

人工知能入門

-探索による人工知能-

Lecture 14

五目並べプログラムの調整

<http://www2.teu.ac.jp/gamelab/LECTURES/ArtificialIntelligence/index.html>

最終日の目標

★ 簡単な五目並べプログラムを紹介

- ◆ Min-max探索、alpha-beta枝刈り、評価関数
- ◆ 追加機能

★ 簡単なプログラミング

- ◆ 評価関数の調整

★ 五目並べ大会

- ◆ 評価関数の調整効果を確認

五目並べプログラムの調整

- ★ Visual Studioでプロジェクト作成
- ★ 講義用のウェブページに公開されたプログラムを利用し、評価関数の特徴を調整
- ★ 最適な特徴の値を保存し、五目並べ大会をやる

Visual Studioでプロジェクト作成

★ Step 1

◆ Visual Studioを起動

★ Step 2

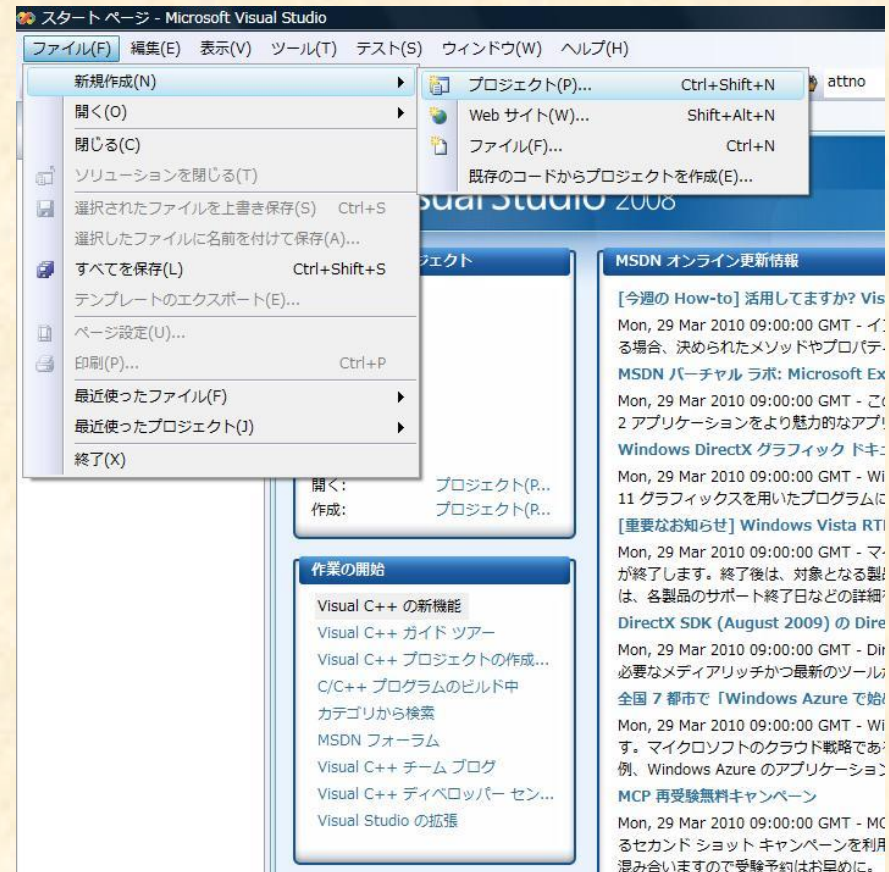
「ファイル」



「新規作成」



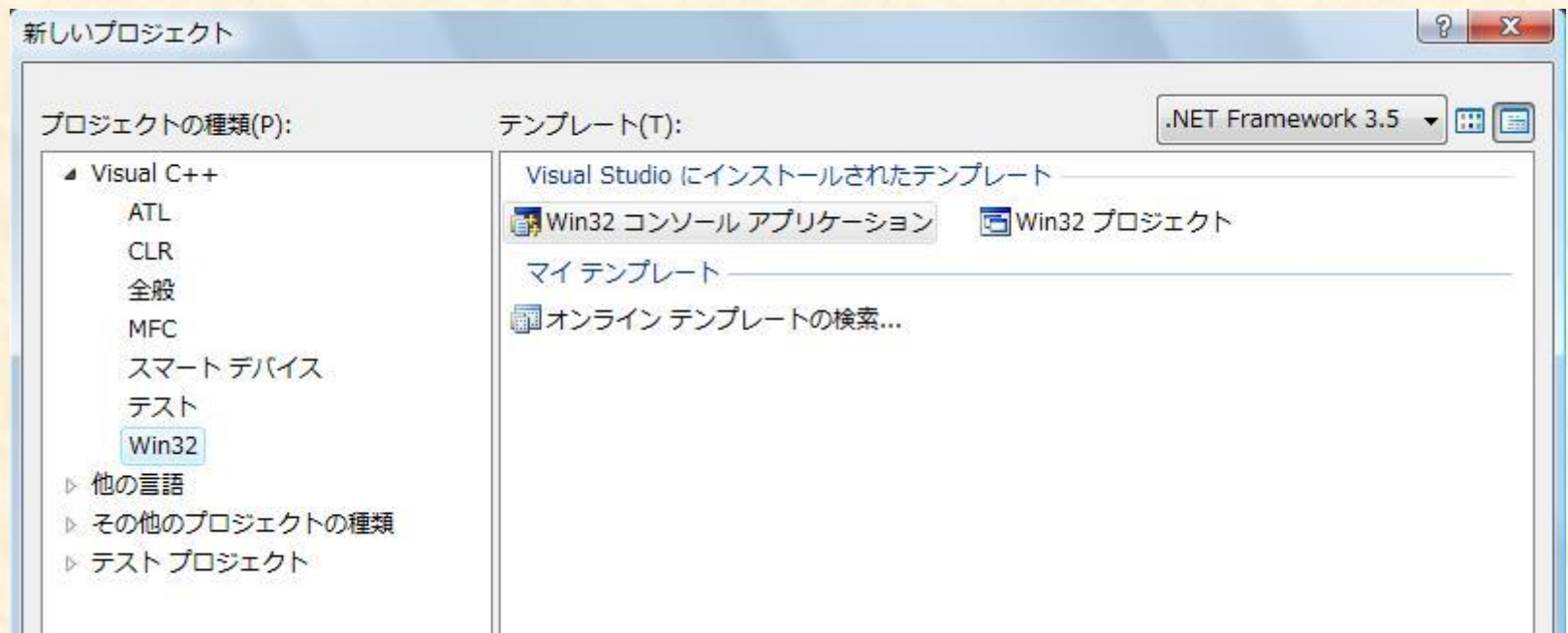
「プロジェクト」



Visual Studioでプロジェクト作成

★ Step 3

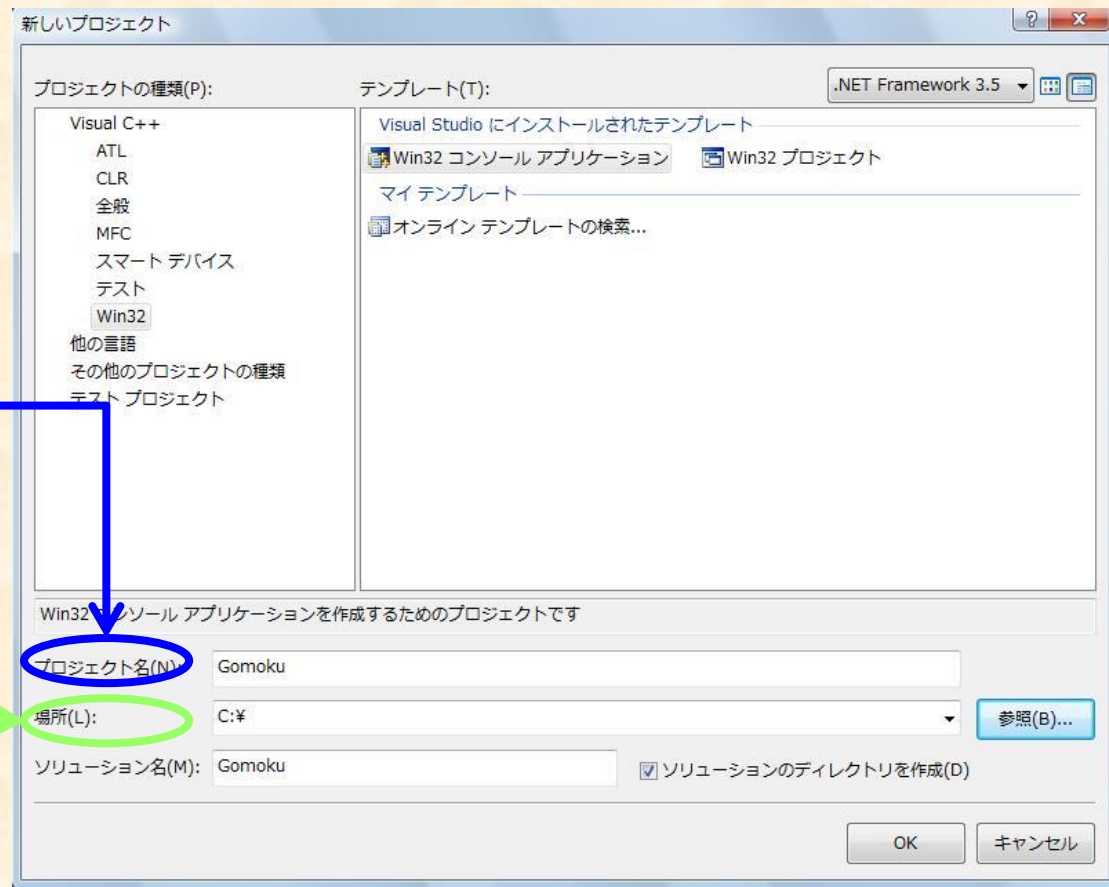
