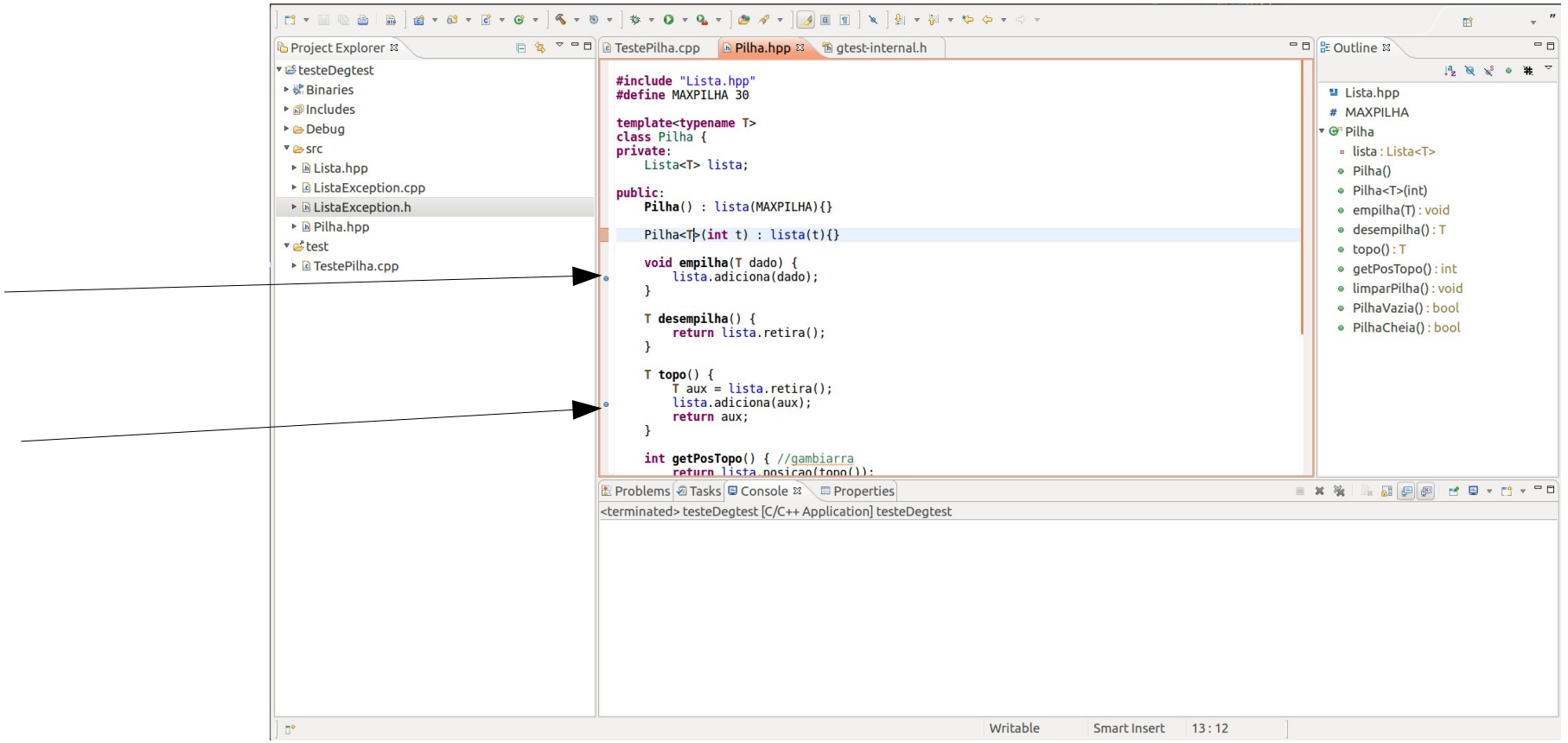
Estruturas de Dados

Depuração e Testes Unitários

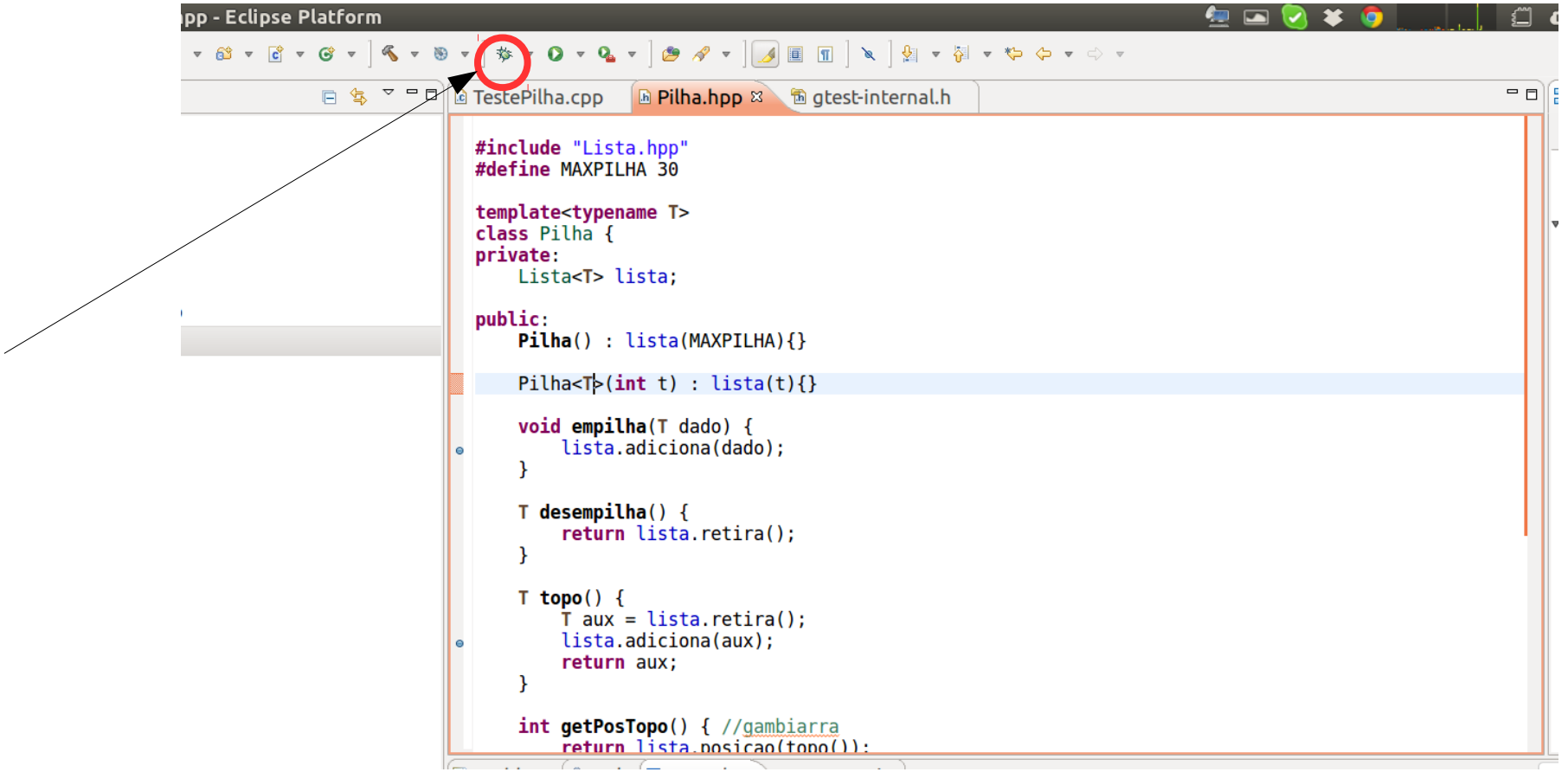
Depurando Código

- O Eclipse vem com uma ótima ferramenta de debug
- Voce pode colocar pontos de parada (break points) so seu código clicando duas vezes na margem direita.
- Pontos de parada são representado por bolinhas azuis



