

1 Criando um projeto

Com o *Code::Blocks* aberto, vá em *File* → *New* → *Project...* Selecione a opção *Console Application*, clique em *Go* e depois *Next >*. Selecione a linguagem *C++* e clique em *Next >*. A seguir será mostrado o que deve ser inserido em cada campo:

- *Project Title* → Entre com o nome do projeto.
- *Folder to create project in* → Entre com o caminho para criar o projeto.
- *Project filename* → Entre com o nome do arquivo do projeto. É gerado automaticamente.
- *Resulting filename* → Caminho completo do arquivo do projeto. É gerado automaticamente.

Clique em *Next >*, deixe o compilador *GNU GCC Compiler* selecionado. Marque as opções *Debug* e *Release* e clique em *Finish*.

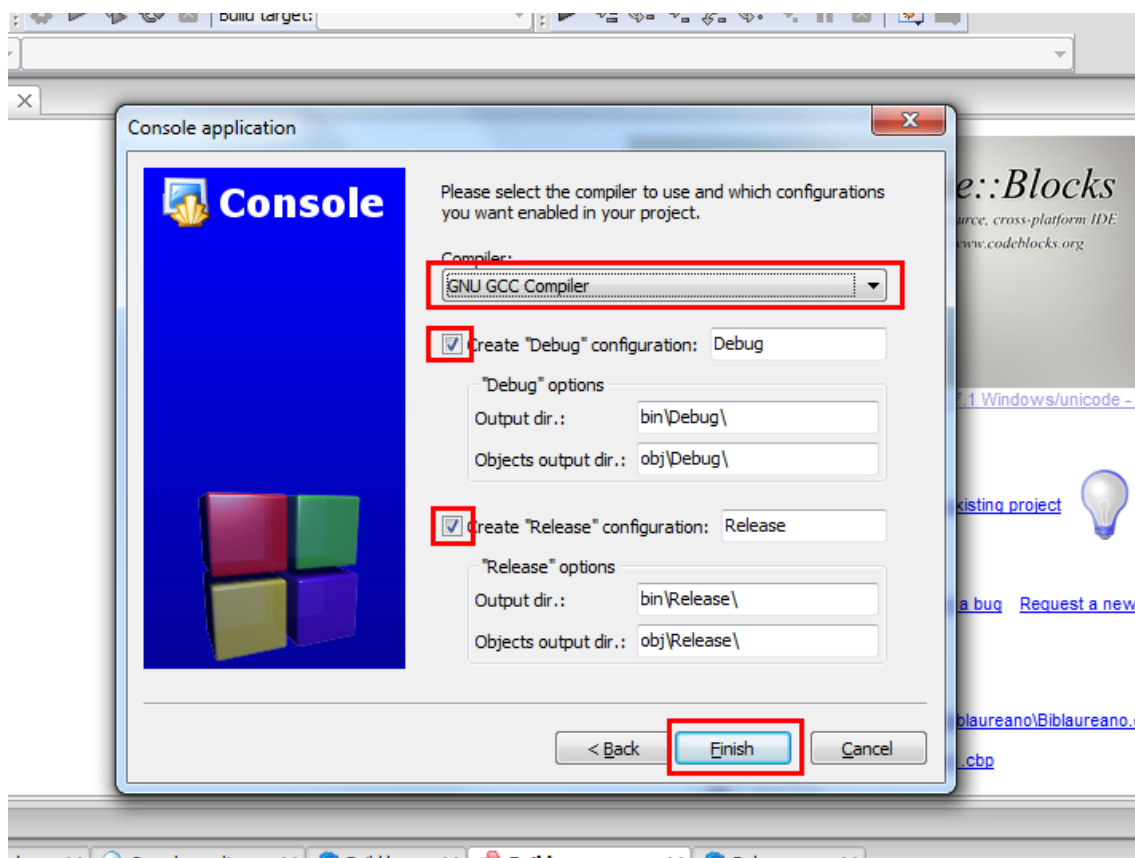


Figura 1: Selecionando o compilador e suas opções

No painel *Management* - que pode ser ativado ou desativado com o atalho *SHIFT + F2*, localizado de maneira padrão à esquerda do *Code::Blocks*, expanda a opção *Sources* e dê dois cliques no arquivo *main.cpp*. Um código-fonte em *C++* será aberto, apague o que estiver ali e coloque o código desejado.

