

# 知能情報実験 1 : レポート課題 1

205713B 朝比奈 太郎

2021 年 6 月 18 日

## 目次

1	実験の目的と内容	2
2	実験 1 の事前予想と結果と考察	2
2.1	事前予想 . . . . .	2
2.2	結果 . . . . .	2
2.3	考察 . . . . .	2
3	実験 2 の事前予想と結果と考察	3
3.1	事前予想 . . . . .	3
3.2	結果 . . . . .	3
3.3	考察 . . . . .	3
4	実験 3 の事前予想と結果と考察	3
4.1	事前予想 . . . . .	3
4.2	結果 . . . . .	4
4.3	考察 . . . . .	4
5	使用するプログラミング言語の選定指針	4
5.1	本実験を通じて得られた結果から、言えること . . . . .	4
5.2	一般論として、どの言語がどんなプロジェクトに利用されているのか . . . . .	4

## 1 実験の目的と内容

一般的にコンパイラ言語、中間言語、インタプリタ言語の順番に実行速度が速いと言われているが実際のところ、コンパイラ言語とインタプリタ言語の実行速度にどの程度の差があるのか実験する。本実験では、いくつかの典型的な処理パターンを複数の言語で実装し、処理速度を比較して確認する。また、コンパイラ言語の代表として C、インタプリタ言語の代表として Python、両者の中間として Java を用いて実行速度の比較を行う。

## 2 実験 1 の事前予想と結果と考察

### 2.1 事前予想

今回の実験は「素数の数を調べる」というもので、for 文と if 文を用いる単純なコードであるため、一般的に言われる、コンパイラ言語、中間言語、インタプリタ言語の順番に実行速度が速いと考える。また C 言語の最適化の有無は、最適化を行ったコンパイルの方が早いと考える。従って、c with clang(最適化有り) - c with clang(最適化無し) - Java - Python の順番で実行速度が速いと予想する。

### 2.2 結果

- c with clang(最適化有り) : 0.265
- c with clang(最適化無し) : 0.296
- java : 0.556
- Python : 13.652

### 2.3 考察

コードが単純なためか、C 言語において、最適化ありとなしでは差が 0.03 秒で大きな差は見られなかった。java は、コンパイル言語とインタプリタ言語の中間言語であるが、タイムはコンパイル言語にかなり近く (0.3 秒差)、ほとんどコンパイラ言語と変わらない結果となった。Python は、13.652 秒と他の 3 つの項目と比べて大きく実行速度の差が出た。これは、C, java はともにユーザーが手動でコンパイルしてからの実行であるが、Python はコンパイルも一気にできてしまうという観点からコンパイル分の時間がかかっているのではないかと考える。また、JavaScript などの他言語には実行のたびにソースコードをコンパイルし、実行速度の高速化を測る JIT(Just in time) コンパイラがあるが、Python にも存在しないわけではないが、既存の処理系を置き換えるほどの成功には至っていないことが挙げられる。

## 3 実験 2 の事前予想と結果と考察

### 3.1 事前予想

今回の実験は「ファイルアクセス」というもので、0 以上 10,000,000 未満のすべての整数を昇順に文字列としてをファイルに書き込むコードであるため、ファイルの読み書きに強いまたはよく使われている感じのする Python が一番実行速度が速いと予想。続いて、c(最適化あり), c(最適化なし), java の順番。

### 3.2 結果

- c with clang(最適化有り) : 1.013
- c with clang(最適化無し) : 1.014
- java : 0.955
- Python : 5.354

### 3.3 考察

ファイルアクセスを伴うコードの実行速度は、ディスクドライブ等のアクセス速度、OS の処理、など外部要因が（プログラマが描いたコードやコンパイラ）大きく影響される。このことから、C 言語の最適化の有りと無しで実行速度に変わりがなかったのは、他のファイルにアクセスし、書き込むという行為に最適化が適応されなかったためであると考ええる。また、私は事前に python の実行速度が最もはやいと予想していたが、やはり自身でコンパイルをするという時間がどうしてもかかってしまい、処理速度が他の 3 つに比べて遅くなったと考える。

## 4 実験 3 の事前予想と結果と考察

### 4.1 事前予想

複雑な処理においては、処理系（言語）の違いよりも、選択するアルゴリズムの影響が大きくなることがある。スクリプト系の言語は、特に文字列、連想配列、ソートなどよく使われる処理の優れたライブラリを内蔵しており、より短いコードで効率的に動作する。このことより、唯一のスクリプト言語である Python が効率的に動作し、一番の実行速度を誇ると予想。次に実験 1、2 より C(最適化あり) が最も早い結果を出したので、C(最適化あり) が 2 番目。最後に、C(最適化なし) の順番になると予想する。

## 4.2 結果

- c with clang(最適化有り) : 1.965
- c with clang(最適化無し) : 3.397
- java : xxx
- Python : 0.223

## 4.3 考察

# 5 使用するプログラミング言語の選定指針

## 5.1 本実験を通じて得られた結果から、言えること

実験 1,2,3 において、1 から 3 にかけて実験内容 (コーディング) が複雑になっていた。それに伴い、コンパイラ言語である C 言語、中間言語である java の 2 言語の実行速度は実験が複雑になるにつれて遅くなった。しかし、インタプリタ言語である Python は実験の複雑さに反比例して実行速度が速くなった。したがって、単純な処理においては、基本的に処理速度の速いコンパイラ言語を用いるべきである。一方で、複雑な処理において、スクリプト系の言語は、特に文字列、連想配列、ソートなどよく使われる処理の優れたライブラリを内蔵しており、より短いコードで効率的に動作することから Python といったインタプリタ言語を用いるべきである。

## 5.2 一般論として、どの言語がどんなプロジェクトに利用されているのか

Python と C では、可読性が高い python の方が開発が容易だといえる。また、C 言語のようなコンパイル言語は、プログラムをまとめてコンパイルするため、エラーもまとめて対応しなければならない。つまり、Python のようなインタプリタ言語に比べてデバッグがしづらいというデメリットもあるため、Python の方が一般的に開発は容易だといえる。どんなプロジェクトでも Python をつかべきかということそうではなく、実験 1,2,3 を通してでた単純な処理であれば C の方が処理速度が早いという結果を元に単純なプロジェクトや、プログラムの実行速度が求められるプロジェクトの場合は C の方が適しているといえる。仮に、メンバのほとんどが java プログラムの場合であれば、java を用いて処理するのが好ましいが、java は c の派生言語であり、java プログラムであれば c を習得することが他のプログラマに比べて容易であるため、c を習得し、c でコーディングするのが良いと思う。

## 参考文献

- [1] ズバリ Python の弱点は？ AI 以外の有望分野は？ 気になる疑問に ..., <https://xtech.nikkei.com/atcl/nxt/cpbook/18/00034/00003/>, 2021/6/15.
- [2] コンパイラ型言語とインタプリタ型言語のメリット, <https://tech-camp.in/note/technology/93845/>, 2021/6/18
- [3] 赤嶺, 2021 実験 1-1 プログラムの実行速度, 2021/6/18