

発展プログラミング

第6回：計算機アーキテクチャとスレッドの導入

宮田章裕 <miyata.akihiro@nihon-u.ac.jp>

計算機アーキテクチャ

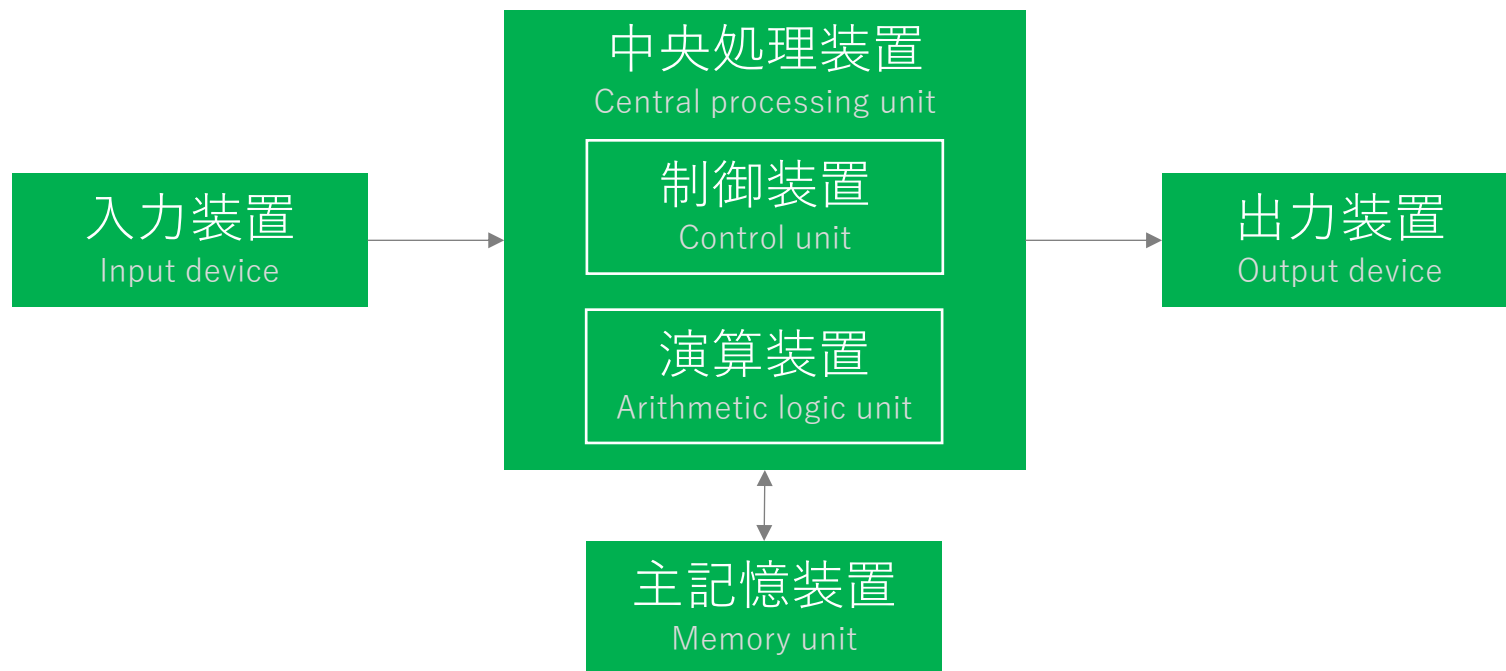


計算機アーキテクチャ (1/4)

Computer architecture

❖ 一般的な計算機を構成する主要装置

- ❖ 制御装置 : 演算, 記憶装置読み書き, 入出力などを制御する
- ❖ 演算装置 : 論理演算, 四則演算などの演算を行う
- ❖ 主記憶装置 : データを記憶する
- ❖ 入力装置 : データを入力する
- ❖ 出力装置 : データを出力する





