecessário o conhecimento de código de máquina
- É estendida por bibliotecas poderosas

Filosofia da Linguagem C

- A linguagem C reflete um pouco a filosofia do sistema Unix:
 - Sensível a caixa (diferencia maiúsculas de minúsculas)
 - Dividir problemas complexos em módulos mais simples e independentes
 - A quantidade pequena de comandos
 - Uso de bibliotecas para resolver problemas

C versus C++

- A linguagem C evoluiu para a C++.
- Sendo assim, a linguagem C++ possui dentro dela a linguagem C, e mais um monte de coisas que fazem dela uma linguagem orientada a objetos.
- De qualquer forma é necessário aprender C para depois migrar para C++.
- Esta evolução ainda ocorre nos dias de hoje, gerando as linguagens mais utilizadas no mercado: C#, Objective C e Java.

Ciclo de Desenvolvimento de uma aplicação

- Antes de começarmos nosso primeiro programa temos que entender o que acontece:
 1. Edição do código fonte.
 2. Compilação do programa.
 3. Linkedição dos objetos
 4. Execução do executável.

Meu primeiro programa

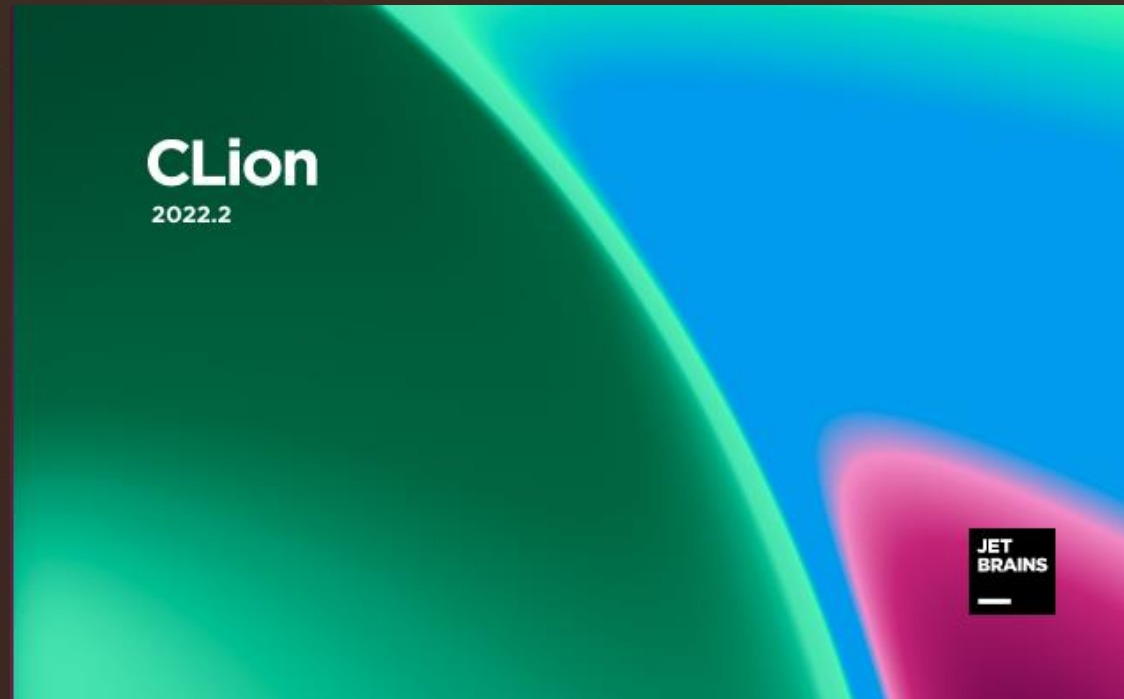
- Veja o programa abaixo:

```
#include <stdio.h>

main()
{
    printf("Ola mundo!\n");
}
```

