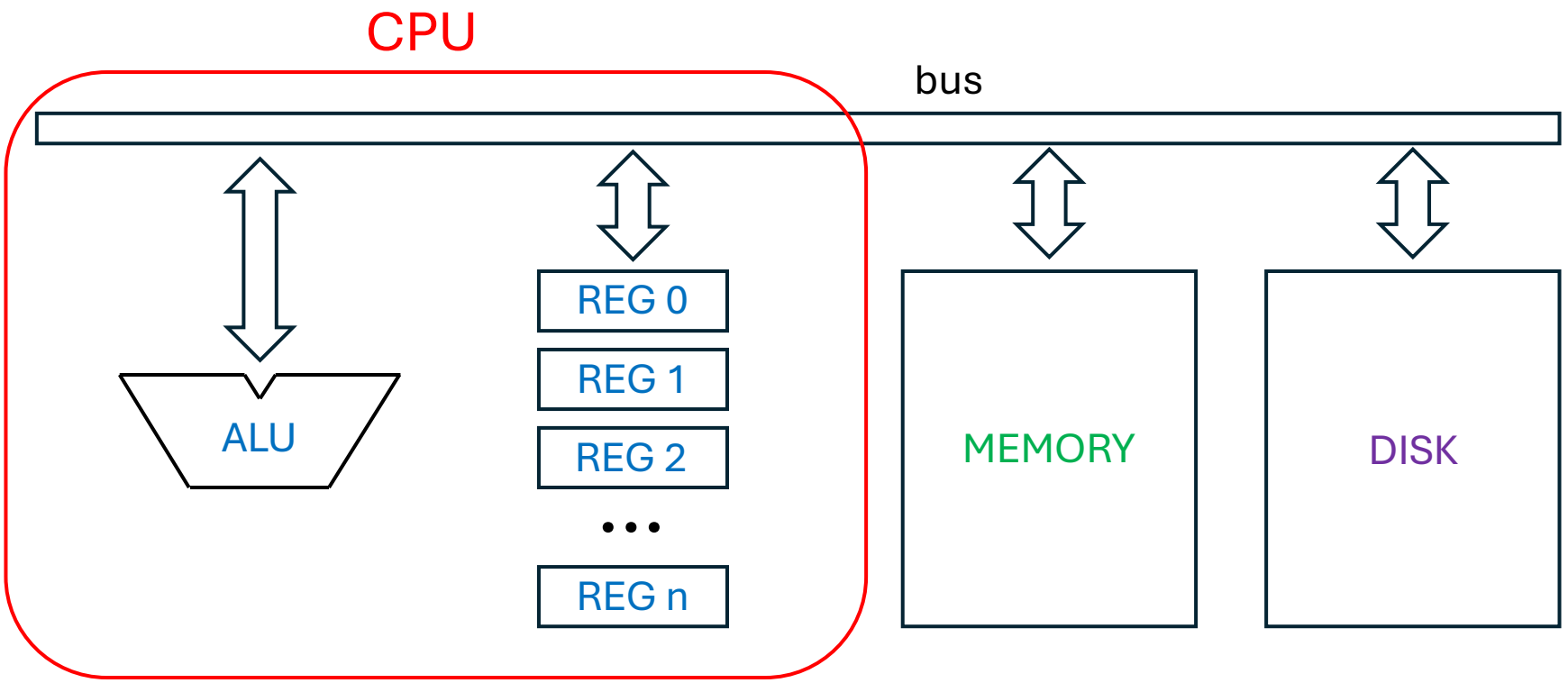
C/C++プログラミング

第1回

コンパイラ言語とインタープリタ言語

コンピュータでプログラムが実行されるしくみ



一般的なコンピュータの処理構造

コンピュータでプログラムが実行されるしくみ

CPU (central processing unit): 中央処理装置
以下のALU，レジスタや処理を行う回路が組み込まれたもの．コンピュータの中枢部．

- ALU (Arithmetic and Logic Unit): 算術論理演算装置
  レジスタに保持されているビット列の和，論理和，論理積などの計算・演算を行い，その結果をレジスタに格納する．
- レジスタ(register)
  ALUで処理される・処理されたビット列のデータ・命令を一時的に保持する格納場所

コンピュータでプログラムが実行されるしくみ

メモリ(memory)
データやアルゴリズムなどが一時的に保存される格納場所. アルゴリズムの指令によって，メモリ内の特定の場所にあるデータがレジスタに転送される．また逆に処理結果のレジスタ内のデータがメモリ内の特定の場所に転送され保存される.

