

中国科学技术大学计算机学院
《计算机系统概论实验报告》



实验题目: Better Angels

学生姓名: 叶子昂

学生学号: PB20020586

完成时间: 2021年12月11日

0.1 实验目的:

通过类斐波那契数列的结果猜测输入，优化类斐波那契数列的结果

0.2 实验环境:

- window11 和 wsl2
- VScode以及LC3TooL

0.3 实验过程:

0.3.1 猜学号部分:

所收到的结果为:

```
.FILL #930
.FILL #1
.FILL #1
.FILL #338
```

写一个C程序列出100以内的所有结果930对应20，1对应00或01，338对应98.

在课程主页中公示的学号只有PB2000098符合，经过与该同学交流后确认。

0.3.2 优化部分

(收到的代码平均指令执行数为29966条)

初版优化

我所收到的同学的L版本的代码主要通过迭代完成计算，这样当输入较大时就会产生不小的计算量，如何替代反复迭代的过程成为优化的重点，通过写C程序发现类斐波那契数列的结果在20及以后以20~147所对应的结果循环，它的周期正好为128，那么对大于或等于20的输入，可以对127取模即可将迭代限制在1到147之间，较大幅度减少计算指令。

- 代码如下:

```
.ORIG X3000
    LD R6 F1      ;读入19用以计数
    ADD R1 R1 #1;读入初始值
    ADD R2 R2 #1
    ADD R3 R3 #2
    LD R5 F2      ;读入1023对结果取余
    ADD R7 R1 #0;考虑结果n=0, n=1, n=2
    ADD R6 R6 #-1
    ADD R0 R0 #-1
    BRN L0
    ADD R7 R2 #0
    ADD R6 R6 #-1
    ADD R0 R0 #-1
    BRN L0
    ADD R7 R3 #0
```

```

        ADD R6 R6 #-1
        ADD R0 R0 #-1
L1      BRN L0          ;考虑n在0~19
        ADD R4 R1 R1
        ADD R7 R3 R4
        AND R7 R7 R5
        ADD R1 R2 #0
        ADD R2 R3 #0
        ADD R3 R7 #0
        ADD R6 R6 #-1;计数判断n是否大于19
        BRN L2
        ADD R0 R0 #-1
        BRNZP L1
L2      LD R6 F          ;考虑n>=20
        AND R0 R0 R6;对127取模
        ADD R0 R0 #-1
L3      BRN L0          ;对取模后的结果迭代计算
        ADD R4 R1 R1
        ADD R7 R3 R4
        AND R7 R7 R5
        ADD R1 R2 #0
        ADD R2 R3 #0
        ADD R3 R7 #0
        ADD R0 R0 #-1
        BRNZP L3
L0      HALT
F        .FILL #127
F1       .FILL #19
F2       .FILL #1023
.END

```

- 采用提供的10个测试点来判断指令条数

指令执行数分别为：28，250，1330，682，610，646，466，970，1186，898，466。

平均指令执行数约为582。ratio=29966/582结果约为51.5。

最终优化

既然结果有在20以后以128为周期的规律，可以将1到19和20到147的结果枚举出来，直接省去迭代求解的过程，根据输入索引输出，能极大地减少指令执行数。

代码如下：

```

. ORG X3000
        LD R2 BASE
        ADD R3 R0 R2
        BRP s2          ;判断是否大于19
        LEA R1 TAG1 ;小于等于19相对TAG1索引

```

```

        ADD R0 R0 R1
        BRNZP res
s2      LEA R1 TAG2 ;大于19相对TAG2索引
        LD   R2 get
        AND R3 R3 R2
        ADD R0 R1 R3
res     LDR R7 R0 #0;读取索引结果
HALT
BASE   .FILL #-19
get    .FILL #127
TAG1   .FILL #1
        .FILL #1
        .FILL #2
        .FILL #4
        .FILL #6
        .FILL #10
        .FILL #18
        .FILL #30
        .FILL #50
        .FILL #86
        .FILL #146
        .FILL #246
        .FILL #418
        .FILL #710
        .FILL #178
        .FILL #1014
        .FILL #386
        .FILL #742
        .FILL #722
        .FILL #470
TAG2   .BLKW #1
        .FILL #930
        .FILL #326
        .FILL #242
        .FILL #54
        .FILL #706
        .FILL #166
        .FILL #274
        .FILL #662
        .FILL #994
        .FILL #518
        .FILL #818
        .FILL #758
        .FILL #770
        .FILL #358
        .FILL #850
        .FILL #342
        .FILL #34

```

. FILL #710
. FILL #370
. FILL #438
. FILL #834
. FILL #550
. FILL #402
. FILL #22
. FILL #98
. FILL #902
. FILL #946
. FILL #118
. FILL #898
. FILL #742
. FILL #978
. FILL #726
. FILL #162
. FILL #70
. FILL #498
. FILL #822
. FILL #962
. FILL #934
. FILL #530
. FILL #406
. FILL #226
. FILL #262
. FILL #50
. FILL #502
. FILL #2
. FILL #102
. FILL #82
. FILL #86
. FILL #290
. FILL #454
. FILL #626
. FILL #182
. FILL #66
. FILL #294
. FILL #658
. FILL #790
. FILL #354
. FILL #646
. FILL #178
. FILL #886
. FILL #130
. FILL #486
. FILL #210
. FILL #470
. FILL #418

. FILL #838
. FILL #754
. FILL #566
. FILL #194
. FILL #678
. FILL #786
. FILL #150
. FILL #482
. FILL #6
. FILL #306
. FILL #246
. FILL #258
. FILL #870
. FILL #338
. FILL #854
. FILL #546
. FILL #198
. FILL #882
. FILL #950
. FILL #322
. FILL #38
. FILL #914
. FILL #534
. FILL #610
. FILL #390
. FILL #434
. FILL #630
. FILL #386
. FILL #230
. FILL #466
. FILL #214
. FILL #674
. FILL #582
. FILL #1010
. FILL #310
. FILL #450
. FILL #422
. FILL #18
. FILL #918
. FILL #738
. FILL #774
. FILL #562
. FILL #1014
. FILL #514
. FILL #614
. FILL #594
. FILL #598
. FILL #802

```
. FILL #966
. FILL #114
. FILL #694
. FILL #578
. FILL #806
. FILL #146
. FILL #278
. FILL #866
. FILL #134
. FILL #690
. FILL #374
. FILL #642
. FILL #998
. FILL #722
. FILL #982
. END
```

- 在十个测试点下指令执行数分别为：

7, 7, 7, 8, 8, 8, 8, 8, 8, 8 。

平均指令执行数约为：8 $\text{ratio}=29966/8$ 约为3746 。

0.4 实验总结与思考

本次实验主要部分为前一次实验的关于指令数的优化，关键在于类斐波那契数列的结果有循环特性的发现，以及针对循环周期较少时使用打表方法进行执行数的优化减少重复步骤。