

【No. 53】 C 言語により記述された次の関数 a, b が実行可能とする。

```
int a(int x){
    display();
    if(x == 0) return 1;
    return(a(x - 1) + b(x));
}
int b(int x){
    if(x == 0) return 1;
    return(a(x - 1) + b(x - 1));
}
```

ここで display(); は画面に 1 個の文字 * を書く働きをする手続きであるとする。

いま、ある整数 n に対して a(n) を呼び出したとき、値を返して停止し、文字 * は画面に 144 個書かれた。このときの n はいくらか。

1. 3 2. 4 3. 5 4. 6 5. 7

あまりうまい方法は見つけていませんが、答えが最大で 7 までですので、実際に規則を探しながら、いざとなったら最後まで調べきってしまう方針がよいかと思います。ここでは漸化式を導いて考えています。もしかしたらうまい規則があるのかもしれませんが。

解答

a(n) を呼び出したときに最終的に画面に書かれる * の数を p_n とし、b(n) を呼び出したときに最終的に画面に書かれる * の数を q_n とする。文字は a(x) を呼び出す毎に 1 回書かれることに注意する。

a(n) を呼び出すと、* が 1 個書かれ、a(n-1) と b(n) が呼ばれる。一方、b(n) が呼ばれると、a(n-1) と b(n-1) が呼ばれる。したがって、結局 a(n) が呼ばれると、a(n-1) が 2 回と b(n-1) が 1 回呼ばれ、画面の * が 1 個増えることになる。2 回の a(n-1) によって、更に $2p_{n-1}$ 個、b(n-1) によって q_{n-1} 個の * が書かれることになるので、次の漸化式が導かれる。

$$p_n = 2p_{n-1} + q_{n-1} + 1 \quad \cdots \textcircled{1}$$

上の式を導く途中に出てきたとおり、b(n) が呼ばれると、a(n-1) と b(n-1) がよばれ、結局画面に $p_{n-1} + q_{n-1}$ 個の * が書かれる。したがって、

$$q_n = p_{n-1} + q_{n-1} \quad \cdots \textcircled{2}$$

以上の連立漸化式を解く。 $p_1=3$ (a(1) と a(0) が 2 回呼ばれる)、 $q_1 = 1$ (a(0) が呼ばれる) より、 $(p_2, q_2) = (8, 4)$ 、 $(p_3, q_3) = (21, 12)$ 、 $(p_4, q_4) = (55, 33)$ 、 $(p_5, q_5) = (144, 88)$ となる。

画面に * が 144 個書かれているので、 $n = 5$ である。

p_n のみの漸化式

$$p_n = 3p_{n-1} - p_{n-2}$$

を導くこともできますが、かえって面倒に思われます