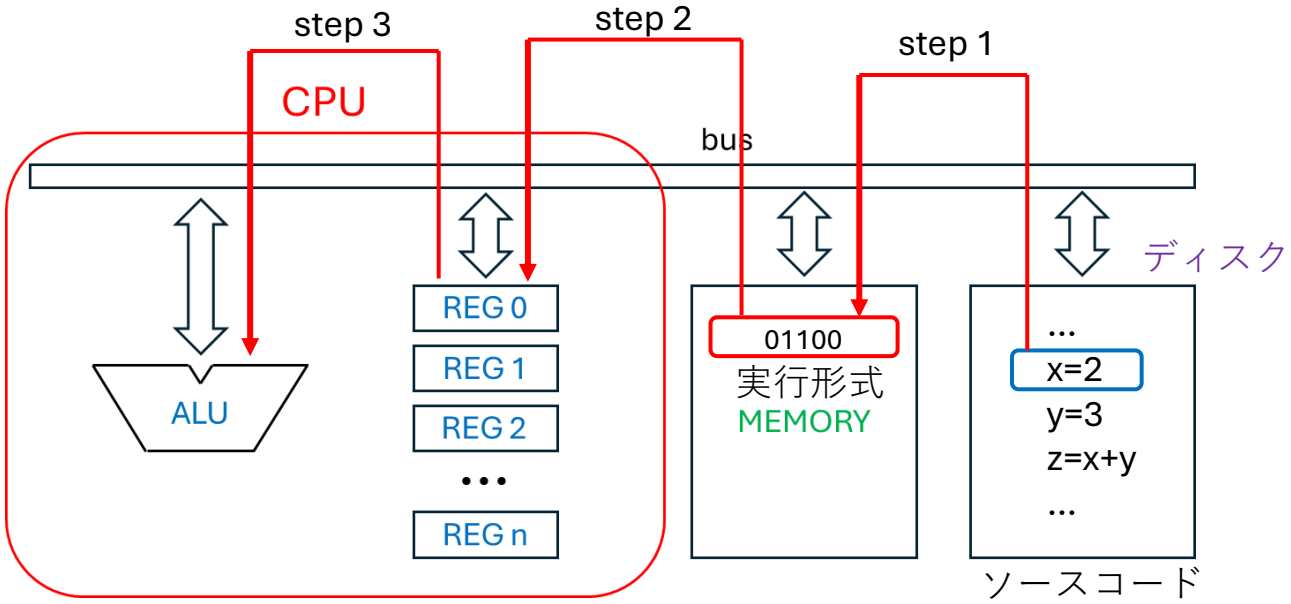
ディスク (disk)
HDD, SSDなどであり，プログラムのソースコードやその実行形式，データなどが長期間保存される格納場所．必要に応じてディスク内のデータ・プログラムの実行形式がメモリに転送され，逆に計算結果のデータがメモリからディスクに返送される.

バス (bus)
データやプログラムが転送される際の伝送路

コンパイラ言語とインタプリタ言語

インタプリタ言語：
人が書いたプログラムのソースコードを1行ずつ解釈し処理していくプログラミング言語.　　例：Python, basic, JavaScript, ...