◆ 「Win32」 → 「Win32コンソール アプリケーション」



Visual Studioでプロジェクト作成

★ Step 4

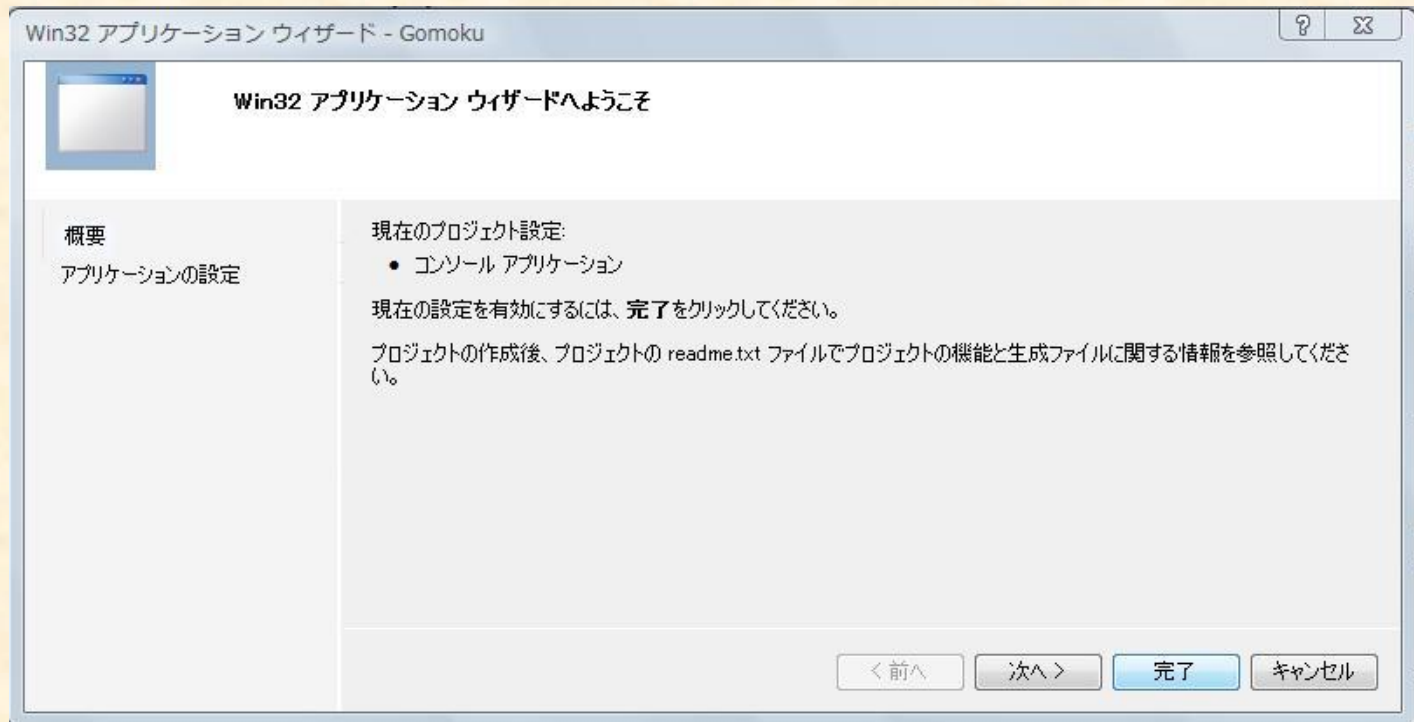
- ◆ プロジェクト名と保存場所を決定



Visual Studioでプロジェクト作成

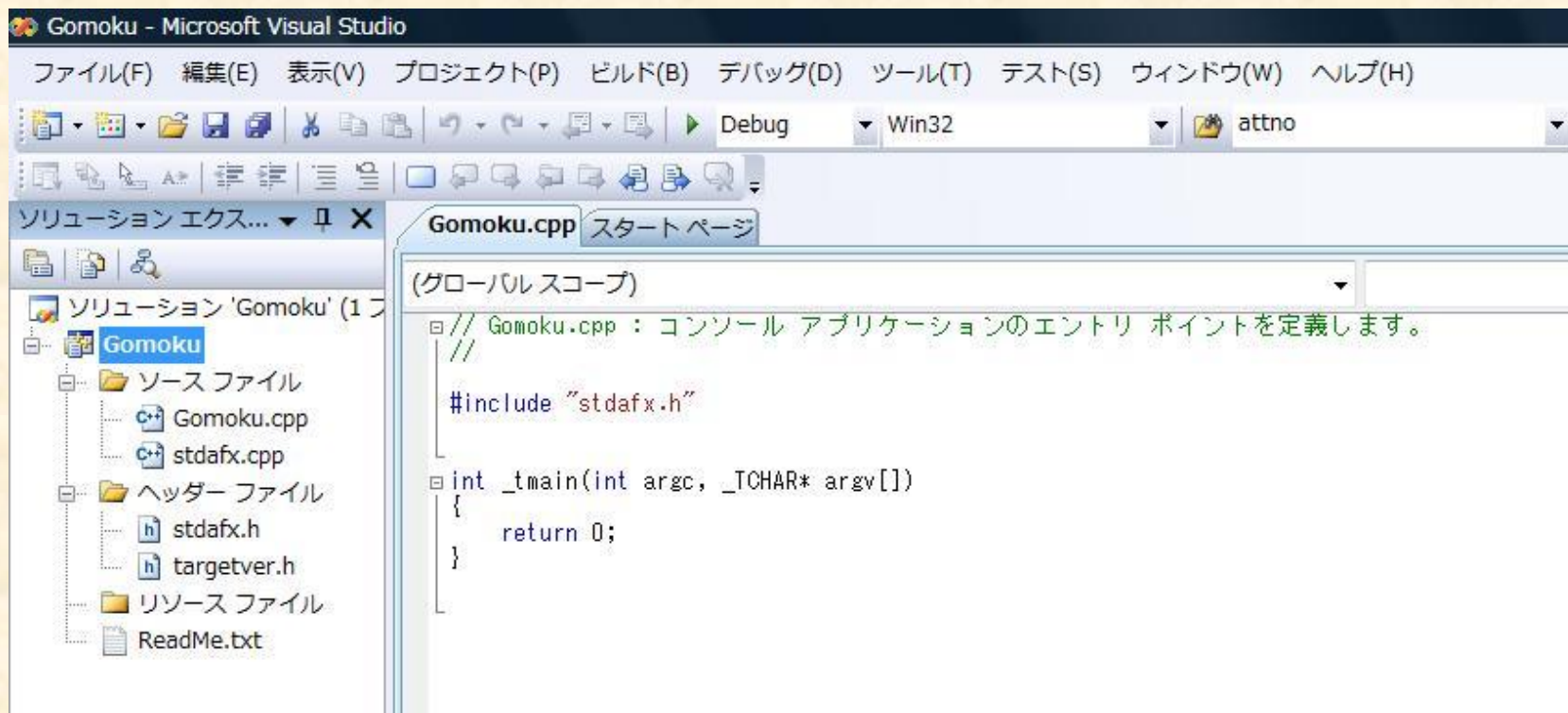
✦ Step 5

◆ 「完了」でプロジェクトを作成



できましたか？

✦ 問題がなければ以下のような画面が見える



The screenshot shows the Microsoft Visual Studio IDE. The title bar reads "Gomoku - Microsoft Visual Studio". The menu bar includes "ファイル(F)", "編集(E)", "表示(V)", "プロジェクト(P)", "ビルド(B)", "デバッグ(D)", "ツール(T)", "テスト(S)", "ウィンドウ(W)", and "ヘルプ(H)". The toolbar shows various icons for file operations and debugging. The status bar at the bottom indicates "Debug", "Win32", and "attno".