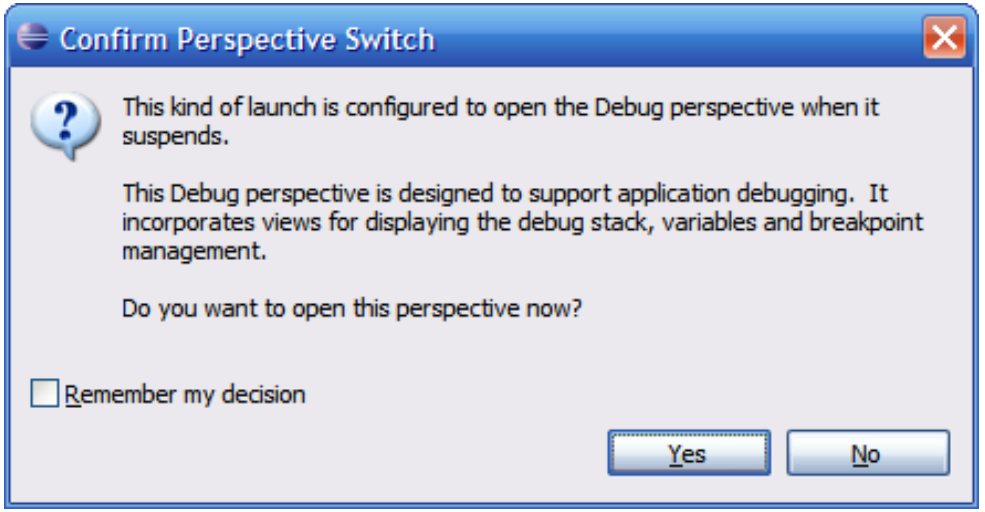
Depurando Código

- Um jeito fácil de entrar no modo de depuração é clicar no botão debug em vermelho na tela abaixo



Depurando Código

- Na primeira tentativa de depuração de código voce vai receber a seguinte mensagem