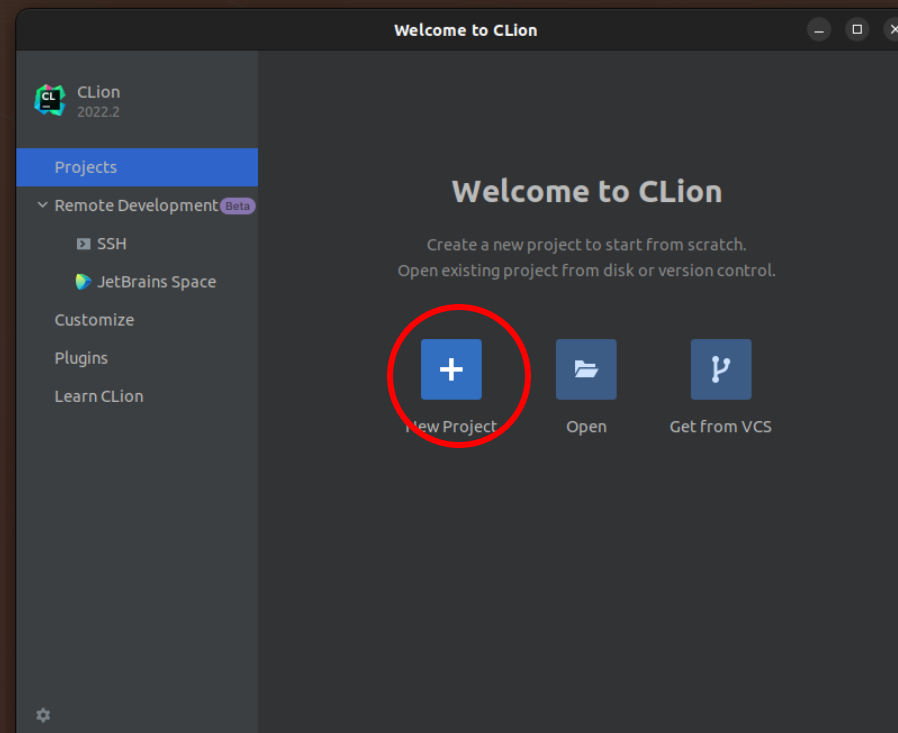

O CLion

- O CLion é um IDE para linguagens C e C++
- Quando executamos o CLion aparece a seguinte tela:



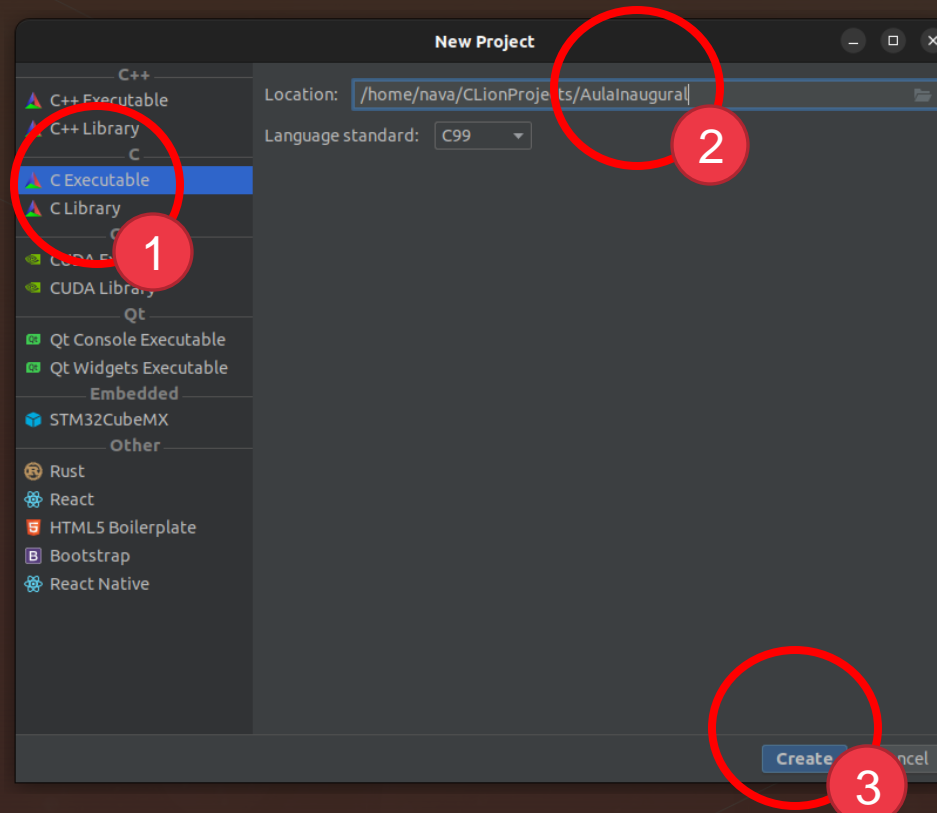
Como criar um projeto

- O CLion trabalha com projetos.
- Para criar um projeto devemos clicar em New Project