The Solution Explorer on the left shows a project named "Gomoku" with the following structure:

- ソリューション 'Gomoku' (1 フォルダ)
- Gomoku
 - ソース ファイル
 - Gomoku.cpp
 - stdafx.cpp
 - ヘッダー ファイル
 - stdafx.h
 - targetver.h
 - リソース ファイル
 - ReadMe.txt

The main editor window displays the code for "Gomoku.cpp" in the "グローバル スコープ" (Global Scope). The code is as follows:

```
// Gomoku.cpp : コンソール アプリケーションのエントリ ポイントを定義します。
//
#include "stdafx.h"

int _tmain(int argc, _TCHAR* argv[])
{
    return 0;
}
```


プログラムを実行

✦ Step 1

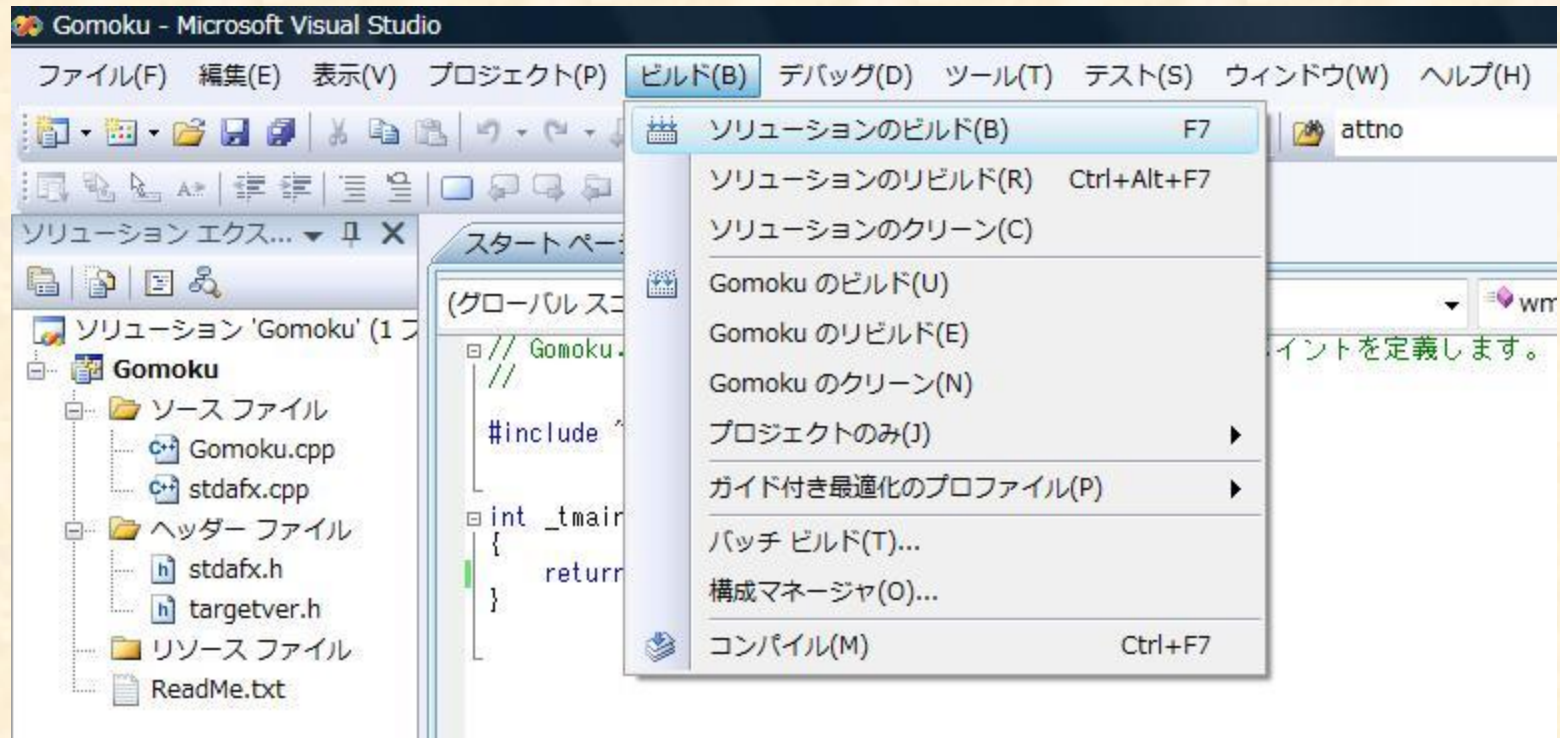
- ◆ 「Debug」のところに「Release」を選択



プログラムを実行

★ Step 2

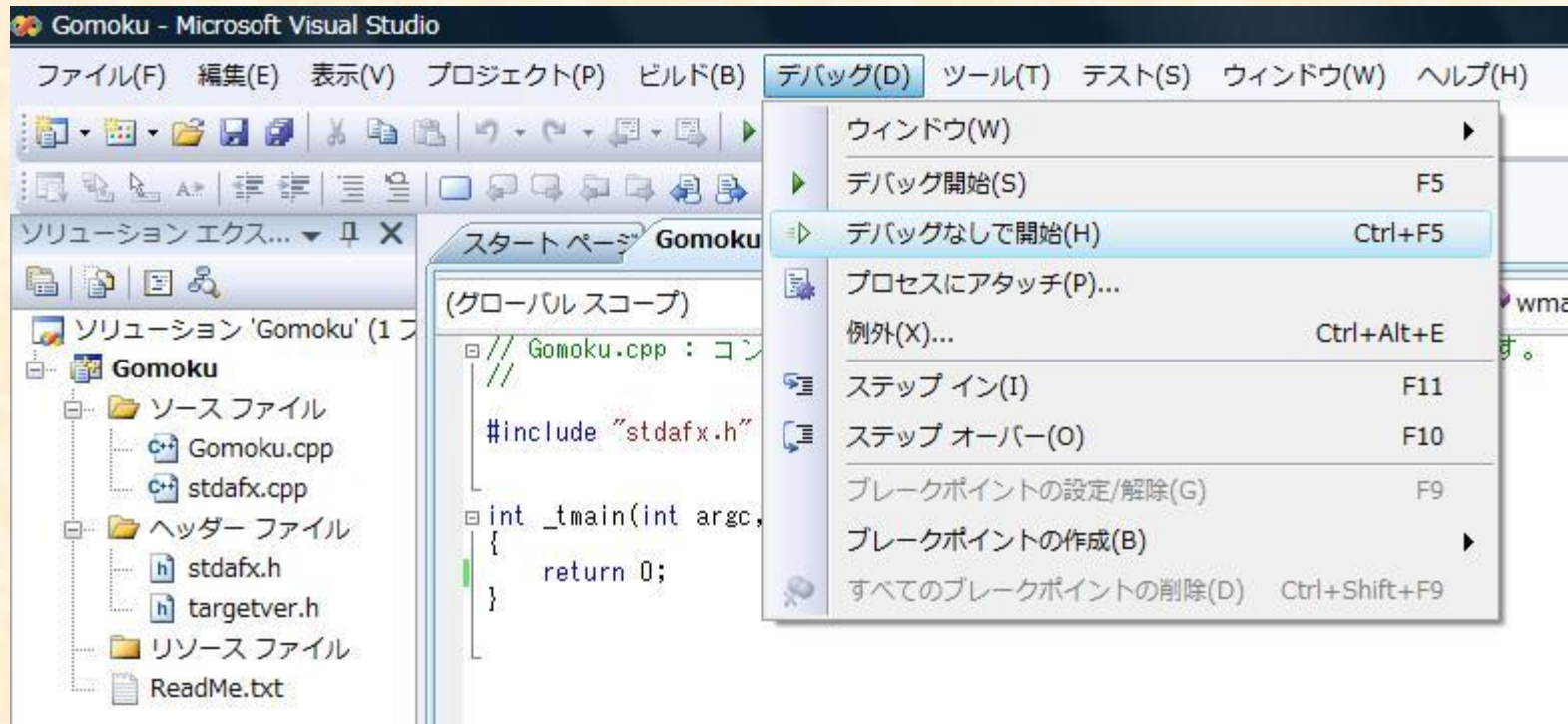
◆ 「ビルド」を選択



