

精密工学科プログラミング基礎Ⅱ

追加資料

習得してほしいこと:

- Visual Studio を用いたプログラミング
 - プロジェクトの作成
 - コンパイルと実行
 - 高機能エディタによる効率的なプログラム編集
 - ブ레이크ポイントの設定によるデバッグ法

資料のURL : <http://www.den.t.u-tokyo.ac.jp/prog>

作業の概要

1. プロジェクトを作成
2. ソースファイル(*.cファイル)を作成
 - 従来と同じ
3. ソースファイルをプロジェクトへ追加
`do{`
4. ソースファイルを編集
5. コンパイルと実行
6. デバッグ
`}while(うまくいかない);`

手順1: プロジェクトの作成(その1)

① Visual Studio 2013 を起動



② ファイル ⇒ 新規作成 ⇒ プロジェクト

③ 下記のようにして「OK」ボタンを押す

The screenshot shows the 'New Project' dialog in Visual Studio 2013. The 'Visual C++' category is selected in the left sidebar. The 'Win32 Console Application' template is highlighted in the center list. The project name is 'kadai1' and the location is 'C:\Users\Ohtake\Desktop\prog1120\'. The 'OK' button is circled in blue. Red arrows point from text boxes to various elements: 'C++を選ぶ' points to the 'Visual C++' category; '好きな名前' points to the project name field; 'チェックを外す' points to the 'Create directory for solution' and 'Add to source control' checkboxes; 'Win32コンソールアプリケーションを選択' points to the selected template; and 'デスクトップに作成したフォルダを選ぶ' points to the location dropdown.

C++を選ぶ

好きな名前

チェックを外す

Win32コンソールアプリケーションを選択

デスクトップに作成したフォルダを選ぶ

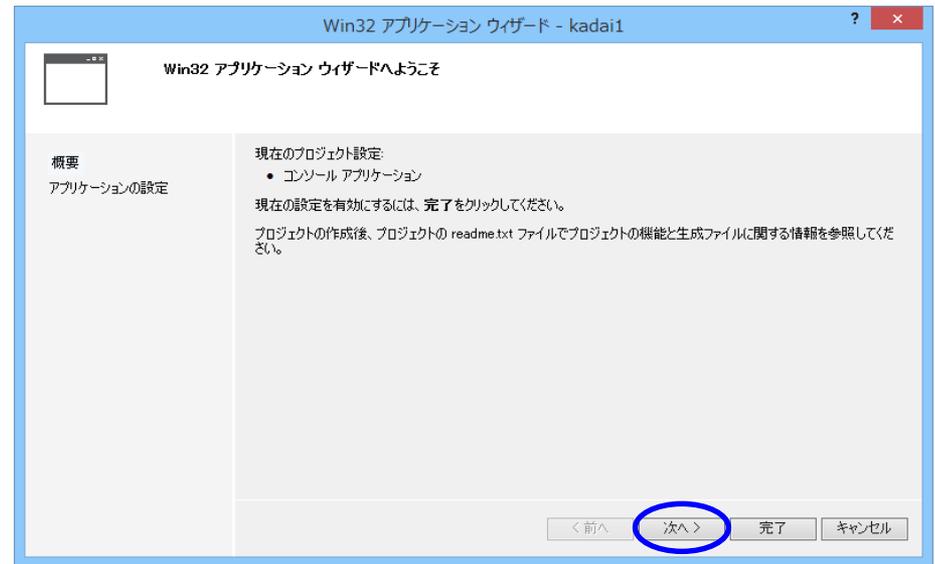
名前(N): kadai1
場所(L): C:\Users\Ohtake\Desktop\prog1120\
ソリューション名(M): kadai1

参照(B)...
 ソリューションのディレクトリを作成(D)
 ソース管理に追加(U)