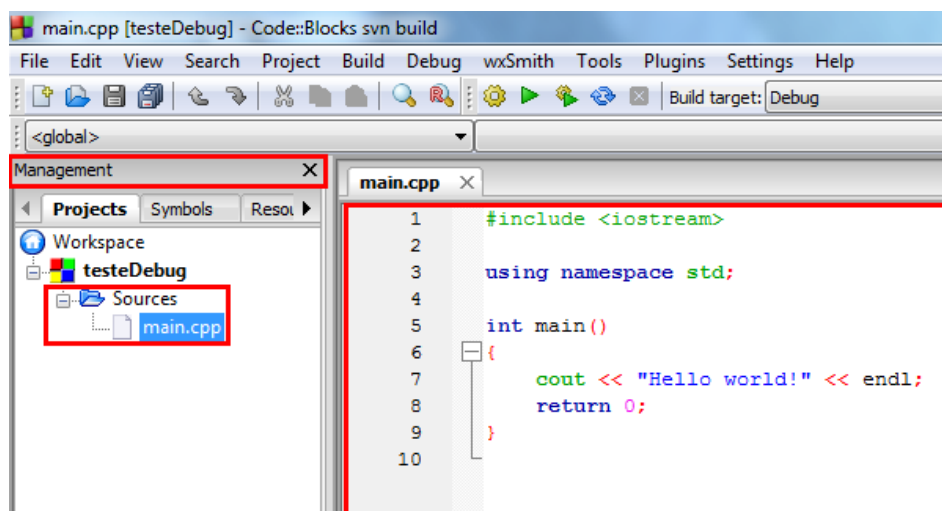


Figura 2: Painel *Management* e o código do seu novo projeto

2 Adicionando *breakpoints*

Para debugar, você deve marcar alguns pontos para "parar" a execução do programa, desviando assim o controle do compilador à você, programador. Esses pontos são chamados de *breakpoints* e podem ser adicionados clicando com o botão esquerdo do mouse ao lado da linha desejada ou, com o cursor do teclado na linha desejada, apertando F5, como na imagem.

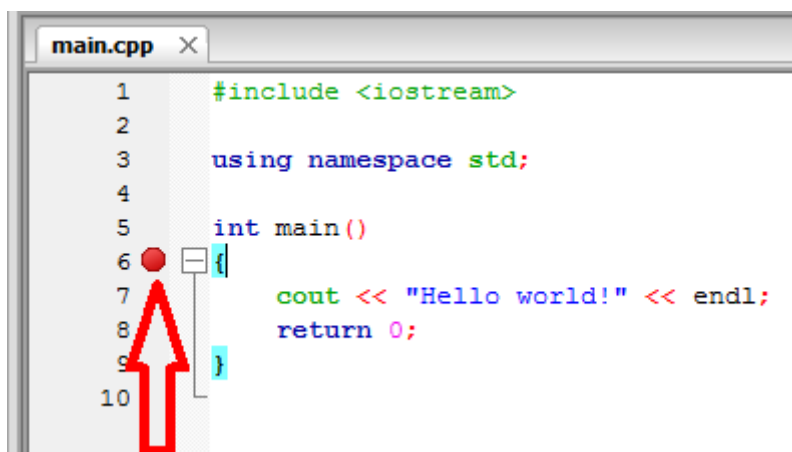


Figura 3: Adicionando um *breakpoint*

É possível adicionar mais de um breakpoint, caso queira parar em várias funções diferentes.

3 Debugging Windows

As Debugging Windows são janelas que nos permitem analisar algumas mudanças que ocorrem em nosso programa durante a execução do mesmo.

Vá em Debug → Debugging Windows → Watches. Uma janela abrirá e de acordo com a sua versão do CodeBlocks duas coisas podem acontecer:

- Aparecer dois menus: *Local variables* e *Function Arguments*. Expanda o menu Local variables para ter acesso às variáveis do escopo atual. O nome das variáveis e o conteúdo que elas armazenam será mostrado.
- A janela ficará em branco: Nesse caso você deve clicar no primeiro campo livre dela e digitar o nome da variável que quer visualizar. Tome cuidado para não digitar o nome da variável errado, ou não será possível visualizá-la.



Figura 4: Um exemplo da janela *Watch*, que nesse caso obedece o segundo caso apresentado acima

4 Debugando

Agora nós vamos começar a debugar o nosso projeto.

4.1 Iniciando o debugador

Vá no menu *Debug* → *Start* ou aperte F8. Um terminal será aberto e uma seta amarela aparecerá sobre o primeiro breakpoint. Essa seta mostra qual será a próxima linha a ser processada.

4.2 Controlando o debugador

Para avançar uma linha aperte F7. Se quiser, pode posicionar o cursor do mouse em uma linha e clicar, então aperte F4 para avançar até a linha do cursor. Nas operações de Input a seta amarela desaparecerá, indicando que o sistema está lendo o teclado. Vá até o console aberto, digite algo e tecele *Enter* para voltar a executar o programa.