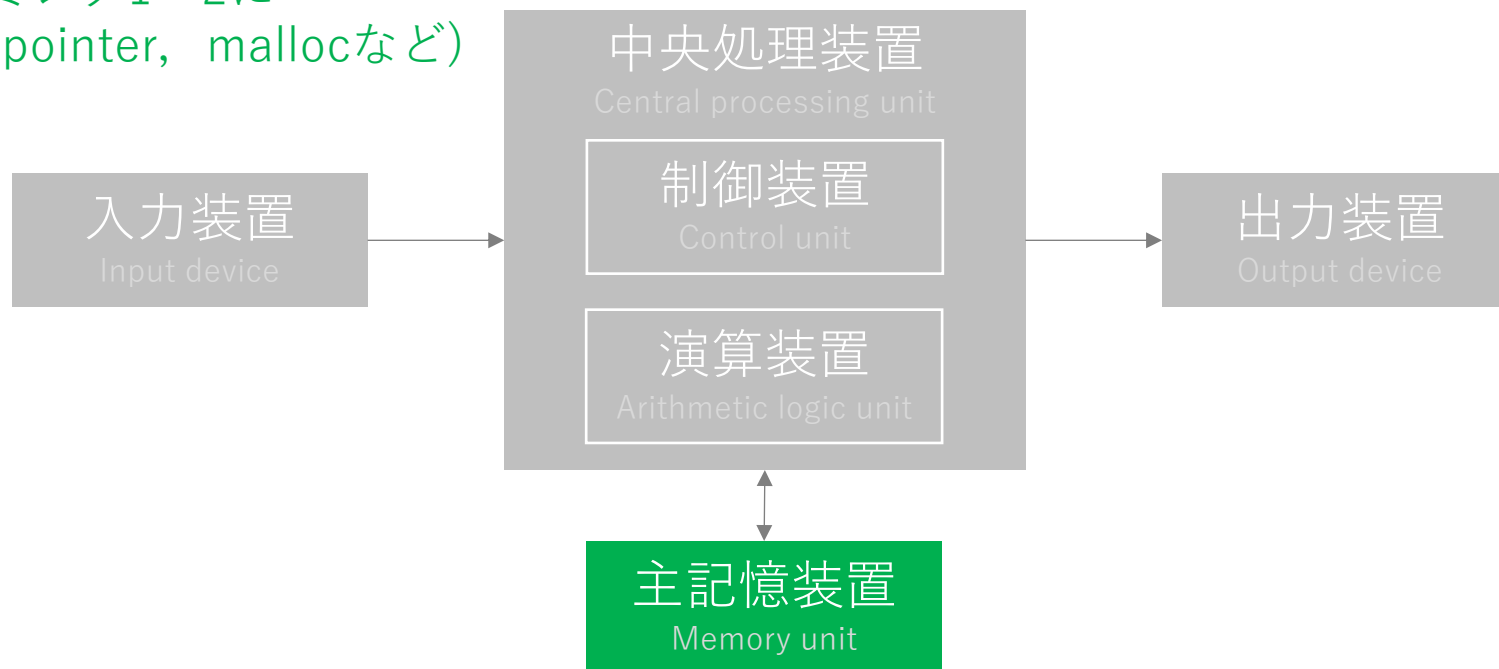
計算機アーキテクチャ (2/4)

Computer architecture

❖ 一般的な計算機を構成する主要装置

- ❖ 制御装置 : 演算, 記憶装置読み書き, 入出力などを制御する
- ❖ 演算装置 : 論理演算, 四則演算などの演算を行う
- ❖ 主記憶装置 : データを記憶する
- ❖ 入力装置 : データを入力する
- ❖ 出力装置 : データを出力する

実践プログラミング1・2に
関係する装置(pointer, mallocなど)