OK キャンセル

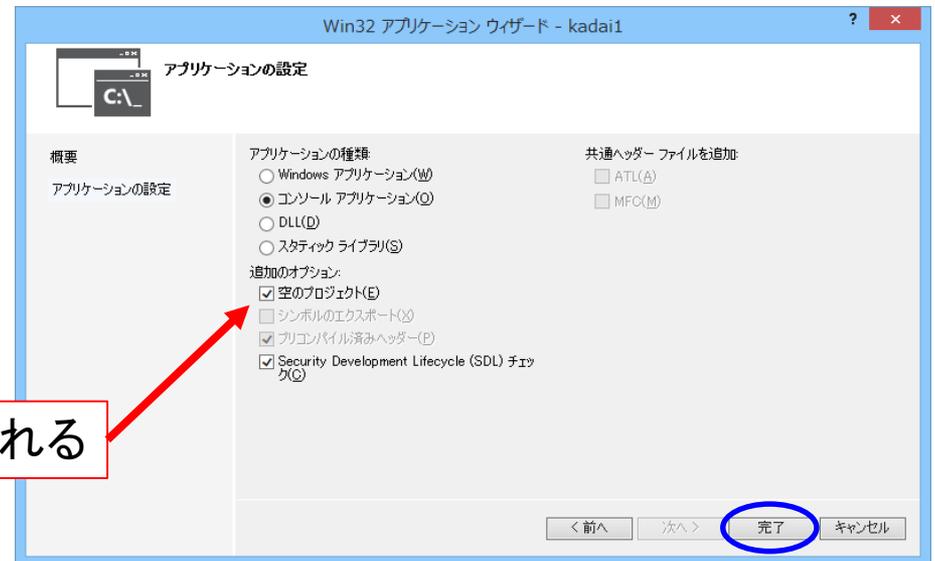
手順1: プロジェクトの作成(その2)