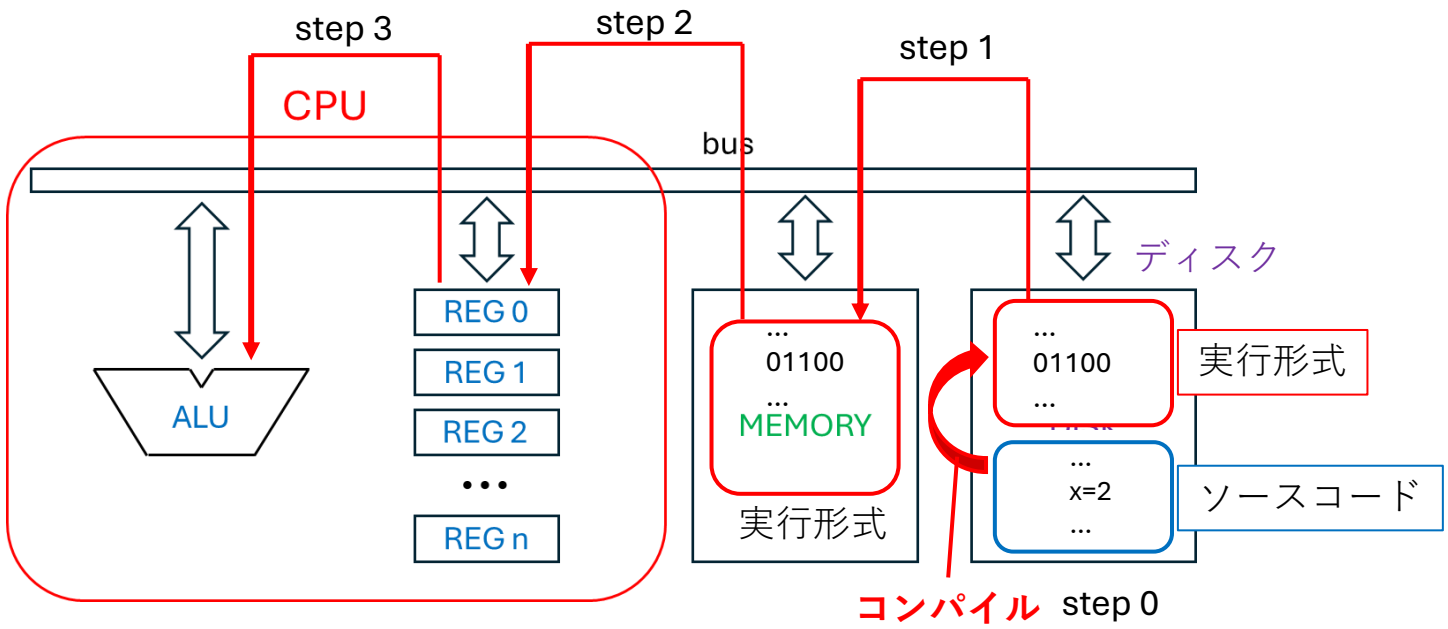
ソースコードはコンピュータのディスク上に保存されたプログラムであり，インタプリタはそのソースコードを1行ずつ解釈し，処理手続きやデータをメモリ上に格納する．次にメモリ上の処理手続きに従いCPUで処理が実行される．処理結果は逆方向に転送される．コンパイルが不要だが動作は遅い.



コンパイラ言語とインタプリタ言語

コンパイラ言語：
人が書いたプログラムのソースコード全体をコンパイラが一括でCPUが解釈できる動作内容（機械語）に変換する（「コンパイルする」という）．変換された形式のプログラムを「実行形式」という．その実行形式のプログラムに従って動作するプログラミング言語． 例：C, C++, Java, ...

ソースコードとその実行形式はコンピュータのディスク上に保存されている．プログラムの実行時には実行形式の中の必要部分がメモリ上に格納され，次にメモリ上の実行形式の処理内容に従いCPUで処理が実行される．動作が速い．