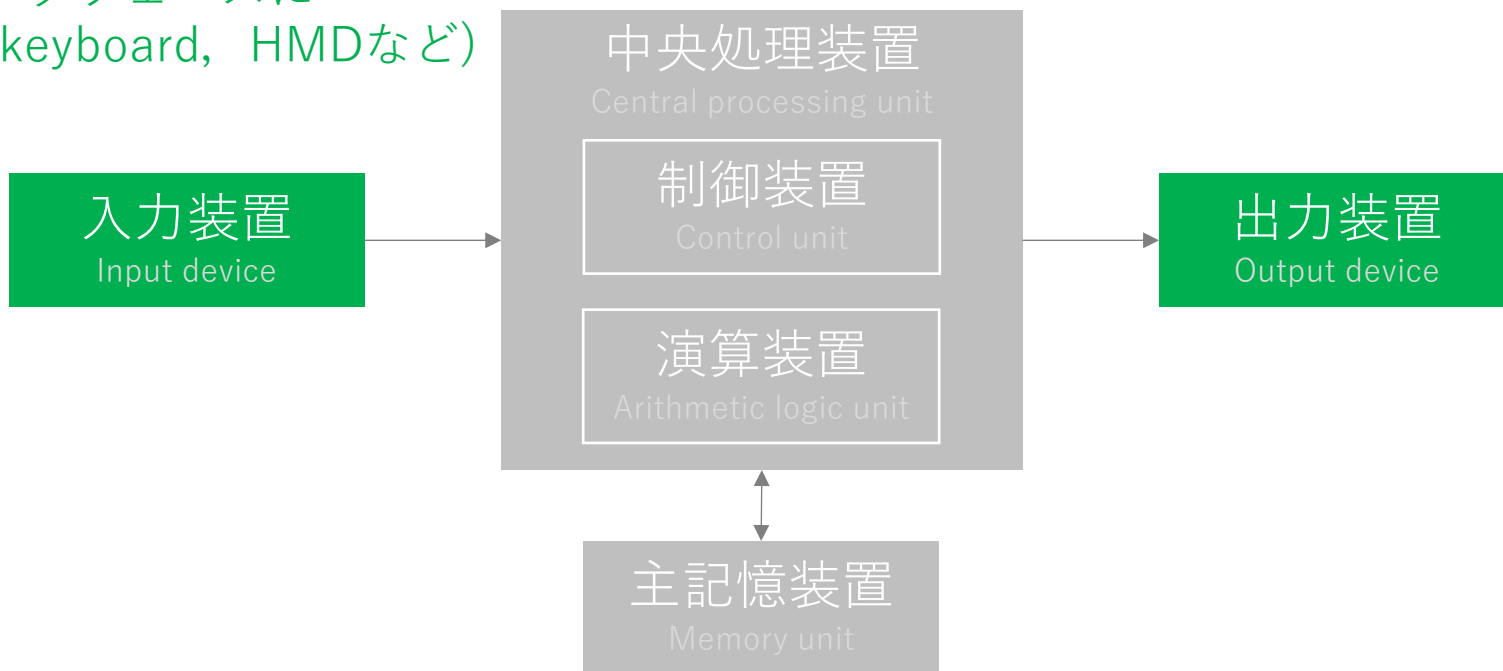
計算機アーキテクチャ (3/4)

Computer architecture

❖ 一般的な計算機を構成する主要装置

- ❖ 制御装置 : 演算, 記憶装置読み書き, 入出力などを制御する
- ❖ 演算装置 : 論理演算, 四則演算などの演算を行う
- ❖ 主記憶装置 : データを記憶する
- ❖ 入力装置 : データを入力する
- ❖ 出力装置 : データを出力する

ヒューマンインタフェースに
関係する装置(keyboard, HMDなど)