④ 「次へ」ボタン
を押す



⑤ 「空のプロジェクト」
をチェックして、
「完了」ボタン
を押す

チェックを入れる

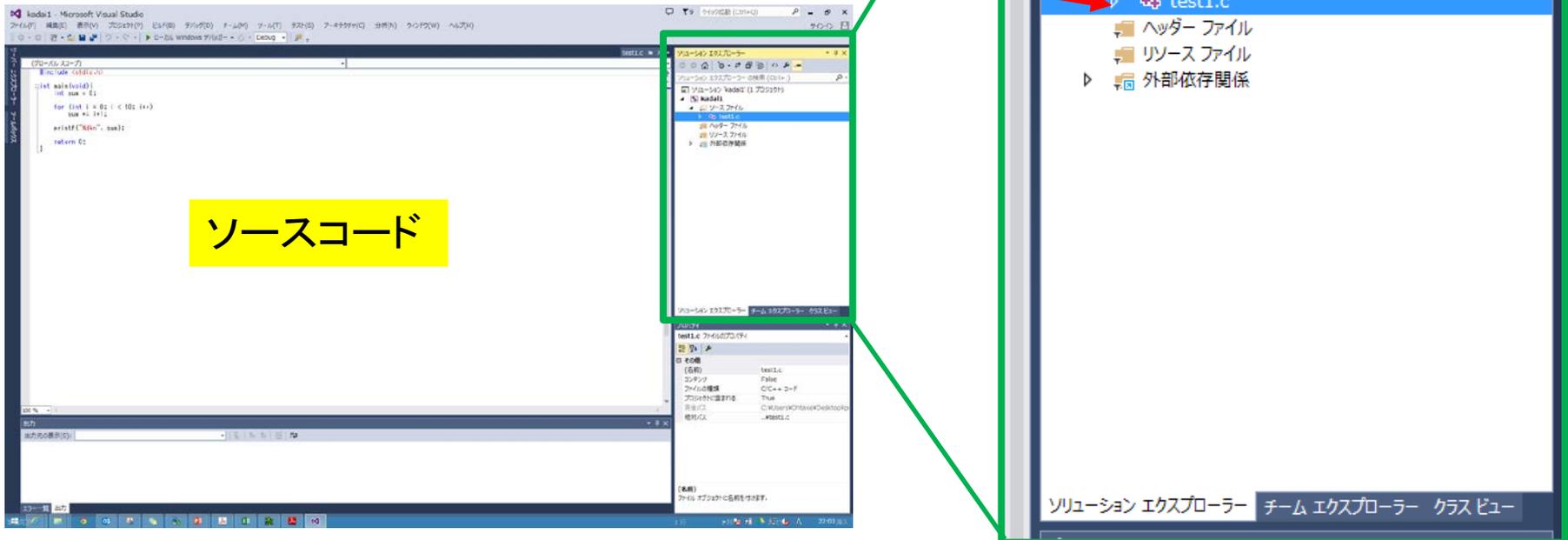


手順2: ソースファイルの作成

- 手順1で作成したプロジェクトのフォルダの中に、好きな名前でも *.c ファイルを作成する。
 - *.txt ファイルをいったん作成して、
拡張子を変更する。

手順3: ソースファイルのプロジェクトへの追加

- ① プロジェクト ⇒ 既存項目の追加
- ② 手順2で作成した*.cファイルを選択
- ③ 右の小窓に*.cが出るので、ダブルクリック



手順4: ソースファイルの編集

- 例えば下記を入力する
 - タイプミスなどは**赤の波線**で警告される(便利!)

```
#include <stdio.h>

int main(void) {
int sum = 0;

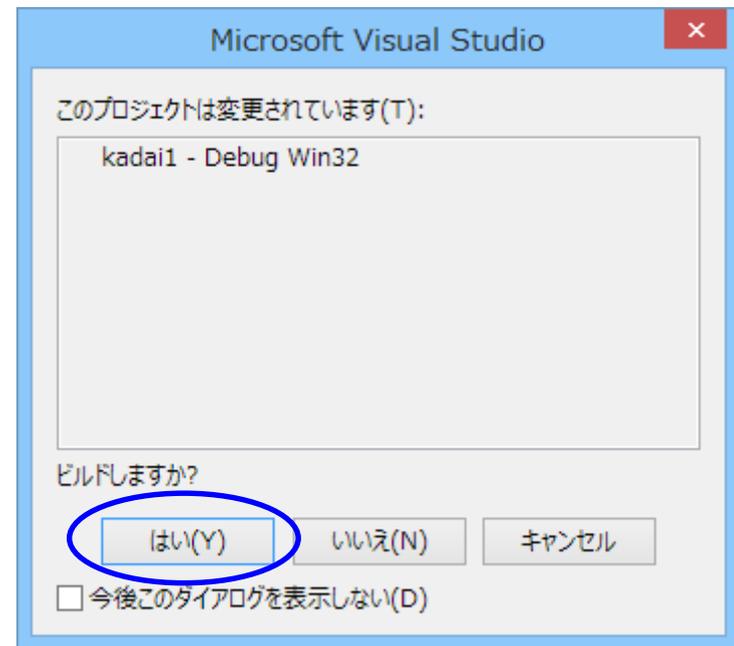
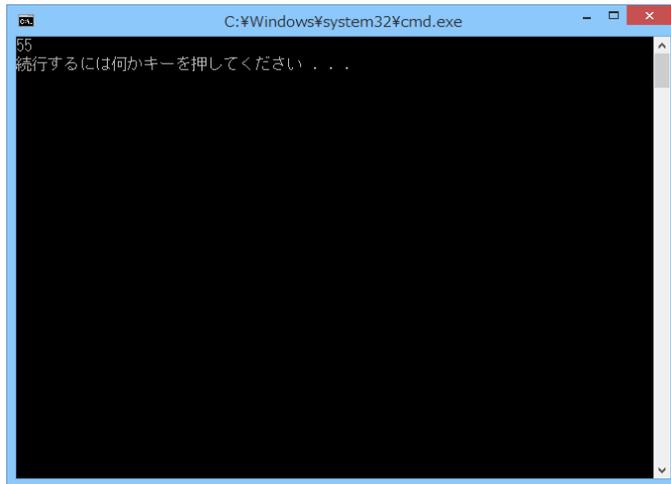
for (int i = 0; i < 10; i++)
sum += i+1;

printf("%d¥n", sum);

return 0;
}
```

