

システムソフトウェア特論 課題# 1

演習 1-1 の a,b,c を選択し解答した。授業資料に演習 1-1 が 2 つ存在しているが、私が解答したのは資料 5 ページ目に記載されているものである。

課題内容

次の計算を C 言語の「関数型」スタイルで書いてみよ。

- a. N を与えて $\sum_{i=1}^N i$
- b. N を与えて $\sum_{i=1}^N i^2$
- c. N を与えて $fib(N)$ ただし $fib(0) = fib(1) = 1, fib(n) = fib(n-1) + fib(n-2) (n \geq 2)$

方針

授業資料のものを参考に書き、いくつかの実行結果を実際の数値と比較し正当性を評価することにした。

成果物とその実行結果

演習問題の計算を C 言語の関数型スタイルで書いたものは次の通りである。ただし、プログラム内の `f_a`, `f_b`, `f_c` はそれぞれ、演習問題の a, b, c に対応している。いずれも再帰的に問題の計算を表現した。

Listing 1: 成果物

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 int f_a(int);
4 int f_b(int);
5 int f_c(int);
6 void main(int argc, char *argv[]) {
7     int x = atoi(argv[1]);
8     printf("f_a(x)=%d\nf_b(x)=%d\nf_c(x)=%d\n", f_a(x), f_b(x), f_c(x));
9 }
10
11 int f_a(int n) {
12     if (n < 1) { return 0; }
13     else { return n + f_a(n - 1); }
14 }
15 int f_b(int n) {
16     if (n < 1) { return 0; }
```

```

17         else { return n * n + f_b(n - 1); }
18     }
19     int f_c(int n) {
20         if (n < 2) { return 1; }
21         else if(n >= 2){ return f_c(n - 1) + f_c(n - 2); }
22         else { printf("x<0\n!!"); return -1; }
23     }

```

x を 1,11,39 として実行した結果は以下の通りであった。

————— $x = 1$ での実行結果 —————

```

a.exe_1
f_a(x)=1
f_b(x)=1
f_c(x)=1

```

————— $x = 11$ での実行結果 —————

```

a.exe_11
f_a(x)=66
f_b(x)=506
f_c(x)=144

```

————— $x = 39$ での実行結果 —————

```

a.exe_39
f_a(x)=780
f_b(x)=20540
f_c(x)=102334155

```

考察

問題 a,b の計算結果はそれぞれ式 (1), (2) から確認できる。これらより、 $f_a(1) = f_b(1) = 1$, $f_a(11) = 66$, $f_b(11) = 506$, $f_a(39) = 780$, $f_b(39) = 20540$ であることが確認できた。これらの値は実行結果と一致するため、作成した関数は、問題の計算を正しく書いたものであると考えられる。 f_c のフィボナッチ数列はオンライン整数列大辞典 [1] より $f_c(1) = 1$, $f_c(11) = 144$, $f(39) = 102334155$ であることが確認できた。ただし、この文献では $F(0) = 0$, $F(1) = 1$ としていたため、それに対応させ $f_c(n) = F(n+1)$ として考えた。

$$f_a(x) = \frac{n(n+1)}{2} \quad (1)$$

$$f_b(x) = \frac{1}{6}n(n+1)(2n+1) \quad (2)$$

プログラムを実行したとき、 $x = 1$ および $x = 11$ のときは即座に結果が出力されたが $x = 40$ の

ときは1秒ほど時間がかかった。入力する x の値が大きくなればさらに計算時間が大きくなると予想される。これは、このプログラムが再帰的なものであり、与えられた x 以下すべての数に対しての $f(x)$ を計算しているからと考えられる。例えば $x = 11$ としたとき、 $f(1), f(2), \dots, f(11)$ すべてを計算しているということである。さら問題に c のフィボナッチ数列では、 $f(x)$ の関数内で $f(x-1)$ と $f(x-2)$ を同時に呼び出しているため、 $f(x-3), f(x-4)$ など同じ計算を複数行っていることもあると考えられる。

実行時間の短縮を考えるなら問題 a, b の場合、式 (1), (2) の形をプログラムにすれば、入力された x が大きいとき計算時間を削減できると考えられる。問題 c は、一度計算した値を保存するメモ化を行えば重複計算を避けて計算時間を削減できると考えられる。ただし、これは課題の「関数型」であることに反してしまう。

アンケート

Q1. プログラミングは好き/得意ですか、苦手ですか。それはなぜですか。どういうところが特にそう思う？

プログラミングは好きだと感じています。特に自分が想像したものや、望んだものを自由に形にできるところが好きです。プログラミングを行う多くの方を見ると自分はまだまだ得意とは言えないと思っています。

Q2. さまざまなプログラミング言語があるということについてどのように考えていますか。

それぞれの言語には得意、不得意があると感じています。自身の用途に合った言語を選択し設計できればよいと考えています。

Q3. リフレクション (課題をやってみて気付いたこと)、感想、要望など。

この課題は授業初期のものであるため難しいと感じるところはありませんでした。少し物足りなさも感じたため他の課題も取り組んでみたいと思いました。

参考文献

[1] オンライン整数列大辞典 <https://oeis.org/A000045>