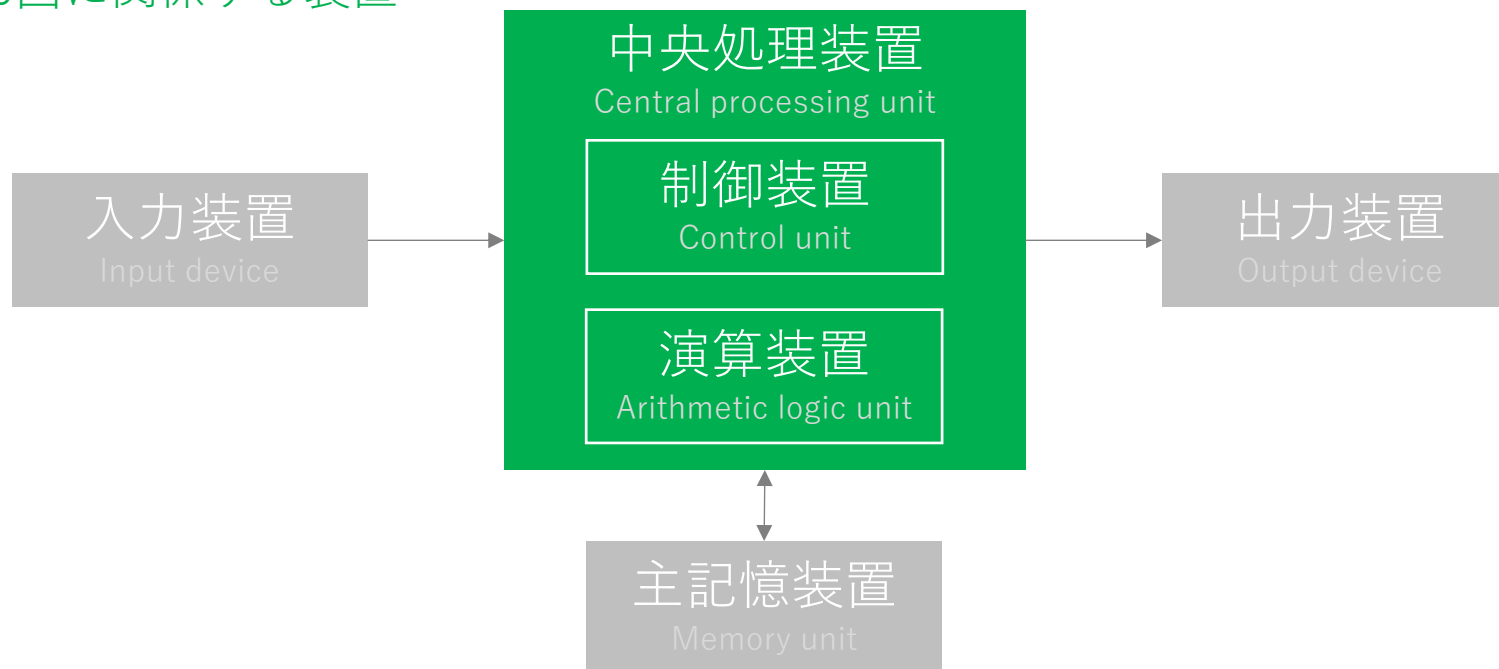
計算機アーキテクチャ (4/4)

Computer architecture

❖ 一般的な計算機を構成する主要装置

- ❖ 制御装置 : 演算, 記憶装置読み書き, 入出力などを制御する
- ❖ 演算装置 : 論理演算, 四則演算などの演算を行う
- ❖ 主記憶装置 : データを記憶する
- ❖ 入力装置 : データを入力する
- ❖ 出力装置 : データを出力する

本講義第6～10回に関する装置





中央処理装置 (1/3)

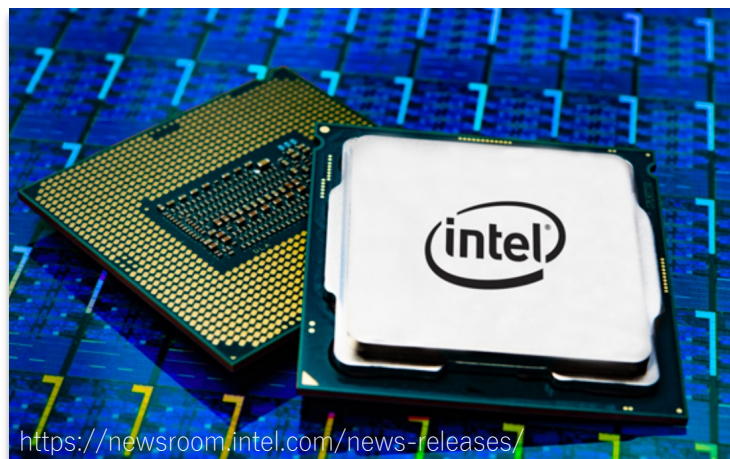
Central processing unit / CPU

❖ CPU

- ❖ 制御と演算の役割を担う
- ❖ コア, レジスタ, キャッシュ等からなる

❖ コア

- ❖ 制御, 演算を行うCPUの中核部分
- ❖ 1つのコアで, ある瞬間に同時に行える処理は1つ
仮想的に1コアで同時に複数処理を行えるHyper-threading technologyなどの技術もあるが, 本講義では扱わない。