プログラムを実行

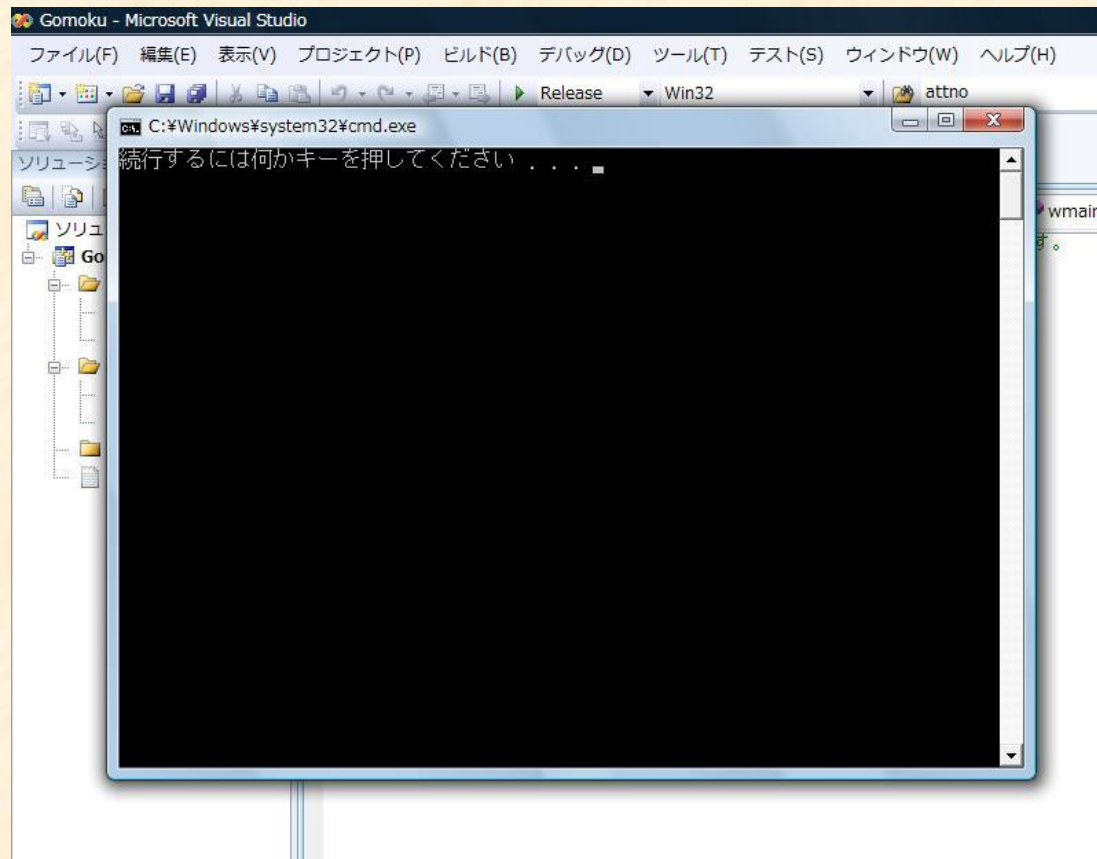
★ Step 3

◆ デバッグなしで開始



できましたか？

✦ 問題がなければ以下のような画面が見える



プログラムをプロジェクトに追加

- ✦ 講義用のウェブページから
GomokuAITester.cをダウンロード
- ✦ 現在のプロジェクトファイルに上書き
 - ◆ Visual StudioでGomokuAITester.cを開く
 - ◆ GomokuAITester.cのコードを全て現在のプロジェクトファイルにコピーとペースト
- ✦ 実行してみよう！

Precompiled Headerよりエラー

✳ Precompiled Headerの設定をオフ

- ◆ 「プロジェクト」メニュー

- プロパティ (Alt+F7でもか)

- Configuration Properties

- C/C++

- Not using Precompiled Headers

- OK

✳ 実行してみよう！

できましたか？

```
C:\Users\Reijer Grimbergen\Documents\Visual Studio 2010\Projects\GomokuEvaluationTest\R...
1 * * * * * * * * * * * * * * *
2 * * * * * * * * * * * * * * *
3 * * * * * * * * * * * * * * *
4 * * * * * * * * * * * * * * *
5 * * * * * * * * * * * * * * *
6 * * * * * * * * * * * * * * *
7 * * * * * * * * * * * * * * *
8 * * * * * * * * * * * * * * *
9 * * * * * * * * * * * * * * *
10 * * * * * * * * * * * * * * *
11 * * * * * * * * * * * * * * *
12 * * * * * * * * * * * * * * *
13 * * * * * * * * * * * * * * *
14 * * * * * * * * * * * * * * *
15 * * * * * * * * * * * * * * *

=====
This is a very weak Gomoku program made by Reijer Grimbergen.
=====
Main menu:
1) Play human against human
2) Play with black against computer
3) Play with white against computer
4) Play computer against computer
Q) Quit program
Please enter your selection: _
```