C, C++ 言語

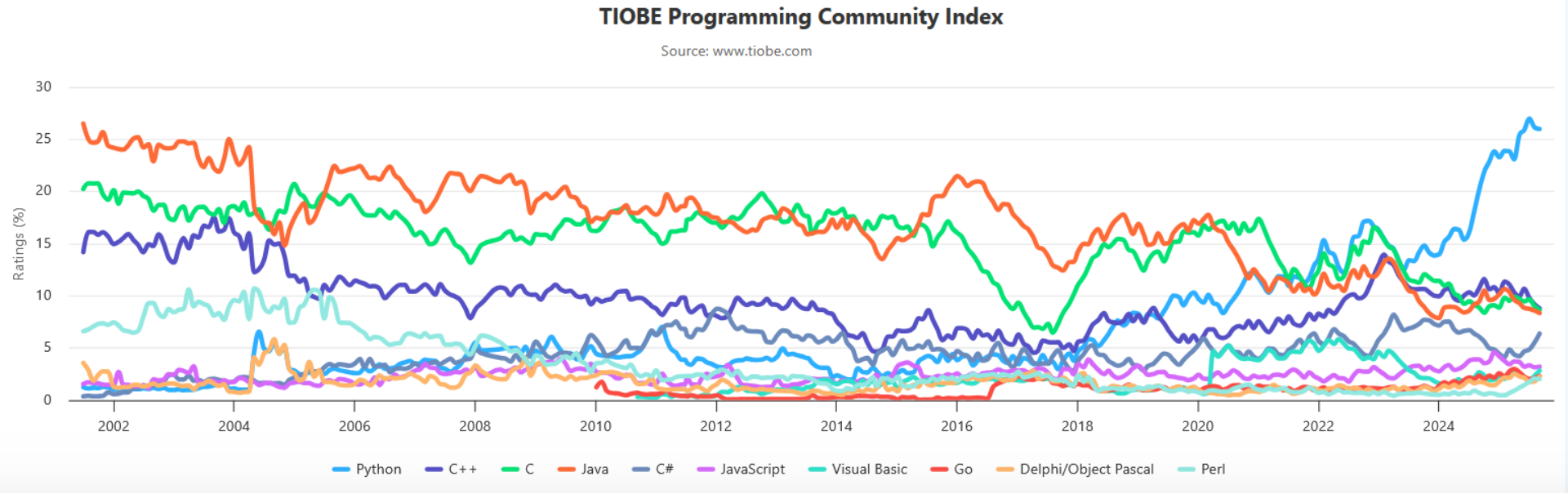
C言語 (C Language, C)：

1972年，AT&Tベル研究所，Dennis MacAlistair Ritchie が開発．OS である UNIX を開発するために開発されたプログラミング言語．手続き型言語．高水準言語（高級言語，人が理解しやすい言語）でありかつハードウェアを直接扱う低水準言語の性質をも併せ持つ．動作が高速で汎用性が高く広く用いられている．特にOSや組み込み系に使われる．

C++：

1983年，AT&Tベル研究所，Bjarne Stroustrupが開発．C言語をベースに，オブジェクト指向（複数の「オブジェクト」の集まりとしてプログラムを構築する．「オブジェクト」とはデータとその操作がまとまったもの．）の機能を持った言語として開発された．

プログラミング言語の人気度



https://www.tiobe.com/tiobe-index/

コンパイラ言語のプログラミングと実行までの手順

1．エディタでソースコードを作成する．
2．ソースコードをコンパイルし実行形式に変換する．
3．実行形式を実行する．

以上の作業を進めるには，
「エディタ」，「コンパイラ」
が必要．

Visual Studio Code (VSCode)：エディタと，コンパイル・実行の作業を補助するプログラム開発環境．C/C++ 以外にも多くのプログラム言語に対応している．

MinGW：gnu (OSおよびオープンソフトウェア開発のプロジェクト)で開発された C/C++ のコンパイラ gcc, g++ および関連するツールからなる開発環境．VSCodeの中で gcc, g++が使われる．

プログラミング環境のセッティング

1. MinGW (C/C++のコンパイラ一式)をインストールする.
   (Apple の場合は不要，次の 2 にスキップ)

2. VSCode (エディタ等)をインストールする.

MinGW (C/C++のコンパイラ一式)のインストール

Windows の場合
“C/C++をVSCodeで開発するための環境構築”などを参考にする
https://qiita.com/ochx/items/01449d09777187790ee4

1. すでにmingw64をインストールしているなら以下の作業は不要．VSCodeのインストールに進む.

2. Windows が何ビットのOSかを確認する.
https://qiita.com/ochx/items/cc56b6f06fcee03dad96
-> ほぼ64bitであろう

3. MinGWの7zipファイルをダウンロードする
https://github.com/niXman/mingw-builds-binaries/releases/download/15.2.0-rt_v13-rev0/x86_64-15.2.0-release-win32-seh-ucrt-rt_v13-rev0.7z

MinGW (C/C++のコンパイラ一式)のインストール

4. ダウンロードしたファイルを解凍する．解凍方法は
ファイル名を右クリックして「解凍」or「すべて展開」を
選ぶ．(解凍に数分かかる)

5. フォルダ“c:\tools”を作成し解凍されたフォルダ
“mingw64”を“c:\ tools\”の下へ移動する．

6. PATHを設定する．
Windowsの「設定」→「システム」→「バージョン情報」→
「システムの詳細情報」→「環境変数」→「***のユーザ環
境変数」の中の「Path」を選び「編集」→「新規」→
「C:\ tools\mingw64\bin」を入力する(カギカッコ内の文字
列)→「OK」→「OK」→「OK」

MinGW (C/C++のコンパイラ一式)のインストール

7. Windows のコマンドプロンプト (cmd) を起動し(Windowsの窓のアイコンをクリック→検索窓でcmdと入力してreturn), “gcc –v” と入力し Return する.
表示される文字列の最後が,
“(X86_64-win32-she-rev0, Built by MinGW-Builds project)”
となっていれば完了.

VSCode のインストールとセッティング

1. VSCode のHP を開く(VSCodeが既にインストールされている場合は5までスキップする)

https://code.visualstudio.com/

2. “Download for Windows” をクリック(Windowsの場合).