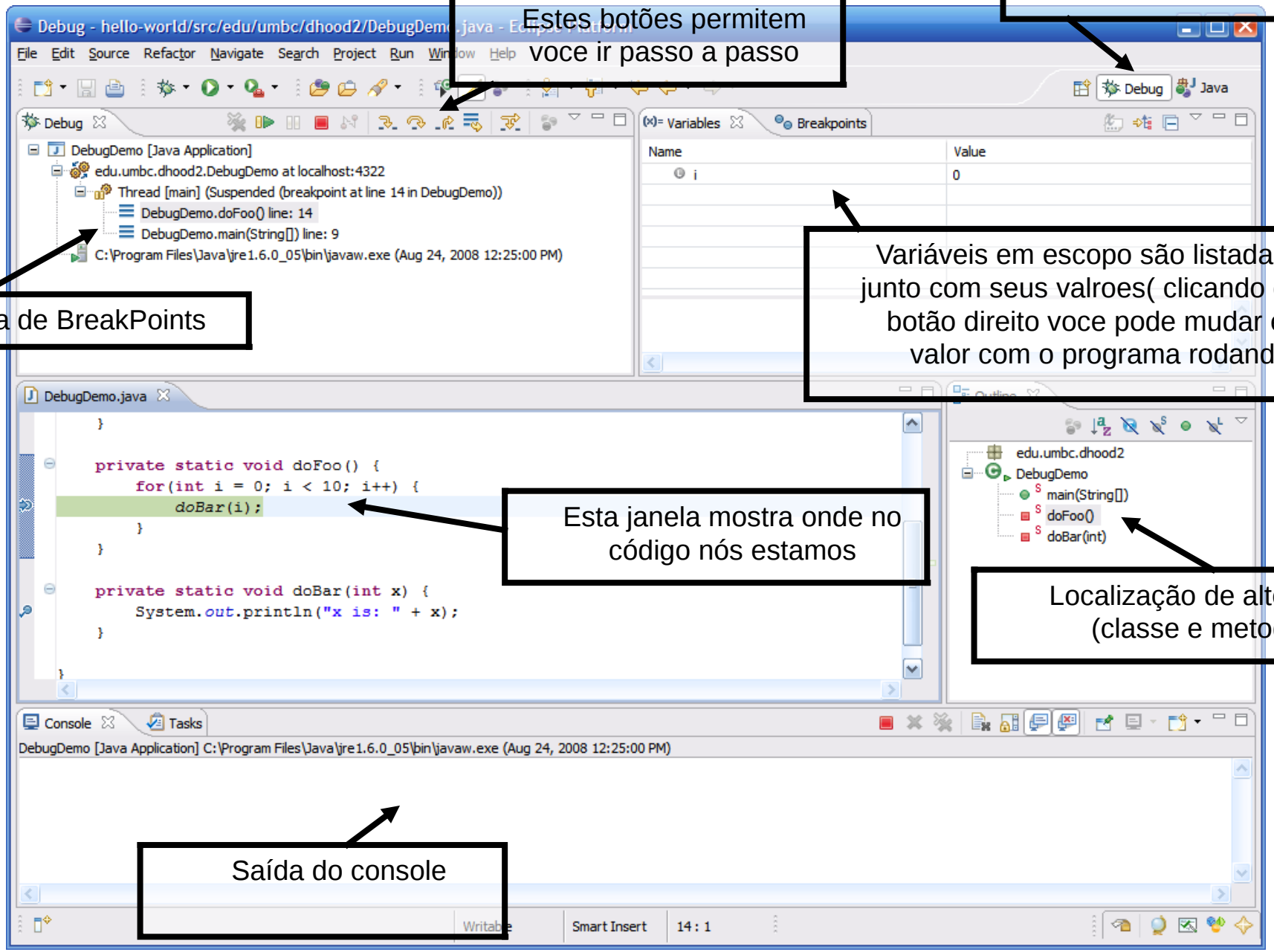


- O Eclipse esta pedindo pra voce mudar para a perspectiva de debug que é o mais indicado. Clique YES
- O Eclipse tem um monte de perspectivas. Nós nurmalmente usamdo a para programar em C++

Perspectiva de Debug

Veja a perspectiva de debug – Clique em C++ para voltar ao normal

Estes botões permitem voce ir passo a passo



Lista de BreakPoints

Variáveis em escopo são listada aqui junto com seus valroes(clicando com o botão direito voce pode mudar este valor com o programa rodando)

Esta janela mostra onde no código nós estamos

Localização de alto nível (classe e metodo)

Saída do console

Testes Unitários

- Não são um conceito novo
- Historicamente ao se fazer algo novo se testam os conceitos primeiro





Porque automatizar os testes

- Todos os programadores sabem que devem testar seu código
- Quanto mais curto o prazo menos testes são realizados
- Necessidade de testar cada método separadamente
- Centralizar o código de testes
- Regras já definidas para apresentação de erros
- Possibilidade de automatização

Tipos de Testes

- Testes de Unidades
- Testes de Integração
- Testes Funcionais

Testes de Unidades

- Testam a aplicação em seus menores componentes, isoladamente
- Testam unidades lógicas
  - Métodos
  - Objetos
- Maior número de erros detectados
- Erros mais fáceis de corrigir
- Devem ser executados continuamente

Testes de Integração

- Testam como uma coleção de unidades interage entre si ou com o ambiente onde executam

- Executados continuamente (caso as unidades em desenvolvimento dependam de outras)

Testes Funcionais

- Ponto de vista do usuário
- Testam casos de uso
- Validam a interface com o usuário, as operações requisitadas, etc.
- São menos estáveis do que os outros tipos