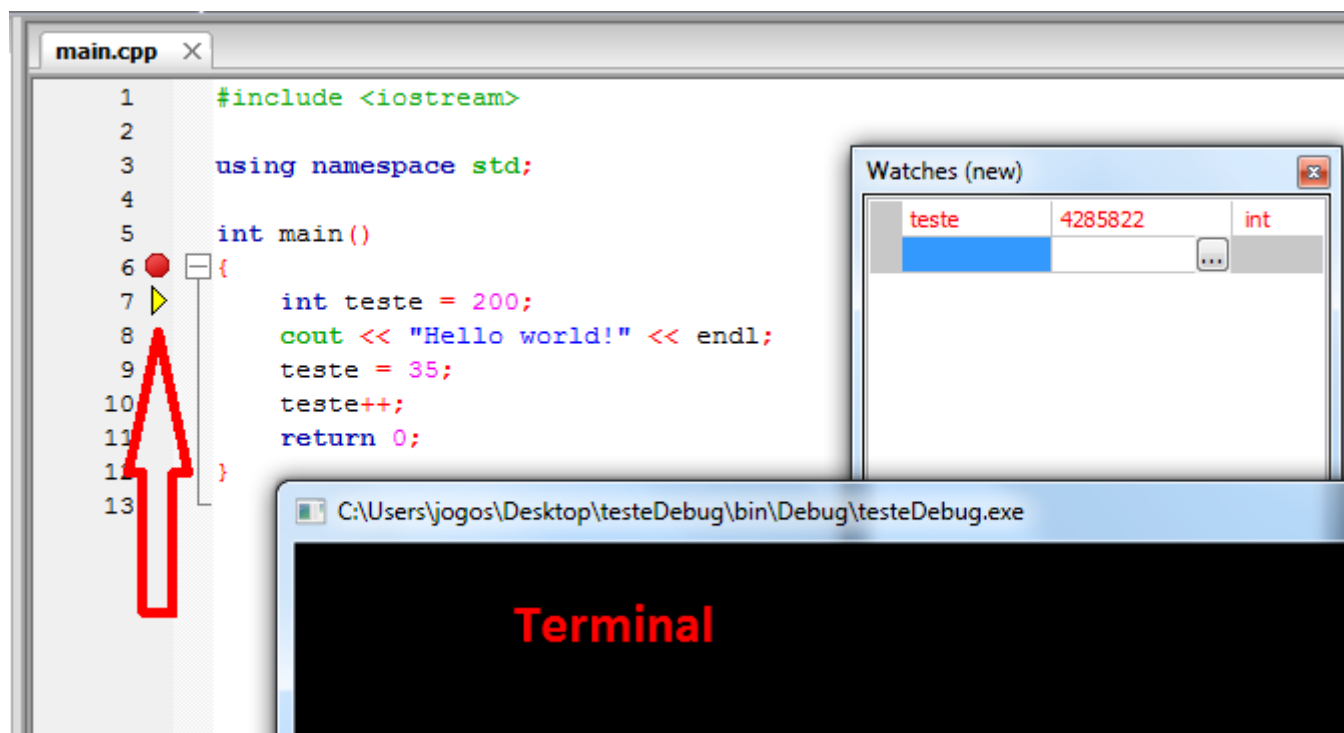


Figura 5: Iniciando o *debugger*

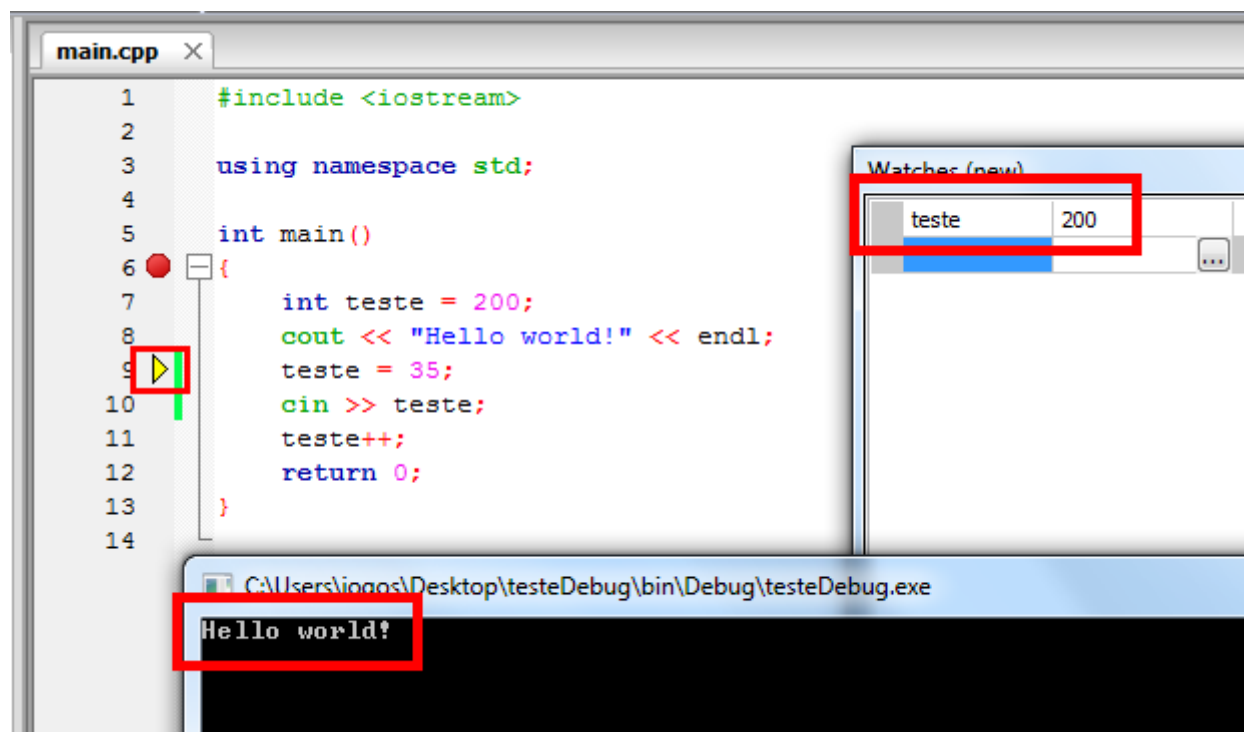


Figura 6: Observe as mudanças que ocorrem no terminal e na janela *Watch* a cada linha que é executada