<https://newsroom.intel.com/news-releases/>



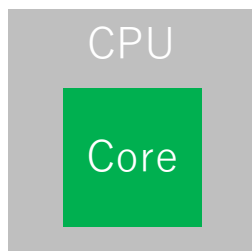
中央処理装置 (2/3)

Central processing unit / CPU

❖ シングル／マルチコアCPU

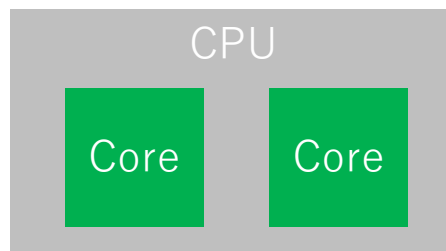
シングルコアCPU

1個のコアを持つCPU



マルチコアCPU

複数個のコアを持つCPU





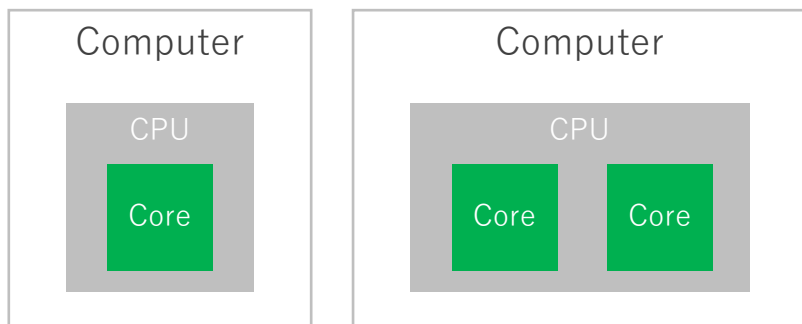
中央処理装置 (3/3)

Central processing unit / CPU

❖ シングル／マルチプロセッサ

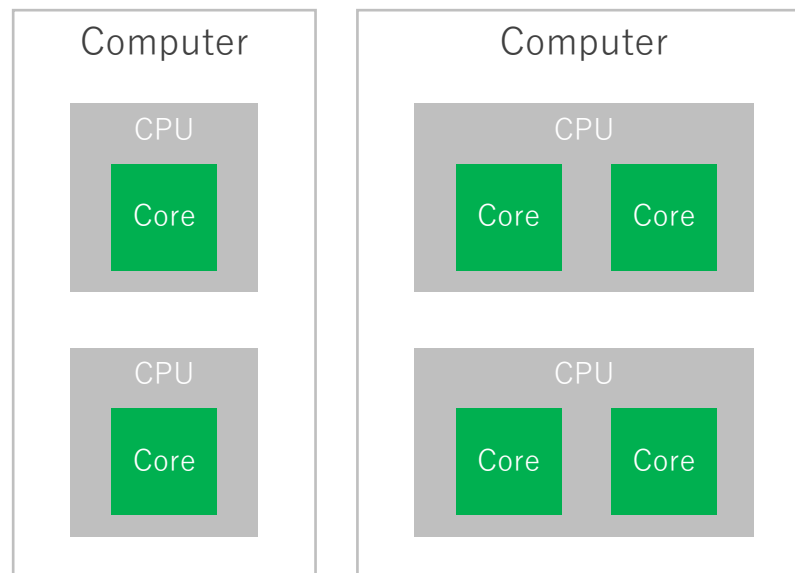
シングルスプロセッサ

1個のCPUを搭載した計算機



マルチプロセッサ

複数個のCPUを搭載した計算機



IR06-1

❖ 各自のPCのゲストOS（Ubuntu）とホストOSのCPUコア数を調べよ。

❖ Ubuntu上でのCPU情報の調べ方

- `less /proc/cpuinfo`
- 単純にコア数を調べるだけなら `grep -c processor /proc/cpuinfo`
- コア数はVirtual Boxの設定により異なる