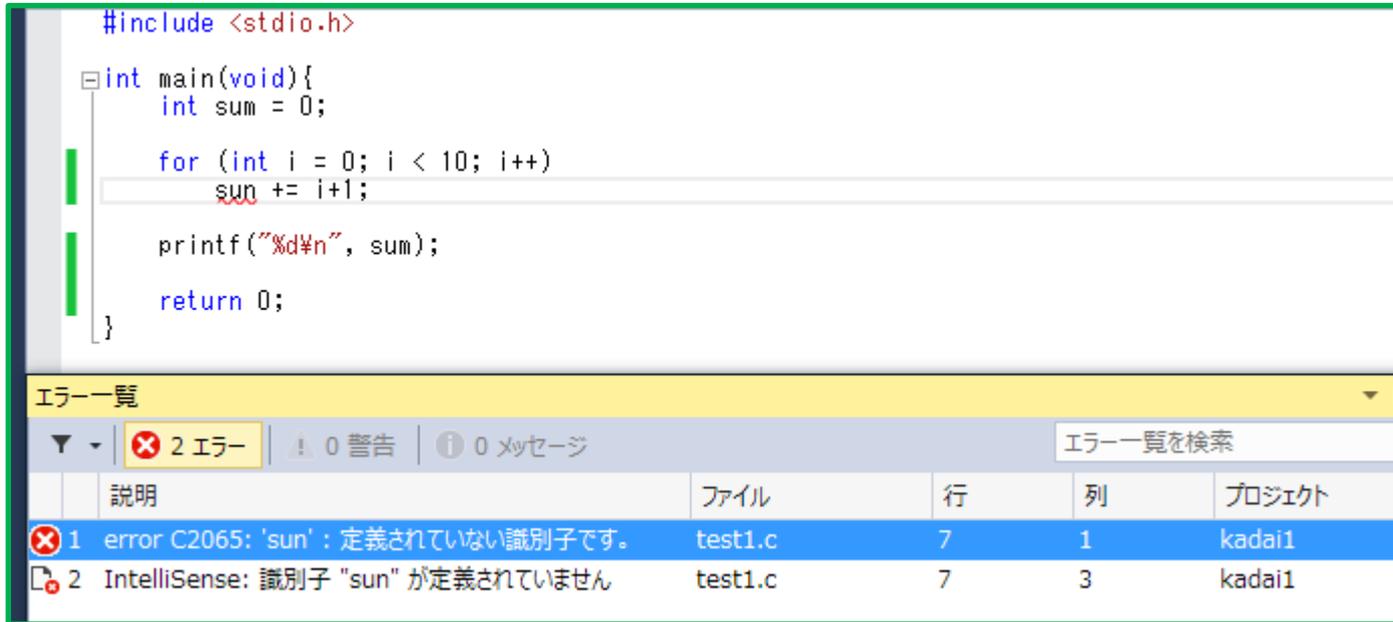
手順5: コンパイルと実行

- ① デバッグ ⇒ デバッグなしで開始
– Ctrl + F5 キーがショートカット
- ② ビルド (=コンパイル) するか聞かれるので、「はい」を選択
- ③ 結果が表示される



手順5(追加): コンパイルエラー

- 表示 ⇒ エラー一覧
 - エラーが表形式で表示されるので便利
 - エラー文をダブルクリックすると、その箇所にジャンプする。



```
#include <stdio.h>

int main(void){
    int sum = 0;

    for (int i = 0; i < 10; i++){
        sun += i+1;

        printf("%d\n", sum);

        return 0;
    }
}
```