3. ダウンロードしたファイル
“VSCodeUserSetup-x64-1.104.1” (20250919現在)
をダブルクリック.

4. 「使用許諾契約書の同意」の画面で「同意する」をチェックし「次へ」→インストール先指定で「次へ」→スタートメニューフォルダーの設定で「次へ」→追加タスクの選択で「デスクトップ上にアイコンを作成する」にチェックを入れ，「その他」はデフォルトのまま「次へ」→「インストール」→「完了」→VSCode起動 (“Setup VS Code” の画面は閉じる)

VSCode のインストールとセッティング

5. Extension の検索バーで「C/C++ Extension Pack v1.3.1」(Microsoft版, 紫色のアイコン, 46.1M)を探し同様にインストールする．「プレリリース版をインストールするか？」と問われたら「しない」を選ぶ．

6. Extension の検索バーで「Code Runner」を探す．表示された一覧から「Code Runner」をクリック→「発行元を信頼してインストール」

7. Extension バーの Code Runner の歯車アイコンをクリック→「設定」→「Code-runner: Run in Terminal」を探し，チェックボックスにチェックを入れる．一旦 VSCode をkillし(右上のXをクリック)，再度立ち上げる．

VSCode のインストールとセッティング

8. 左上の「ファイル」→「新しいテキストファイル」→「言語の選択」→「C」→次のソースコードを打ち込む.

```
#include <stdio.h>

void main(void){
  printf("Hello World");
}
```

---

9. 右上の 三角マーク(Run Code) をクリック

10. 下部の「ターミナル」に幾つかの文字列と
“Hello World” が表示されたらコンパイル＆実行が成功

11. 「ファイル」→「名前を付けて保存」→「デスクトップ」→「新しいフォルダー」→好きな名前(例えばc++)のフォルダを作成→「c++」をクリック→ファイル名に好きな名前(例えばtest1.c)をつけて「保存」

VSCode のインストールとセッティング

12. 左上の「ファイル」→「新しいテキストファイル」→「言語の選択」→「C++」→次のソースコードを打ち込む.

```
#include <iostream>
using namespace std;

int main(){
  cout << “Hello World” << endl;
}
```

---

13. 右上の 三角マーク(Run Code) をクリック

14. 下部の「ターミナル」に幾つかの文字列と
“Hello World” が表示されたらコンパイル＆実行が成功

15. 11と同様にして好きな名前(例えばtest2.cpp)をつけて「保存」

VSCode のインストールとセッティング

16. Windows のコマンドプロンプト (cmd) を起動しc++ のディレクトリ(c:\Users***\Desktop\c++)でソースファイル test1.c のコンパイル

gcc –o test1m test1.c

を実行し，実行ファイル test1m.exe が生成されているのを確認する．gcc はCのコンパイラ．

17. コマンドプロンプトで test1m を入力 & return し，

Hello World

が表示されるか確認する．

VSCode のインストールとセッティング

18. コマンドプロンプトで

g++ -o test2m test2.cpp

を実行し，test2m.exe が生成されているのを確認する．
g++ はC++のコンパイラ．

19. コマンドプロンプトで test2m を入力 & return し，

Hello World

が表示されるか確認する．

【注意】VSCode で三角マークを押すとなにが実行されるのかというと，上のコンパイルと生成された実行ファイルの実行が自動的に実行されているということ．

課題

(1) 本日作成したtest1.c のソースコードに含まれる
"Hello World"
の文字列を、次の内容に書き換えること。
"C program: Hello World by [your name](1234567890)"
※ [your name] には自分の英語表記の名前を、括弧内には学籍番号を入れること。
(2) 本日作成したtest2.cpp のソースコードに含まれる
"Hello World"
の文字列を、次の内容に書き換えること。
"C++ program: Hello World by [your name](1234567890)"

修正後のプログラムを実行し、その **実行結果のスクリーンショット** を保存し、UNIPAに2枚のスクリーンショットファイル、または2枚のスクリーンショットを含む1つのPDFファイルを提出すること。

スクリーンショットファイルの保存方法

- Windowsの場合
  - **範囲を指定して**：Windows + Shift + S → 範囲選択 → クリップボードにコピーされる。
  - **保存方法**：PowerPoint ➔ 新規スライド ➔ 貼り付け ➔ [名前を付けて保存] ➔ PDF形式で保存。
- Macの場合
  - **範囲を指定して保存**：Command + Shift + 4 → 十字カーソルで範囲を選択 ➔ デスクトップに保存。