❖ macOS／Windows上でのCPU情報の調べ方

- 各自Webで調べる



プロセスとスレッド

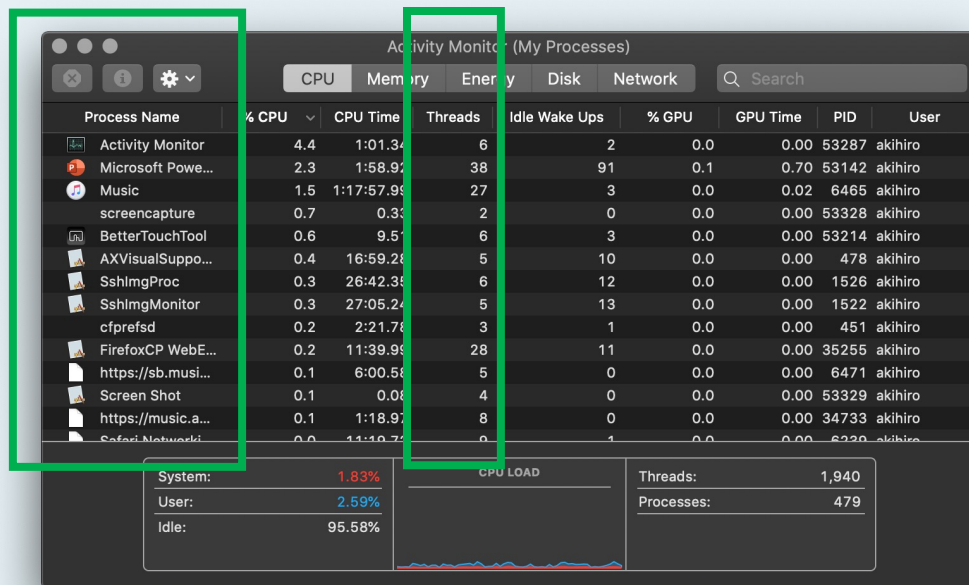
Process & Thread

❖ 処理の単位

- ❖ コンピュータ上の各動作はプロセスとして実行される
- ❖ 現代的なOSは、同時に複数のプロセスを実行できる
- ❖ 1つのプロセスは、1つ以上のスレッドから構成される

各プロセスは1つ以上のスレッドから構成される

アプリケーションや
バックグラウンドサービスは
プロセスとして実行される



Activity Monitor (macOS)

IR06-2

❖ Ubuntu上※で、Firefoxで複数のタブを立ち上げた状態で、下記の情報を取得して解答用紙に転載せよ。方法はWeb検索して調べて良い。

(1)Firefoxに関するプロセスの一覧

(2)Firefoxに関するスレッドの一覧 (一部で構わない)

