

# ICS LAB\_02 自定义斐波那契数列 实验报告

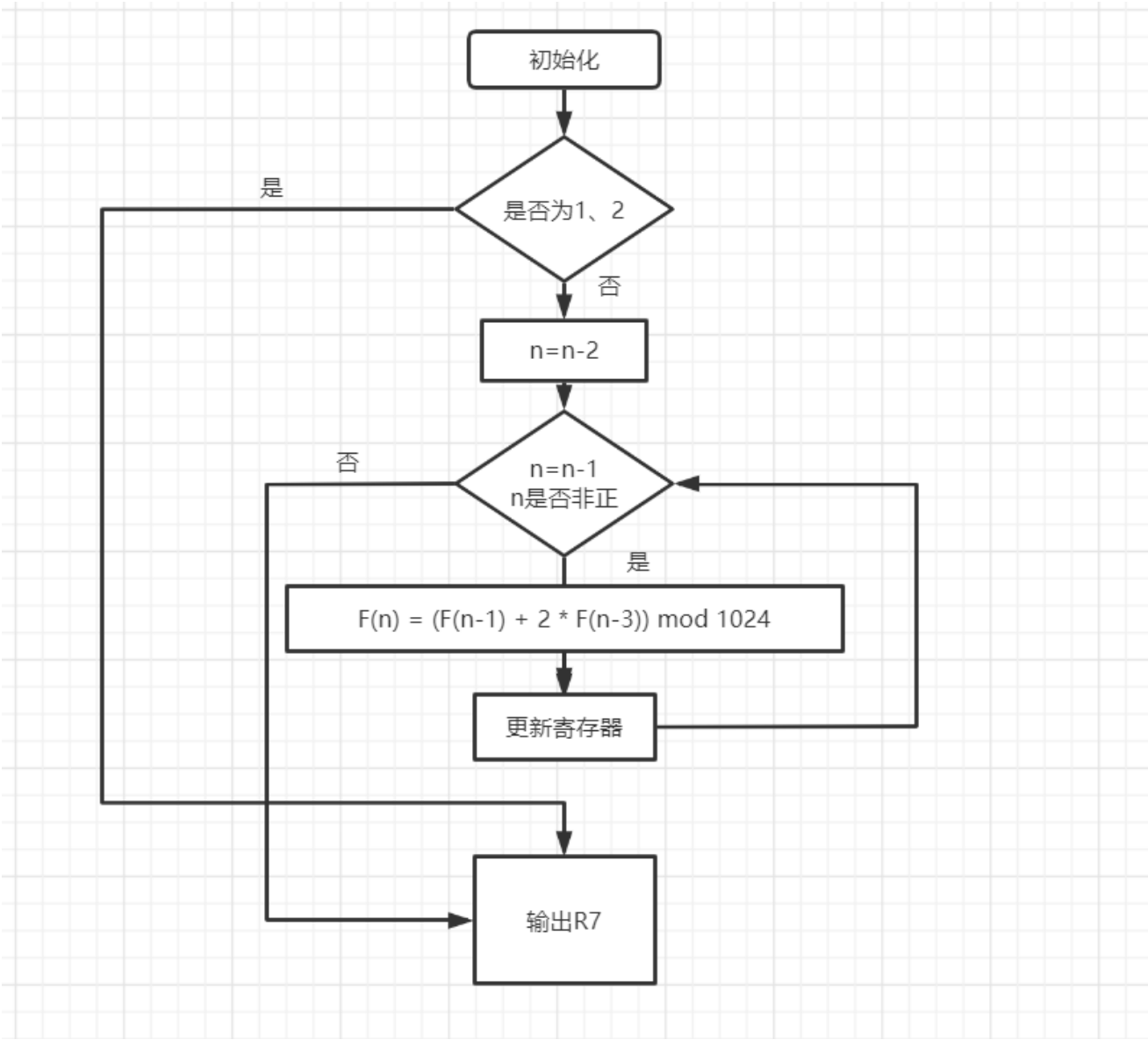
## 代码实现

采用c风格书写，用注释进行解释。共29行。

```
.orig x3000
LD R1 NUMBERA;//R1存储F(n-3)
LD R2 NUMBERA;//R2存储F(n-2)
LD R3 NUMBERB;//R3存储F(n-1)
LD R6 NUMBERC;//R6存储1023 (10'b11_1111_1111) 用于取模
LD R7 NUMBERA;//R7初始化为F(1)
ADD R0, R0, #-1;//判断n是否是1
BRz END;//输出
LD R7 NUMBERB;//R7初始化为F(2)
ADD R0, R0, #-1;//判断n是否是2
BRz END;//输出
LOOP ADD R0, R0, #-1;//n-1, 判断运行次数
BRn END;//为负数代表运行完成
ADD R1, R1, R1;//R1=2*F(n-3)
ADD R4, R3, R1;//R4=(F(n-1)+2*F(n-3))
AND R5, R4, R6;//R5=(F(n-1)+2*F(n-3)) mod (1024)
ADD R1, R2, #0;//更新F(n-3)
ADD R2, R3, #0;//更新F(n-2)
ADD R3, R5, #0;//更新F(n-1)
ADD R7, R3, #0;//将F(n) 存储进入R7
BRnzp LOOP;//循环
END HALT;//结束
NUMBERA .FILL #1//F(0)、F(1)
NUMBERB .FILL #2//F(2)
NUMBERC .FILL #1023
a .FILL #930//学号第一部分结果
b .FILL #50//学号第二部分结果
c .FILL #1014//学号第三部分结果
d .FILL #470//学号第四部分结果
.end
```

# 设计思路

通过流程图解释



# 样例测试与正确性检验

## 验证方法

对于 $n > 2$ 的情况，通过一个简单的c程序进行验证，代码如下。

```
#include <stdio.h>
#include<stdlib.h>

int f(int n0, int n1, int n2, int n) {
    int fn;
    fn=(2*n0+n2)%1024;
    if(n==3)
        return fn;
    else if(n>3)
        return f(n1, n2, fn, n-1);
    else
        return 0;
}

int main() {
    int n;

    scanf("%d", &n);
    printf("%d\n", f(1, 1, 2, n));

    system("pause");
    return 0;
}
```

代入我的学号PB20081583，得到 $n=20, 8, 15, 83$ 的结果依次为930, 50, 1014, 470.

## 代码测试（此处a、b、c、d均为学号）

1. 进行“LD R0 a”操作

<b>R7</b>	x03A2	930
-----------	-------	-----

2. 进行“LD R0 b”操作

<b>R7</b>	x0032	50
-----------	-------	----

3. 进行“LD R0 c”操作

<b>R7</b>	x03F6	1014
-----------	-------	------

4. 进行“LD R0 d”操作

R7	x01D6	470
----	-------	-----

均成功通过。