エラー一覧

▼ 2 エラー | 0 警告 | 0 メッセージ | エラー一覧を検索

	説明	ファイル	行	列	プロジェクト
✖ 1	error C2065: 'sun': 定義されていない識別子です。	test1.c	7	1	kadai1
🔍 2	IntelliSense: 識別子 "sun" が定義されていません	test1.c	7	3	kadai1

手順6: デバッグ(その1)

- プログラムを好きなところで一時停止させて、変数の値を確認できる。
 - これが Visual Studio を使う一番のご利益です

ソースコードの
左端をクリックして
一時停止する箇所
(ブレイクポイント)
を設定

```
#include <stdio.h>

int main(void){
    int sum = 0;

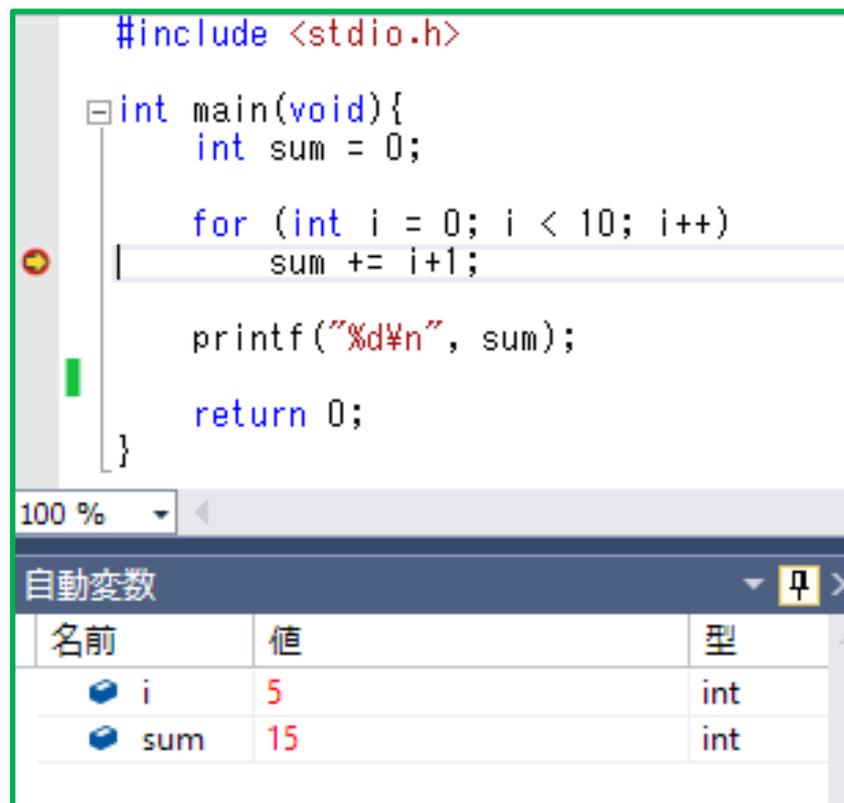
    for (int i = 0; i < 10; i++)
        sum += i+1;

    printf("%d\n", sum);

    return 0;
}
```

手順6: デバッグ(その2)

- デバッグ ⇒ デバッグ開始
 - F5 キーがショートカット
- ブレイクポイントで一時停止し、自動変数ウィンドウで変数の値が見れる
- F5キーを押すと次に進む
 - 画面は5回目



```
#include <stdio.h>

int main(void){
    int sum = 0;

    for (int i = 0; i < 10; i++){
        sum += i+1;
    }

    printf("%d\n", sum);

    return 0;
}
```

自動変数

名前	値	型
i	5	int
sum	15	int

Visual Studio に関する諸注意

- 引き続き Cygwin で進めても問題ありません
 - 比較的長いプログラムを作成するときに有効
- バックアップ時にはデータ量節約のために、
 - * .sdf ファイルは消しても問題ありません
 - 変数の辞書的なファイル
- * .exe ファイルは Debug フォルダの中
- 計算速度を上げたい場合
には Release モードを選択