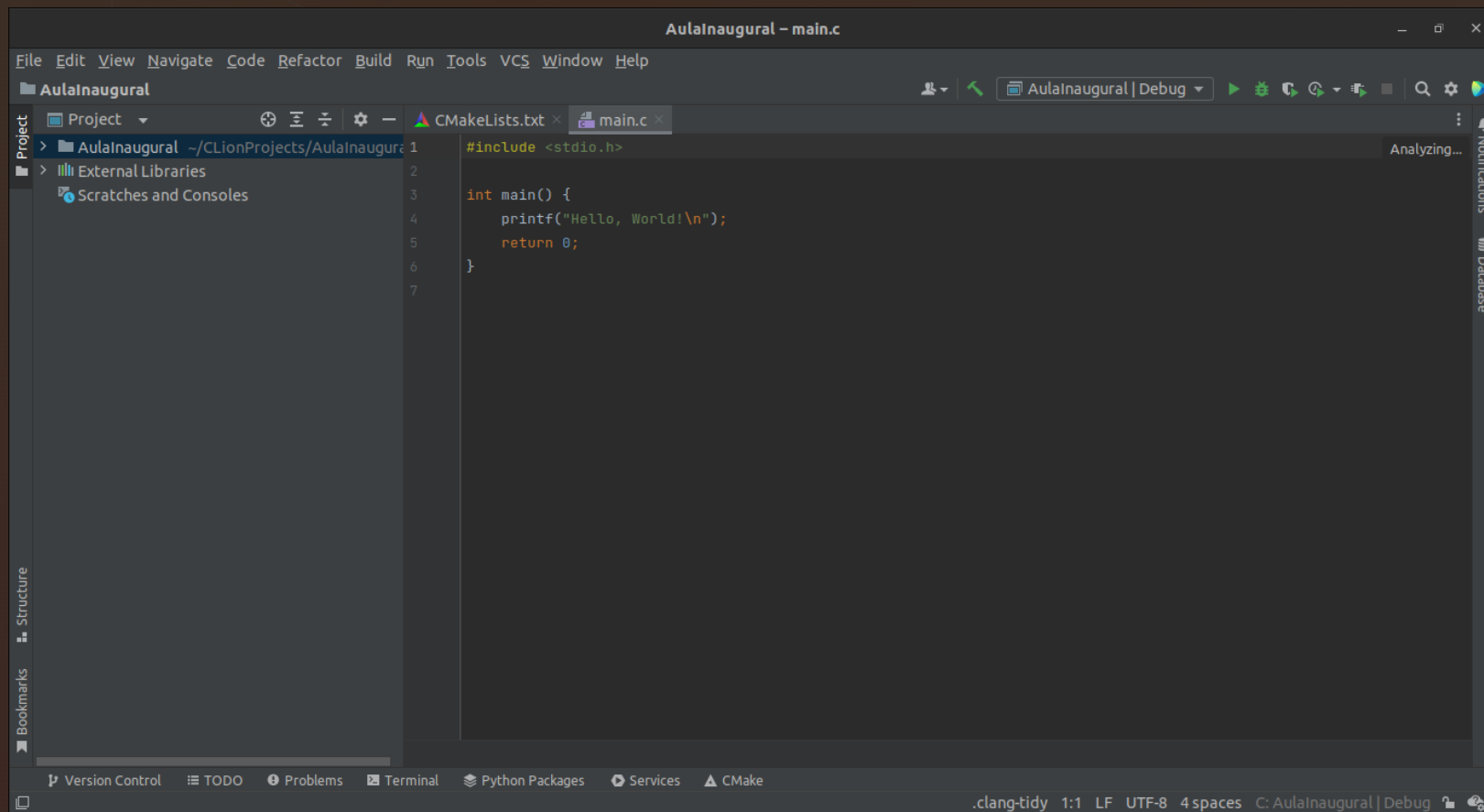
Como criar um projeto

- Logo em seguida aparecerá a tela abaixo e devemos escolher C Executable, digitar o nome do projeto e em seguida em Create.



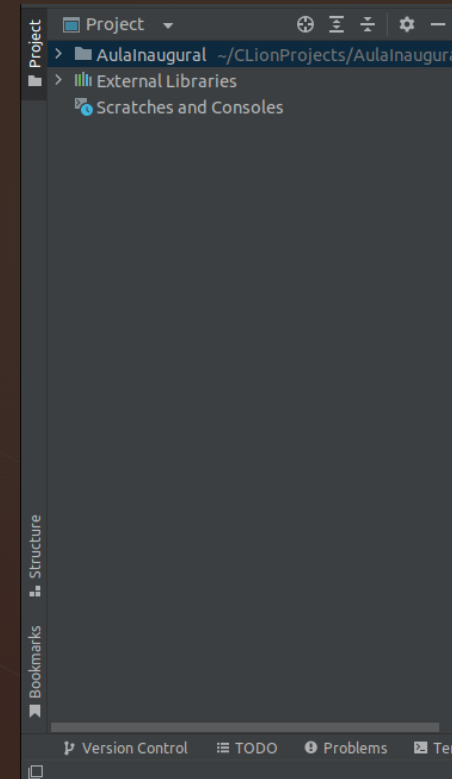
Como criar um projeto

- Aguarde até aparecer a seguinte Janela.