評価値の特徴を調整

★ ファイルの一番上にあるコードを注目

```
// 現在の最強の評価値
const int bestEvalValues[] = { 10, 10, 10, 10, 10, 10 };
// テストしたい評価値
const int testValues[] = { 10, 10, 10, 10, 10, 10 };

// 評価値配列の中、評価特徴の場所
#define CLOSED_FOUR_VALUE 0 // 1番目の値
#define OPEN_THREE_VALUE 1 // 2番目の値
#define CLOSED_THREE_VALUE 2 // 3番目の値
#define OPEN_TWO_VALUE 3 // 4番目の値
#define CLOSED_TWO_VALUE 4 // 5番目の値
#define INITIATIVE 5 // 6番目の値
```

← 現在の最適な値
← テストしたい値

} 評価関数の特徴
と値の関係


評価値の特徴を調整

★ 評価値の特徴を調整

1. testValuesの値を変更
2. プログラムを実行し、AI同士を対戦させる
3. テスト用のプログラムの勝率が高ければ
bestEvalValuesをtestValuesに上書き
4. 時間がある限り繰り返し

testValuesの値を変更

```
// 現在の最強の評価値  
const int bestEvalValues[] = { 10, 10, 10, 10, 10, 10 };  
// テストしたい評価値  
const int testValues[] = { 15, 14, 12, 10, 9, 5 };
```



妥当と思う値に変更

AI同士を対戦させる

```
C:\Users\Reijer Grimbergen\Documents\Visual Studio 2010\Projects\GomokuEvaluationTest\R...
4 * * * * * * * * * * * * * *
5 * * * * * * * * * * * * * *
6 * * * * * * * * * * * * * *
7 * * * * * * * * * * * * * *
8 * * * * * * * * * * * * * *
9 * * * * * * * * * * * * * *
10 * * * * * * * * * * * * * *
11 * * * * * * * * * * * * * *
12 * * * * * * * * * * * * * *
13 * * * * * * * * * * * * * *
14 * * * * * * * * * * * * * *
15 * * * * * * * * * * * * * *
=====
This is a very weak Gomoku program made by Reijer Grimbergen.
=====
Main menu:
1) Play human against human
2) Play with black against computer
3) Play with white against computer
4) Play computer against computer
Q) Quit program
Please enter your selection: 4
Enter number of games to play: 10
Enter filename to save data: test.txt
```

← AI同士の対戦

← 対局の数と結果保存ファイル

勝率が高ければ上書き

```
C:\Users\Reijer Grimbergen\Documents\Visual Studio 2010\Projects\GomokuEvaluationTest\R...
Best move evaluation: 30000
Computer move: 31) (4,3)
  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15
1  ○ * * * * * * * * * * * * * *
2  * * * * * * * * * * * * * *
3  * * * ● * * * ○ * * * * * * *
4  * * * ● * * ● ● * * * * * * *
5  * * ○ ● ● ● ● ○ * ○ * * * *
6  * * ○ ● ● ● ● ○ * * * * * *
7  * * * ● ○ ○ ○ ○ ● * * * * *
8  * * ○ * * ● * ○ * * * * * *
9  * * * * ○ * ○ ● * * * * * *
10 * * * * * * * * * * * * * *
11 * * * * * * * * * * * * * *
12 * * * * * * * * * * * * * *
13 * * * * * * * * * * * * * *
14 * * * * * * * * * * * * * *
15 * * * * * * * * * * * * * *

Black player wins.

Final result: BestAI - TestAI 3-7(0)

Press any key to continue.
```

```
// 現在の最強の評価値
const int bestEvalValues[] = { 15, 14, 12, 10, 9, 5 };
// テストしたい評価値
const int testValues[] = { 15, 14, 12, 10, 9, 5 };
```

課題：五目並べプログラムを調整

★ テスト用の特徴の値を決めて、テストをしましょう

- ◆ 重要と思われる特徴を高くする
- ◆ 重要ではないと思われる特徴を低くする
- ◆ 対局数は多い方が良い
 - ・ まず10局をやって、結論がしにくい場合に50局など
- ◆ 必要に応じて、テストファイルも確認しましょう

★ 最適な値を保存して、五目並べ大会を優勝しましょう！

五目並べ大会について

- ✦ 五目並べプログラムが決めた手を相手に伝える必要がある所以对戦相手の隣に座る
 - ◆ プログラムの実行ファイルを作成し、フラッシュメモリに保存すること
 - 実行ファイルの場所:ドキュメント→Visual Studio →プロジェクト→プロジェクト名→ Release
 - ◆ 移動する際、実行ファイルを移動先のパソコンに入れること
 - ◆ 混乱しないため、実行ファイル名は「自分の名前」.exeにして下さい