※ 手元にUbuntuが用意できない場合は、他のOSで取り組んでも良い。

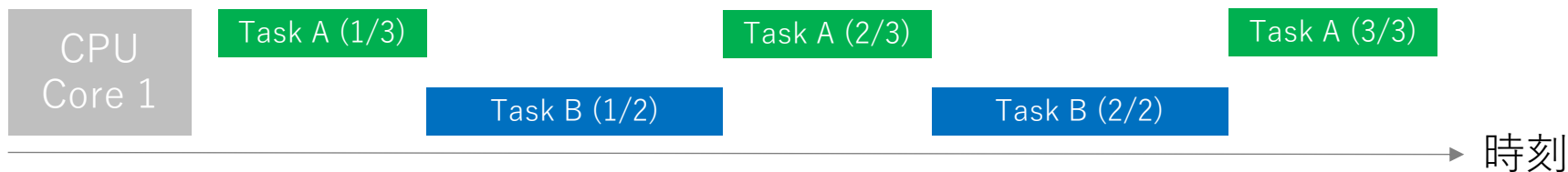


並行と並列

Concurrency vs Parallelism

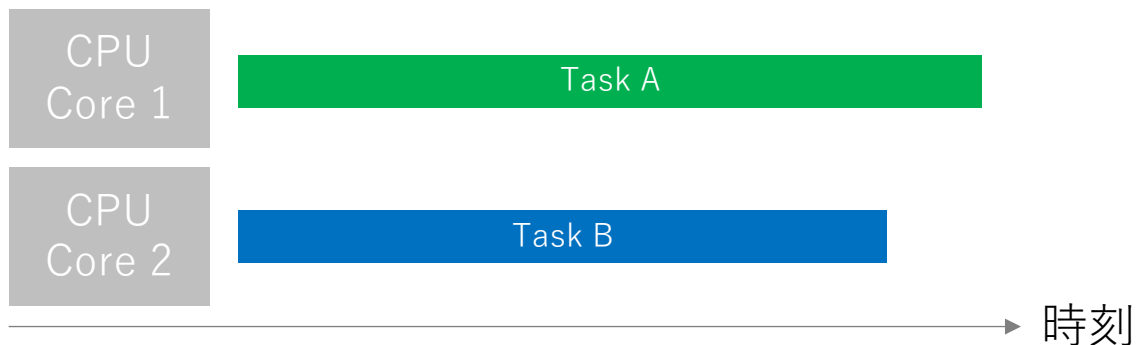
❖ 並行 Concurrency

- ❖ 複数の処理を同時に実行状態にする
- ❖ シングルコア／シングルスプロセッサでも可能
- ❖ たとえ：明日までに、社員1が、仕事Aと仕事Bをこなす



❖ 並列 Parallelism

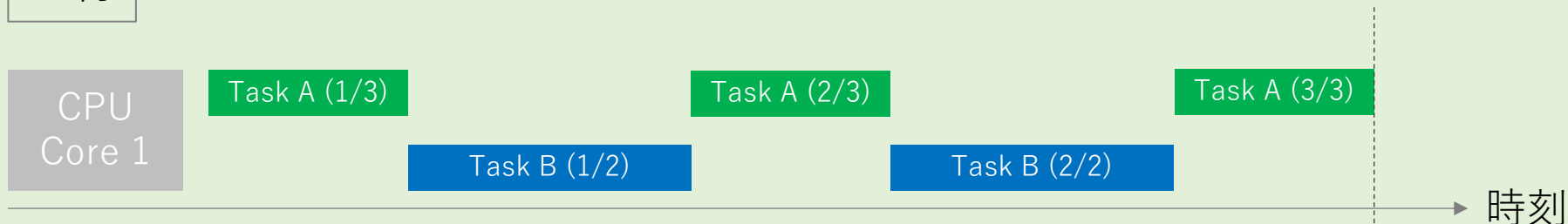
- ❖ 複数の処理を同時に実行する
- ❖ マルチコア／マルチプロセッサでないと実現できない
- ❖ たとえ：明日までに、社員1が仕事A、社員2が仕事Bをこなす



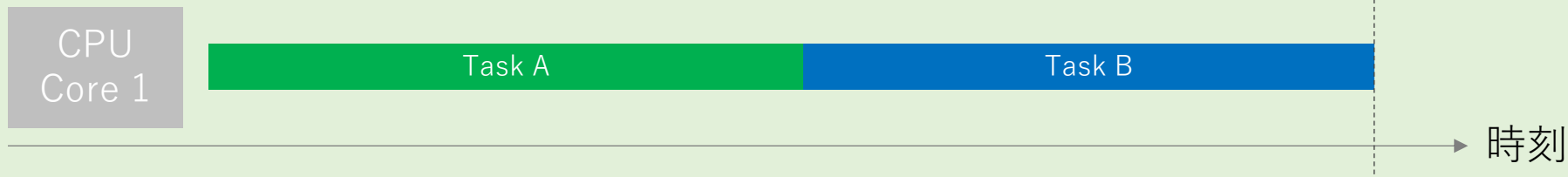
IR06-3

❖ 並行は，一見すると処理全体が早く終わらず（実際，多くのシーンでは早く終わらない），有益でないようにも思える。一体，並行はどのようなシーンで利用すると有益なのか考えよ。