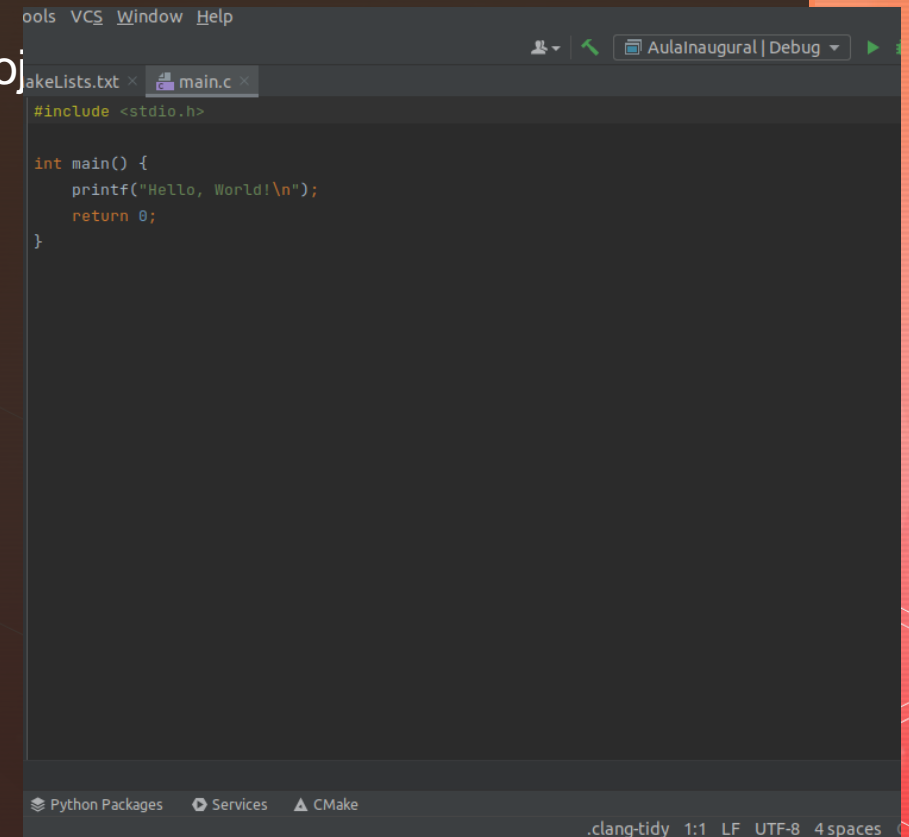
Como criar um projeto

- Nesta tela temos as seguintes áreas:
 - Projeto – Onde podemos ver o seu projeto e outras informações sobre ele



Como criar um projeto

- Nesta tela temos as seguintes áreas:
 - Projeto – Onde podemos ver o seu projeto e outras informações sobre ele.
 - A área de código - Onde vamos trabalhar com o projeto em si.



The screenshot shows a code editor window with a dark theme. The title bar at the top reads 'ools VCS Window Help'. Below the title bar, there are two tabs: 'akeLists.txt' and 'main.c'. The 'main.c' tab is active, showing the following C code:

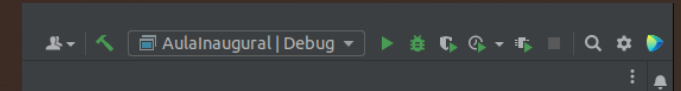
```
#include <stdio.h>

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

At the bottom of the editor, there is a status bar with the text '.clang-tidy 1:1 LF UTF-8 4 spaces'.

Como criar um projeto

- Nesta tela temos as seguintes áreas:
 - Projeto – Onde podemos ver o seu projeto e outras informações sobre ele.
 - A área de código, - Onde vamos trabalhar com o projeto em si.
 - Barra de ferramentas - O podemos executar e debugar nosso código.



Como criar um projeto

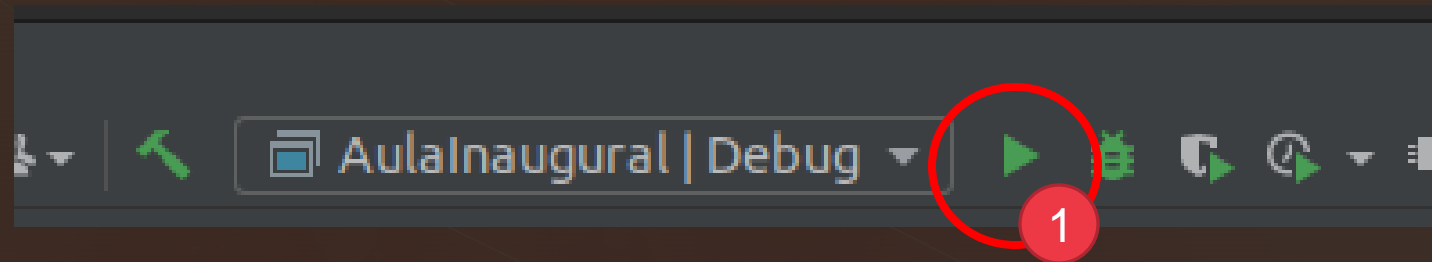
- Agora é só alterar o código conforme a necessidade:

```
#include <stdio.h>

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

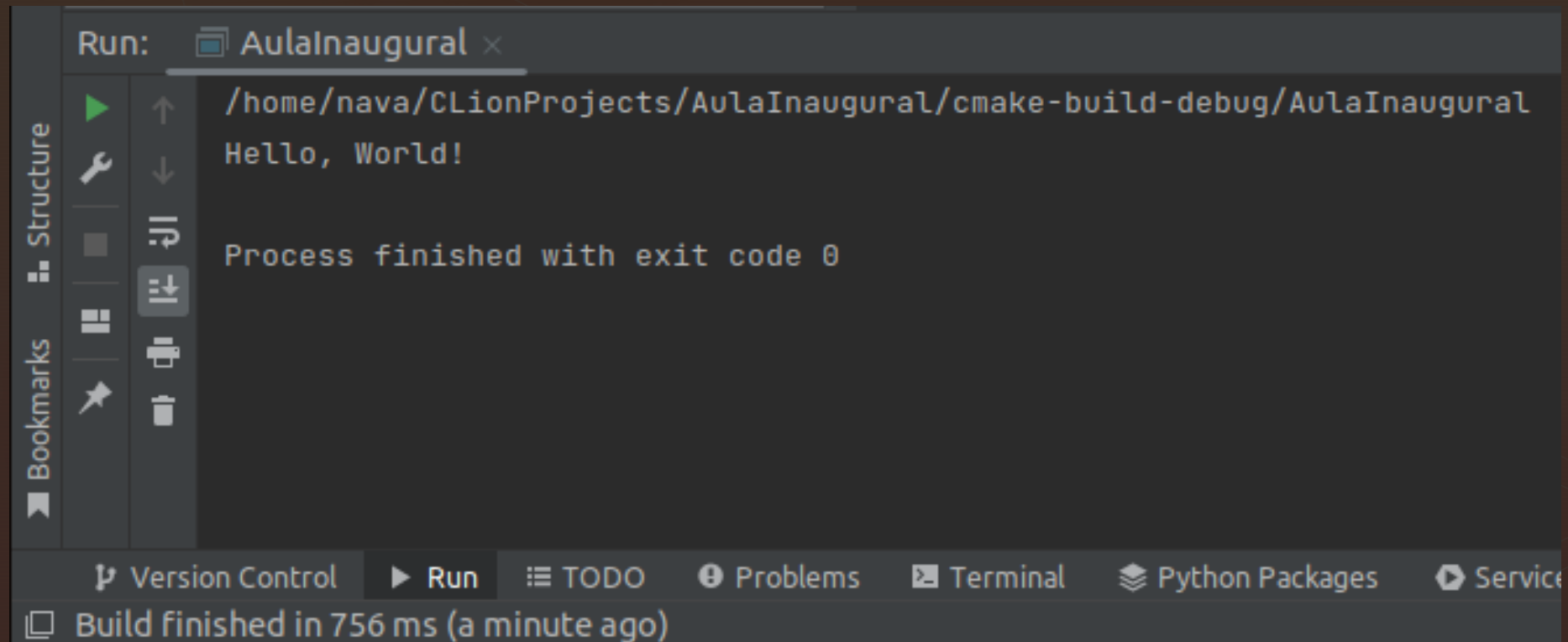
Como executar o programa

- Depois de digitar o programa você só precisa clicar no ícone Run



Como executar o programa

- Se tudo correr bem você receberá tela Run com a execução do seu programa:



Como executar o programa

- Se der algum erro no código você terá aonde o erro está ocorrendo e na tela messages você terá a explicação:

```
#include <stdio.h>

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

Messages: Build x

```
[1/2] Building C object CMakeFiles/AulaInaugural.dir/main.c.o
FAILED: CMakeFiles/AulaInaugural.dir/main.c.o
/usr/bin/cc -g -std=gnu99 -MD -MT CMakeFiles/AulaInaugural.dir/main.c.o -MF CMakeFiles/AulaInaugural.dir/main.c.o.d -o CMakeFiles/AulaInaugural.dir/main.c.o /home/nava/CLionProjects/AulaInaugural/main.c
/home/nava/CLionProjects/AulaInaugural/main.c: In function 'main':
/home/nava/CLionProjects/AulaInaugural/main.c:4:30: error: expected ';' before 'return'
4 |     printf("Hello, World!\\n");
  |                               ^
  |                               ;
5 |     return 0;
  |
```

Build failed in 933 ms

Reverendo o programa

Sempre existe mais de uma solução

Versão Original