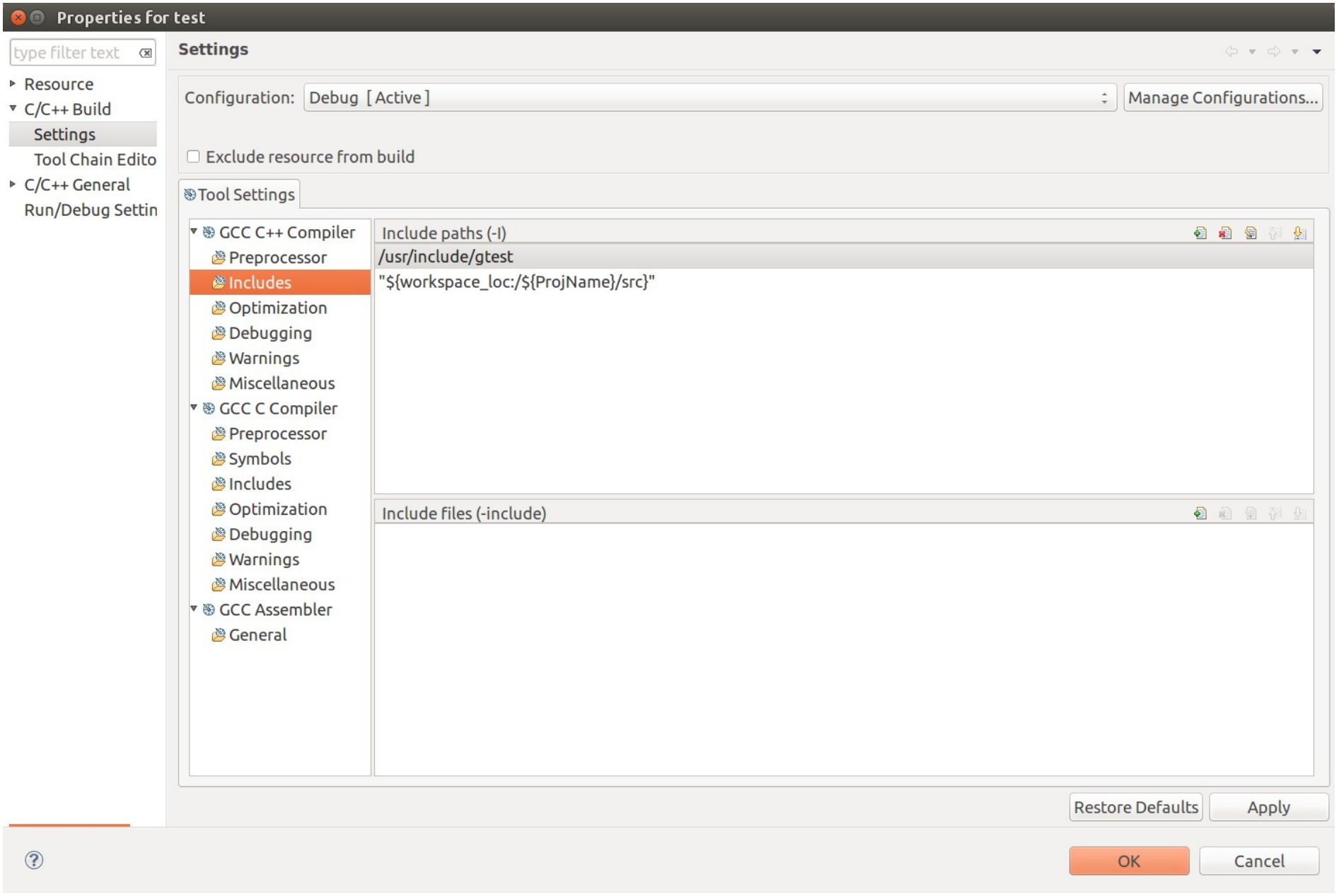
O que testamos em Estruturas de Dados?

- Testes de Unidade
- Os outros testes fazem parte da metodologia usada pelo processo de engenharia de software
  - Não nos interessam no momento
- Como fazemos:
  - Google Test Framework para C++

