Mysterious Sequence

Designer: Yi-Chang Lin

As a Start...

- Have you read the solution of Practice 3_Q1: Fibonacci Sequence?
- If yes, currently, do you have clear thoughts about how to solve Practice 3_Q1: Fibonacci Sequence?
- The idea of Mysterious Sequence comes from the problems on Formosa OJ:
  1218_Big Mod, 1221_大大的費式數列, 1230_大大的神秘數列

Break Down the Problem Description

In this problem, you are asked to calculate the value of the ith term of a given sequence modulo $10^7 + 7$.
Below is the definition of the sequence's ith term F_i:

When $i < t$: $F_i = x_i$
When $i \geq t$: $F_i = \sum_{j=1}^{t} C_j * F_{i-j}$

The first term of the sequence is $i = 0$.
t is a positive integer, and both x_i and C_j are nonnegative integers.

Explanation 0

$t = 3, x_0 = 76, x_1 = 62, x_2 = 8, C_1 = 89, C_2 = 36, C_3 = 35, i = 5$

$F_3 = C_1 * F_2 + C_2 * F_1 + C_3 * F_0 = 89 * 8 + 36 * 62 + 35 * 76 = 5604$

$F_4 = C_1 * F_3 + C_2 * F_2 + C_3 * F_1 = 89 * 5604 + 36 * 8 + 35 * 62 = 501214$

$F_5 = C_1 * F_4 + C_2 * F_3 + C_3 * F_2 = 89 * 501214 + 36 * 5604 + 35 * 8 = 44810070$

$44810070 \,\%\, 10000007 = 4810042$

Constraints

$1 \leq t \leq 6$
$0 \leq x_0, x_1, \ldots, x_{t-1} \leq 10^6$
$0 \leq C_1, C_2, \ldots, C_t \leq 10^6$
$0 \leq i \leq 10^9$

Time limit: 1 seconds.
Memory limit: 256 MB.

- ❑ The difference between Iteration Version and Recursion Version?

At least 10 points!!

The Fast Power Approach

- Calculate 2 to the power of 8

  Iteration: 2 * 2 => 4 * 2 => 8 * 2 => 16 * 2 => 32 * 2 => 64 * 2 => 128 * 2 => 256

  Fast Power: 2 * 2 => 4 * 4 => 16 * 16 => 256 **7 times v.s. 3 times**

- $F_i = F_{i-1} + F_{i-2}$ (when i >= 2)

$$\begin{bmatrix}1 & 1\\ 1 & 0\end{bmatrix}\cdot\begin{bmatrix}F_{i-1}\\ F_{i-2}\end{bmatrix}=\begin{bmatrix}F_i\\ F_{i-1}\end{bmatrix} \qquad \begin{pmatrix}F_i = 1*F_{i-1}+1*F_{i-2}\\ F_{i-1} = 1*F_{i-1}+0*F_{i-2}\end{pmatrix}$$

$$\begin{bmatrix}1 & 1\\ 1 & 0\end{bmatrix}\cdot\begin{bmatrix}F_1\\ F_0\end{bmatrix}=\begin{bmatrix}F_2\\ F_1\end{bmatrix} \Rightarrow \begin{bmatrix}1 & 1\\ 1 & 0\end{bmatrix}\cdot\begin{bmatrix}F_2\\ F_1\end{bmatrix}=\begin{bmatrix}F_3\\ F_2\end{bmatrix} \Rightarrow \begin{bmatrix}1 & 1\\ 1 & 0\end{bmatrix}^{\boldsymbol{n}}\cdot\begin{bmatrix}F_1\\ F_0\end{bmatrix}=\begin{bmatrix}\boldsymbol{F_i}\\ F_{i-1}\end{bmatrix} \text{ or } \begin{bmatrix}F_{i+1}\\ \boldsymbol{F_i}\end{bmatrix}$$

