



PCS-3111

**Laboratório de Programação
Orientada a Objetos para
Engenharia Elétrica**

Aula 5: Teclado / Codeblocks

Agenda

1. Entrada via teclado
2. Esqueleto de métodos no Code::Blocks

Entrada via teclado

Detalhes de C++

■ Obter uma string do teclado

```
string texto1, texto2;  
cin >> texto1;  
cin >> texto2;
```

Não considera texto com espaço...

A próxima palavra digitada vai para texto2

• **Solução:** função getline

- Uso: `getline(cin, umaString)`
 - Não captura o '\n'
- Necessário fazer `#include <string>`

• ...o texto digitado anteriormente por um cin pode ser "capturado" pelo getline...

- Necessário **ignorar** os caracteres digitados anteriormente
 - Método `ignore` do `cin`

```
cin.ignore(100, '\n') // ignora 100 caracteres ou até  
                     // encontrar um '\n'
```

Detalhes de C++

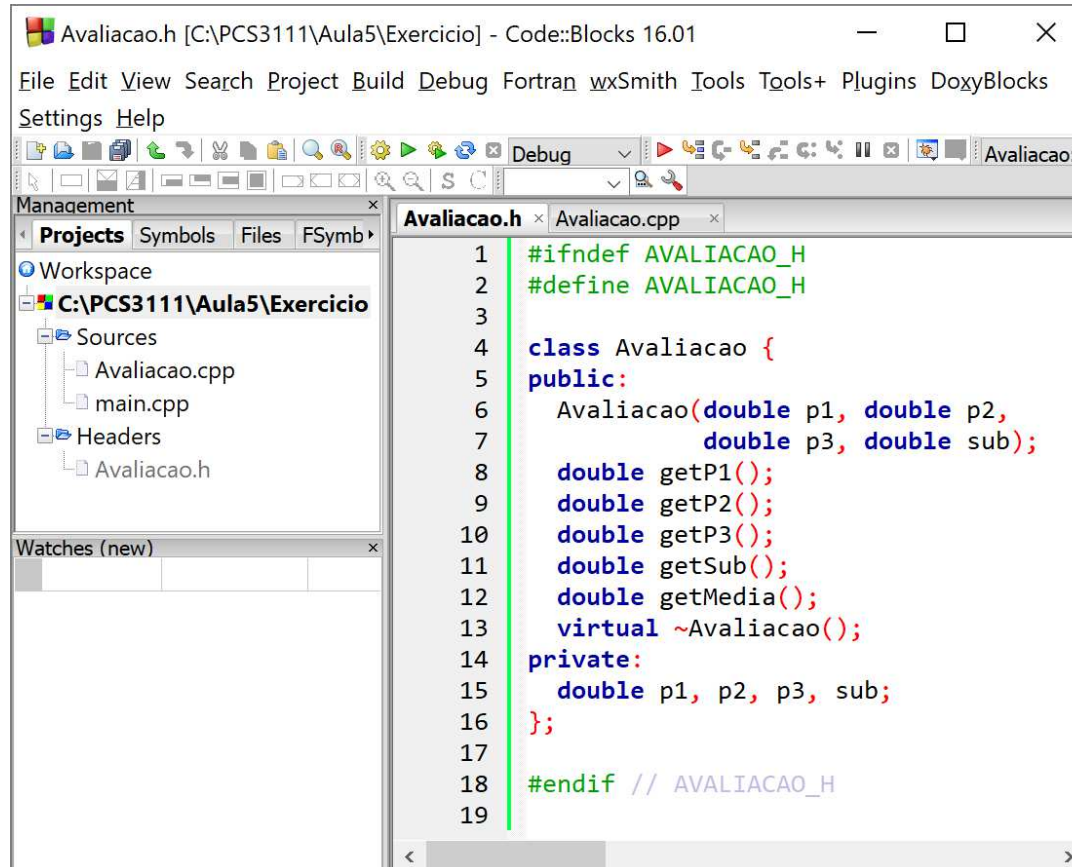
■ Exemplo

```
1  #include <iostream>
2  #include <string>
3  using namespace std;
4
5  int main (int argc, char** argv) {
6      string texto1, texto2;
7      cin >> texto1;
8
9      cin.ignore (100, '\n');
10     getline (cin, texto2);
11
12     cout << texto1 << endl;
13     cout << texto2 << endl;
14     return 0;
15 }
```