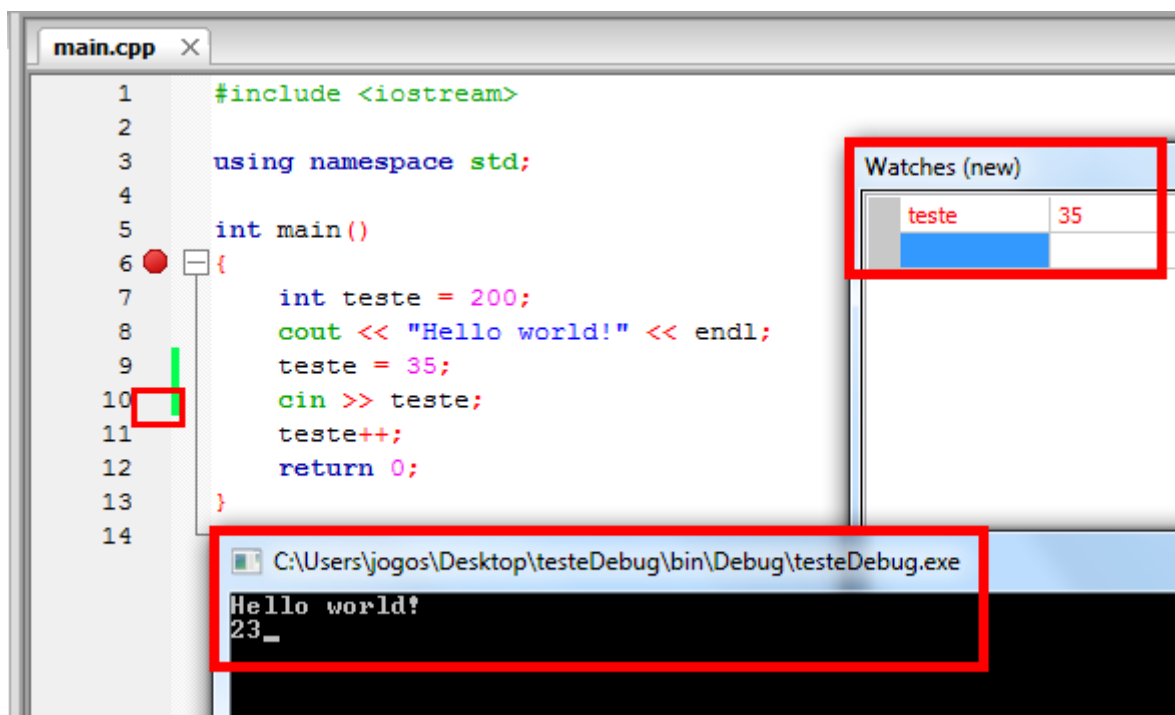


Figura 7: Perceba que a seta desaparece nas operações de leitura

4.3 Atenção

Não esqueça de ir em Debug → Stop debugger - ou apertar F8 - toda vez que parar o programa, ou você não vai conseguir debugar outra vez.