- $F_i = \boldsymbol{C_1} * F_{i-1} + \boldsymbol{C_2} * F_{i-2} + \boldsymbol{C_3} * F_{i-3}$ (when i >= 3)

$$\begin{bmatrix}a_{0,0} & a_{0,1} & a_{0,2}\\ a_{1,0} & a_{1,1} & a_{1,2}\\ a_{2,0} & a_{2,1} & a_{2,2}\end{bmatrix}\cdot\begin{bmatrix}F_2\\ F_1\\ F_0\end{bmatrix}=\begin{bmatrix}F_3\\ F_2\\ F_1\end{bmatrix} \Rightarrow \begin{pmatrix}F_3 = a_{0,0}*F_2 + a_{0,1}*F_1 + a_{0,2}*F_0\\ F_2 = a_{1,0}*F_2 + a_{1,1}*F_1 + a_{1,2}*F_0\\ F_1 = a_{2,0}*F_2 + a_{2,1}*F_1 + a_{2,2}*F_0\end{pmatrix}$$

- In this problem:

$$\begin{bmatrix}a_{0,0} & \cdots & a_{0,t-1}\\ \vdots & \ddots & \vdots\\ a_{t-1,0} & \cdots & a_{t-1,t-1}\end{bmatrix}^{n}\cdot\begin{bmatrix}F_{t-1}\\ \vdots\\ F_0\end{bmatrix}=\begin{bmatrix}\boldsymbol{F_?}\\ \vdots\\ \boldsymbol{F_?}\end{bmatrix}$$

Constraints:
1 <= t <= 6

Implementation details

- Clean Code: naming style, divide your program into small blocks

```
int main(void) {
    read t;
    initializeMatrices(firstTerms, baseMatrix, resultMatrix);
    read i;
    fastPower(baseMatrix, resultMatrix, t, n);
    // copyMatrix(copied, origin), multiplyMatrix(dest, src1, src2) in fastPower()
    calculate the final result;
    print the final result;
    return 0;
}
```

Tips: Debug one block at a time!

```
3
76 62 8
89 36 35
5
```

- Which might generate error message during compilation?

1. `void func(int arr[]);` 2. `void func(int arr[][]);` 3. `void func(int arr[][][]);`

Alternatives: global variable, **pointer**

Implementation details (cont.)

Explanation 0

$t = 3, x_0 = 76, x_1 = 62, x_2 = 8, C_1 = 89, C_2 = 36, C_3 = 35, i = 5$

$F_3 = C_1 * F_2 + C_2 * F_1 + C_3 * F_0 = 89 * 8 + 36 * 62 + 35 * 76 = 5604$

$F_4 = C_1 * F_3 + C_2 * F_2 + C_3 * F_1 = 89 * 5604 + 36 * 8 + 35 * 62 = 501214$

$F_5 = C_1 * F_4 + C_2 * F_3 + C_3 * F_2 = 89 * 501214 + 36 * 5604 + 35 * 8 = 44810070$

$44810070 \,\%\, 10000007 = 4810042$

- Be aware of possible arithmetic overflows

  Print the ranges of different data types using <limit.h>

- Integers modulo n

  Let a and b are arbitrary integers and n is a positive integer:
  - (a + b) % n = (a % n) + (b % n)
  - (ab) % n = (a % n) * (b % n)

- Practice, Practice, and Practice!

Constraints

$1 \le t \le 6$
$0 \le x_0, x_1, \ldots, x_{t-1} \le 10^6$
$0 \le C_1, C_2, \ldots, C_t \le 10^6$
$0 \le i \le 10^9$

Time limit: 1 seconds.
Memory limit: 256 MB.

Questions?

Supplemental Materials

- How to debug?
  - Use ”printf()” on HackerRank directly => limited output length
    (demo: kTimesTransaction_withMsg.c)
  - Redirect I/O
    (demo: kTimesTransaction_withMsg.c)
  - Debugger
    (demo: matMultiplication.c, kTimesTransaction.c)
- Make sure your program terminated
  (demo: matMultiplication.c)
- How to improve yourself?
  - homework, lab practices, Formosa OJ, some online judge platforms…