並行



並行でない（逐次的）



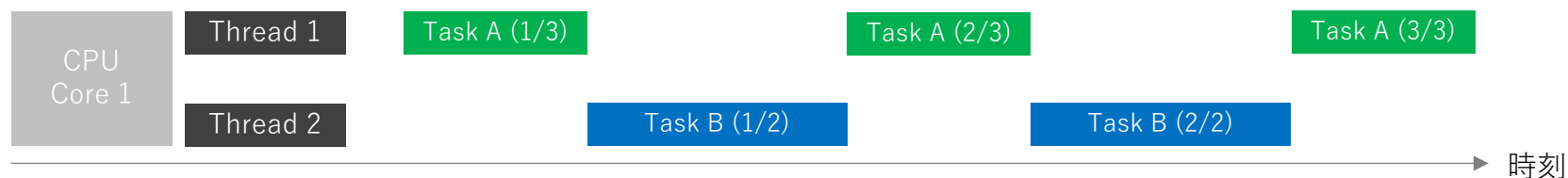


マルチスレッドと並行・並列

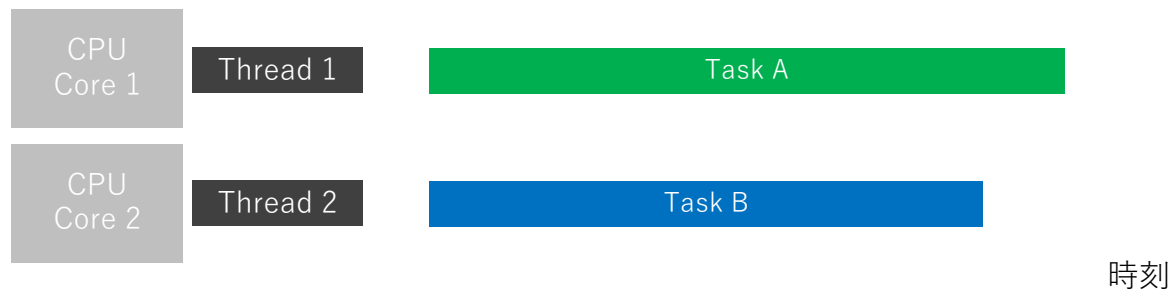
❖ マルチスレッド

- ❖ 複数のスレッドが同時に実行状態にあること
- ❖ 並行や並列を実現する手段

❖ 各スレッドが1つのCPUで動作している場合 → 並行



❖ 各スレッドが別のCPUで動作している場合 → 並列



各スレッドを明示的に別のCPUコアで動かす（＝並列）プログラミングは実行可能環境が限られる上に難易度が高いため、以降、本講義では扱わない。

IR06-4

❖ 次の挙動を実現するプログラムを作成せよ。

- キーボードから1～9の数字を1つ入力してEnterを押すと、その数字がコンソール上に繰り返し表示され続ける。
- 上記の数字が表示され続けている間に、新たに1～9の数字を1つ入力してEnterを押すと、その数字がコンソール上に繰り返し表示され続ける。0を入力してEnterを押すとプログラムが終了する。

ベースとなるコード (Echo.java)

```
package echo;
import java.util.Scanner;

public class Echo {

    private final static long SLEEP_LEN_MSEC = 100;

    private Scanner scanner;
    private int n;

    public Echo() {
        scanner = new Scanner(System.in);
        System.out.print("n > ");
        n = scanner.nextInt();
    }

    private void echo() throws InterruptedException {
        while(n != 0) {
            System.out.print(n);
            Thread.sleep(SLEEP_LEN_MSEC);
        }
    }

    public static void main(String[] args) throws InterruptedException {
        Echo ste = new Echo();
        ste.echo();
    }
}
```

ベースとなるコードの実行方法と挙動

```
% javac echo/Echo.java
% java echo/Echo
n > 1
1111111111111111. . .
```

一度whileループに入ってしまうと
他の処理は一切できないので
題意が満たせない。

- ❖ キーボードから数字を読み込む方法
Scanner scanner = new Scanner(System.in);
int i = scanner.nextInt();
- ❖ InterruptedException
一定時間スリープするThread.sleep()を実行する際に生じうる例外。



本日のまとめ

❖ 講義内容

- シングルコア／プロセッサとマルチコア／プロセッサ
- プロセスとスレッド
- 並行と並列
- マルチスレッドによる並行の実現

❖ 授業内課題提出

- 各授業内課題（IR）の解答を記載せよ。
- 「講義内容のまとめ」の解答欄に
上記「講義内容」の各項目について文章で説明を記載せよ。