```
#include <stdio.h>

main()
{
    printf("Ola mundo!\n");
}
```

Outra versão que dá o mesmo resultado

```
#include <stdio.h>

main()
{
    printf("Ola ");
    printf("mundo!\n");
}
```

Reverendo o programa

Sempre existe mais de uma solução

Versão Original

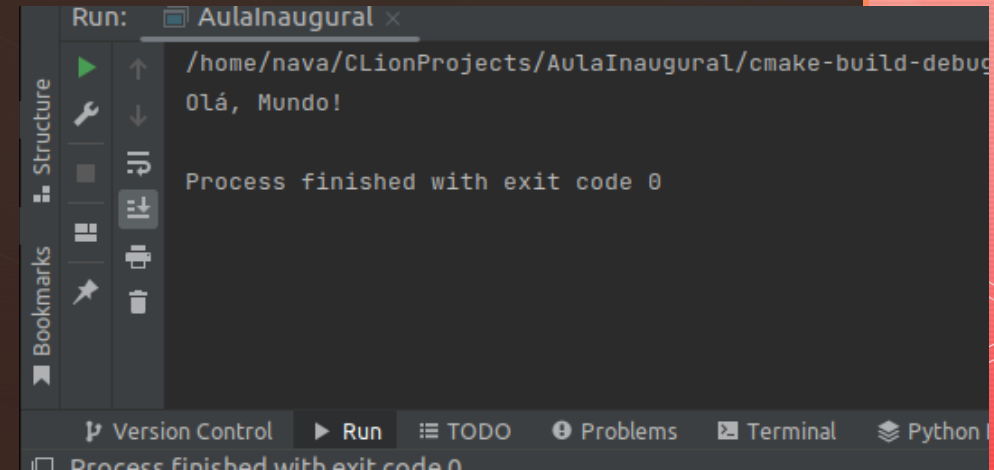
```
#include <stdio.h>

main()
{
    printf("Ola mundo!\n");
}
```

Outra versão que dá o mesmo resultado

```
#include <stdio.h>

main()
{
}
```



Caracteres Especiais e comentários

- Existem alguns caracteres que são especiais. Por exemplo as aspas “ são usadas para um uso específico. Como fazemos se precisarmos imprimi-las?
- Simples, usamos o caracter especial \"

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

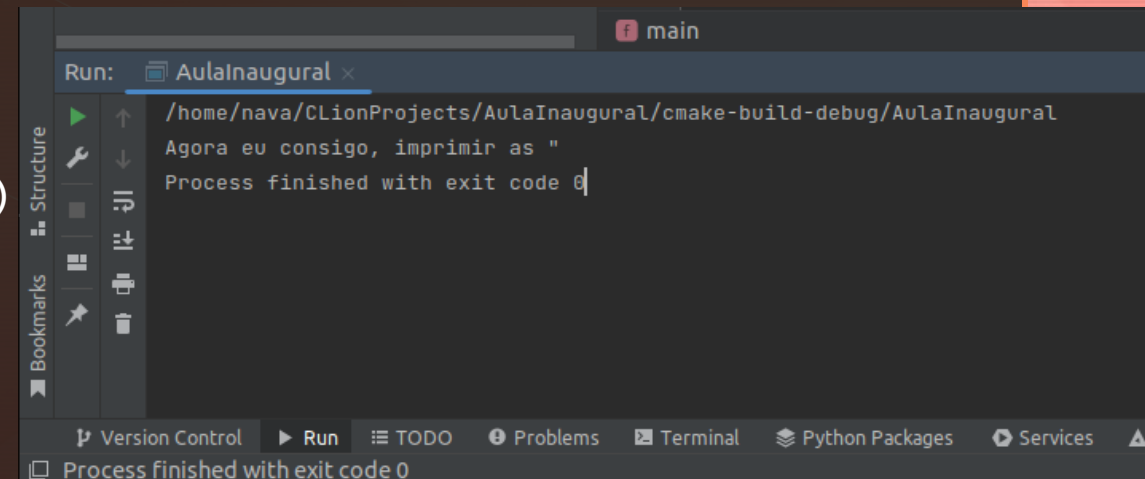
```
main()
```