Esqueleto de métodos

Esqueleto de métodos

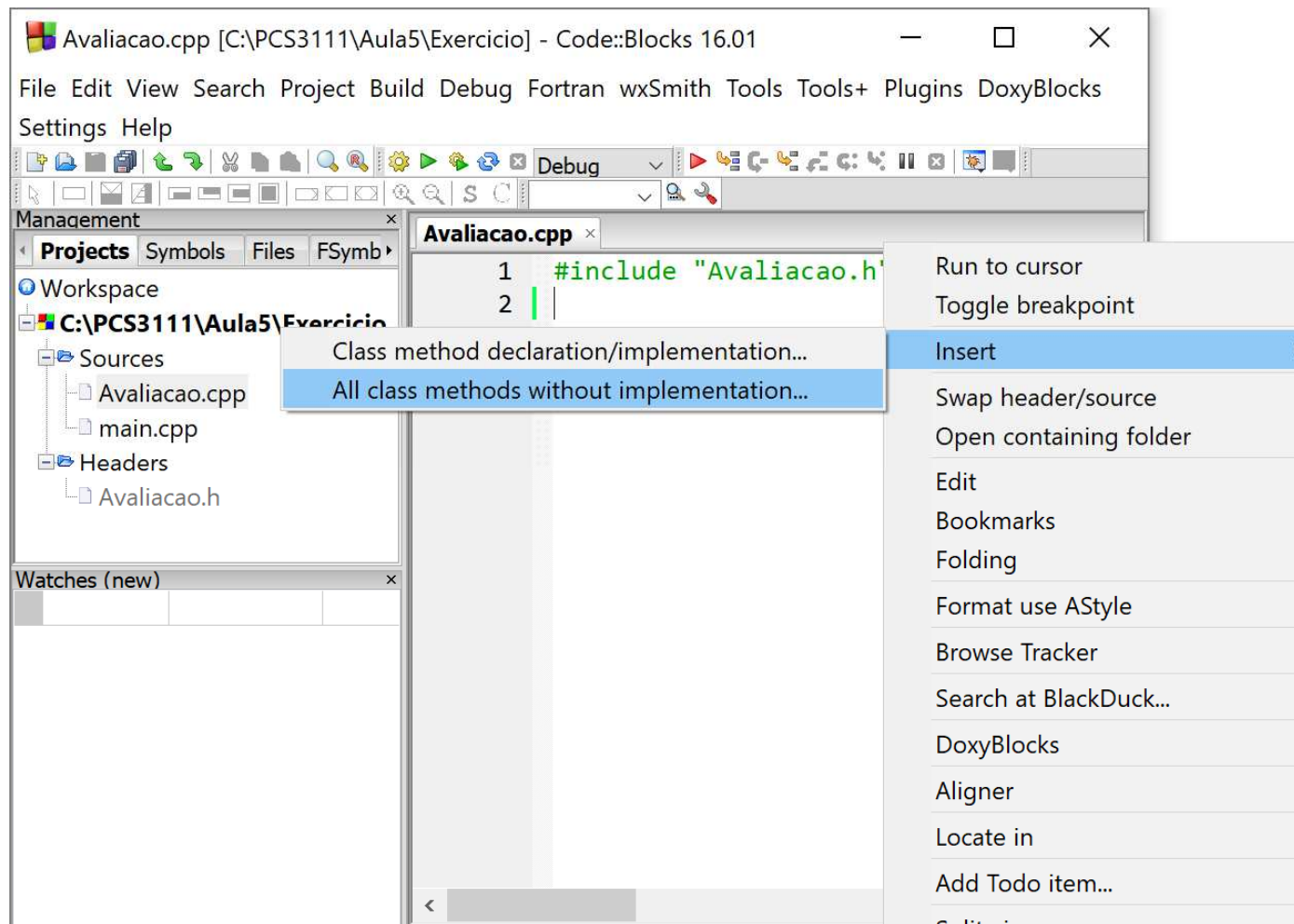
- O CodeBlocks cria o esqueleto dos métodos no “.cpp” a partir das definições no código “.h”
 - *Exemplo:* considere o Avaliacao.h



```
1  #ifndef AVALIACAO_H
2  #define AVALIACAO_H
3
4  class Avaliacao {
5  public:
6      Avaliacao(double p1, double p2,
7                double p3, double sub);
8
9      double getP1();
10     double getP2();
11     double getP3();
12     double getSub();
13     double getMedia();
14     virtual ~Avaliacao();
15 private:
16     double p1, p2, p3, sub;
17 };
18 #endif // AVALIACAO_H
19
```