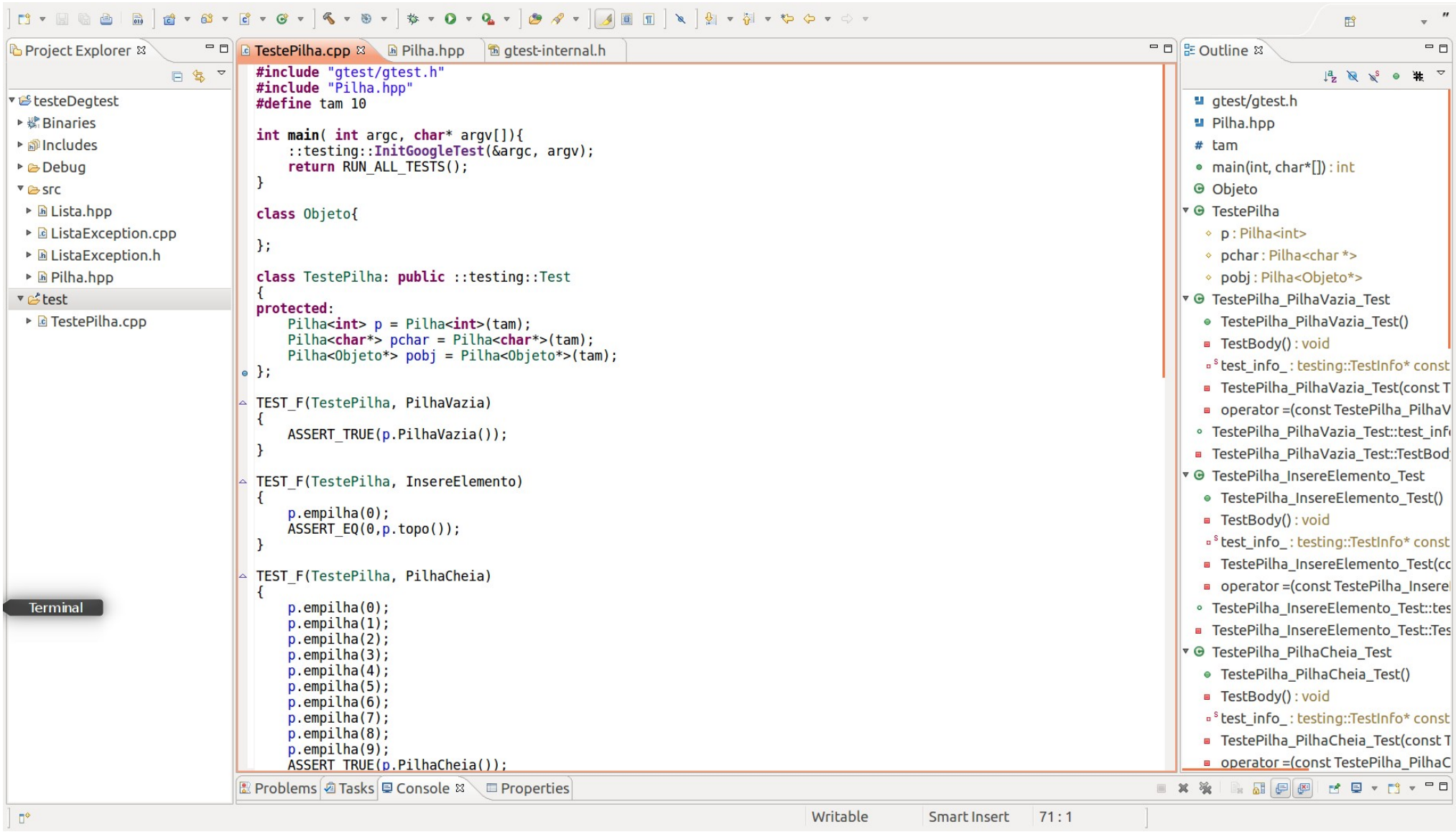
Preparando o Projeto para o Google Test

- Baixe o google test e instale no seu sistema operacional
- Crie seu projeto normalmente
- Dentro do projeto crie um diretório chamado test
- Edite as propriedades do diretório test

Editando as propriedades



Escrevendo o Teste.cpp



Principais Asserções

- ASSERT x EXPECT
- ASSERT_EQ
- ASSERT_TRUE
- ASSERT_NE
- ASSERT_LT
- ASSERT_LE
- ASSERT_GT
- ASSERT_GE
- ASSERT_STREQ
- ASSERT_STRNE
- ASSERT_STRCASEEQ
- ASSERT_STRCASENE
- Ambiente de testes com TEST_F

Exercício

- Crie testes unitários para todas as estruturas que voce for implementar este semestre.
- Os testes unitários devem incluir pelo meno 1 teste por método desenvolvido.
- A entrega de todos os testes unitários no final do semestre tera a nota de um trabalho.
  - A nota será calculada pelo numero de testes entregues.

creative commons

COMMONS DEED

Atribuição-Uso Não-Comercial-Compartilhamento pela Licença 2.5 Brasil

Você pode:
- copiar, distribuir, exibir e executar a obra
- criar obras derivadas

Sob as seguintes condições:
Atribuição — Você deve dar crédito ao autor original, da forma especificada pelo autor ou licenciante.
Uso Não-Comercial — Você não pode utilizar esta obra com finalidades comerciais.
Compartilhamento pela mesma Licença — Se você alterar, transformar, ou criar outra obra com base nesta, você somente poderá distribuir a obra resultante sob uma licença idêntica a esta.

Para ver uma cópia desta licença, visite http://creativecommons.org/licenses/by-nc-sa/2.5/br/ ou mande uma carta para Creative Commons, 171 Second Street, Suite 300, San Francisco, California, 94105, USA.