```
{
```

```
    printf("Agora eu consigo ")
```

```
    printf("imprimir as \");
```

```
}
```



Caracteres Especiais e comentários

- Segue uma lista de caracteres especiais usados em C

Caracter Especial	Descrição
\n	Pular de linha
\7 ou \a	Emite um som de atenção do sistema
\b	Backspace
\r	Vai para a linha de baixo sem ir para o início da linha
\t	Tabulação horizontal
\v	Tabulação vertical
\\	A barra invertida
\” e \’	Imprime as aspas duplas e simples
\?	Imprime a interrogação
\ooo \xnn	Imprime o caracter da tabela ASCII com código ooo em octal ou nn em hexadecimal
%%	Imprime a porcentagem %

Caracteres Especiais e comentários

- Os comentários são textos no seu programa para uso exclusivo do programador.
- Ele não faz parte do programa e é ignorado pelo compilador.
- Existem 2 tipos: o de várias linhas (`/* */`) e o de uma linha (`//`):

```
#include <stdio.h>

main()
{
    /* +-----+
       | Usamos o \n para pular de |
       | linha.                  |
       +-----+ */
    printf("Linha 1 \nLinha 2"); // exemplo
}
```

Exercícios

1. Escreva um programa que emita a seguinte saída na tela:

1 -	Clientes
2 -	Fornecedores
3 -	Faturas
0 -	Sair
2. Qual função deve estar presente em todos os programas em C?

Exercícios

3. Como devem terminar todas as instruções em C?
4. Como é delimitado um bloco de comandos em C?
5. A função printf é parte integrante da linguagem C?
6. Para que serve a linha #include <stdio.h> de um programa?

Exercícios

7. Identifique os erros de compilação dos códigos à seguir:

```
/*  
* Copyright Fatec Mogi Mirim  
*/  
  
#include <stdio.h>  
  
Main()  
{  
    printf("Ola mundo!\n");  
}
```

Exercícios

```
/*  
 * Copyright Fatec Mogi Mirim  
 */  
  
#include <stdio.h>  
  
main  
{  
    printf("Ola mundo!\n");  
}
```

Exercícios

```
/*  
 * Copyright Fatec Mogi Mirim  
 */  
  
#include <stdio.h>  
  
main()  
{  
    print("Ola mundo!\n");  
}
```


Exercícios

```
/*  
 * Copyright Fatec Mogi Mirim  
 */  
  
#include <stdio.h>  
  
main()  
{  
    printf("Ola")("mundo!\n");  
}
```

Exercícios

```
/*  
 * Copyright Fatec Mogi Mirim  
 */  
  
#include <stdio.h>  
  
main()  
{  
    printf("Ola mundo!\n");  
}
```

Exercícios

```
/*  
/* Copyright Fatec Mogi Mirim */  
*/  
#include <stdio.h>  
main()  
{  
    printf("Ola mundo!\n");  
}
```

Exercícios

```
/*  
 * Copyright Fatec Mogi Mirim  
 */  
  
#include <stdio.h>  
  
main()  
{  
    printf(Ola mundo!\n);  
}
```

Exercícios

```
/*  
 * Copyright Fatec Mogi Mirim  
 */  
  
#include <stdio.h>  
  
main()  
{  
    printf("Ola mundo!\n")  
}
```

Exercícios

```
/*  
 * Copyright Fatec Mogi Mirim  
 */  
  
include <stdio.h>  
  
main()  
{  
    printf("Ola mundo!\n");  
}
```


Exercícios

```
/*  
 * Copyright Fatec Mogi Mirim  
 */  
  
#include <stdio.h>  
  
main()  
{  
    printf('Ola mundo!\n');  
}
```

Exercícios

8. Escreva um programa que imprima na tela:

Bem-vindos ao /Mundo\ da programacao em "C"

9. Escreva um programa que imprima uma árvore na tela:

```
*  
***  
*****  
/|\
```