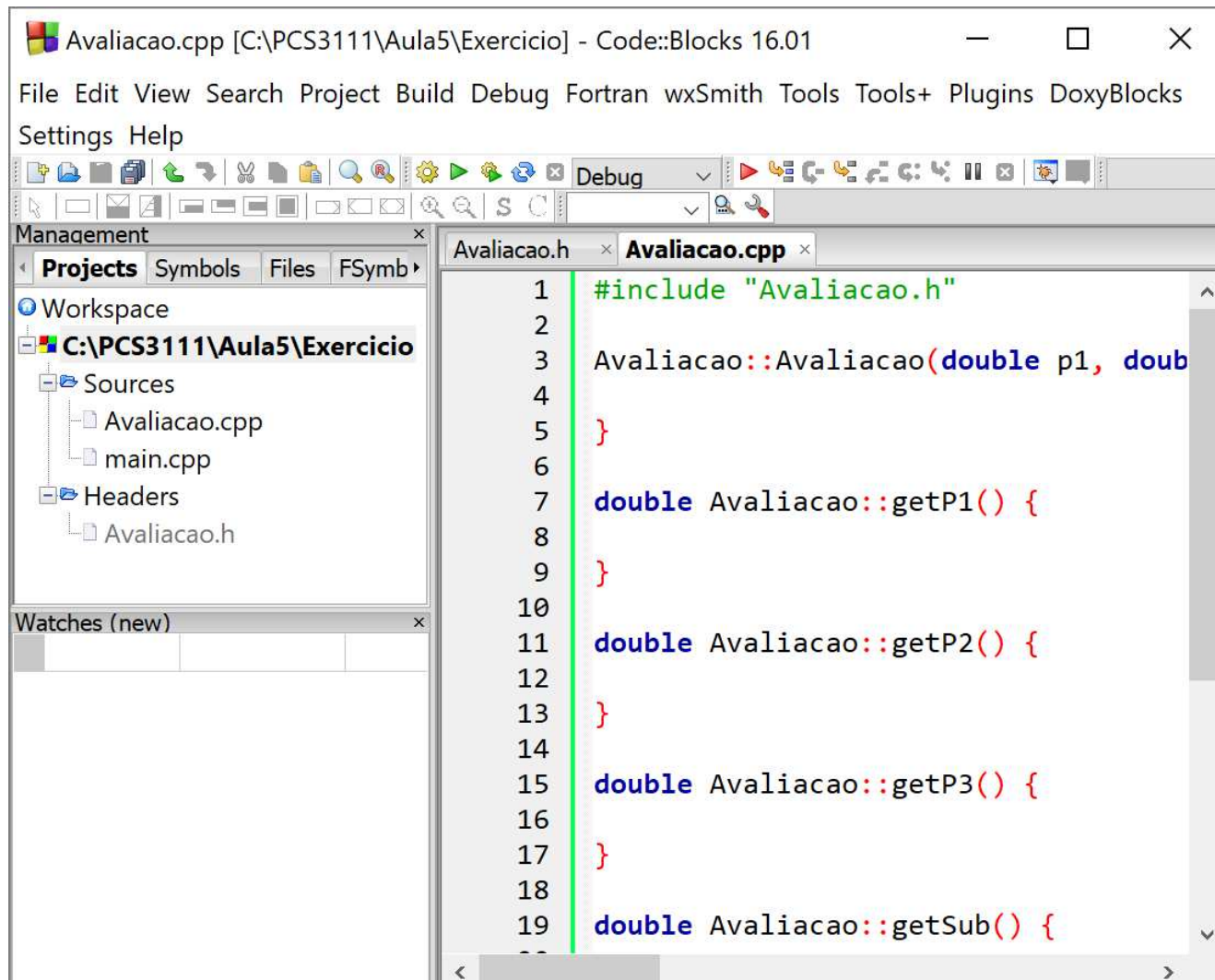
Esqueleto de métodos

- No corpo do arquivo “.cpp”
 - Botão direito → Insert → All class methods without implementation



Esqueleto de métodos

- Resultado



The screenshot shows the Code::Blocks IDE interface. The title bar indicates the file is 'Avaliacao.cpp' located at 'C:\PCS3111\Aula5\Exercicio' and the version is '16.01'. The menu bar includes File, Edit, View, Search, Project, Build, Debug, Fortran, wxSmith, Tools, Tools+, Plugins, DoxyBlocks, Settings, and Help. The toolbar contains various icons for file operations, editing, and debugging. The 'Management' panel on the left shows the project structure: 'Workspace' containing 'C:\PCS3111\Aula5\Exercicio', which has 'Sources' (Avaliacao.cpp, main.cpp) and 'Headers' (Avaliacao.h). The 'Watches (new)' panel is empty. The main editor window shows the code for 'Avaliacao.cpp' with line numbers 1 through 19. The code is as follows:

```
1  #include "Avaliacao.h"
2
3  Avaliacao::Avaliacao(double p1, doub
4
5  }
6
7  double Avaliacao::getP1() {
8
9  }
10
11 double Avaliacao::getP2() {
12
13 }
14
15 double Avaliacao::getP3() {
16
17 }
18
19 double Avaliacao::getSub() {
```