

自动机实验报告

221220028 金文泽

实验完成情况

1. 实现了 PDA 和 TM 的解析器/模拟器
2. 为 PDA 和 TM 均实现了对应的 verbose 模式 **bonus**
3. 完成了 PDA 和 TM 的用例程序 `pda/case.py` 和 `tm/case1.py` `tm/case2.py`
4. 能够检测以下三种错误：
 1. PDA TM 文件语法错误，例如该有括号的地方没有，使用了不合法的字符等。程序中体现为 `AutomataSyntaxException`。最终输出 `syntax error`，且在 verbose 模式下输出更多错误信息。 **bonus**
 2. PDA TM 文件逻辑（语义）错误，例如状态转移函数中的状态不存在，状态转移函数中的符号不存在等。程序中体现为 `AutomataStructureException`。最终输出 `syntax error`（因为此部分实验中未要求，所以同样输出，其实并不合适）。 **bonus**
 3. 非法的输入串，输入串中有符号未在字母表中，程序中体现为 `InputSyntaxError`，最终输出 `illegal input`。
5. TM 支持一些更高级的功能，比如如果有完全确定的状态转移函数，那么可以直接使用，而不会使用可能存在通配符的状态转移函数。

设计思路

PDA 和 TM 的思路类似，总体上均包含以下几个模块（分别用 PDA/TM 作为前缀）：

1. `Context`: 用于存储自动机运行的必要信息，具体而言，就是那个定义中的多元组。同时，`Context` 利用一个 `unordered_map` 来存储状态转移函数，这样可以在 $O(1)$ 的时间复杂度内找到对应的状态转移函数。为了能够使用 STL 的 `unordered_map`，设计了键值类 `TransitionKey` 和 `TransitionValue`，分别用于作为 `unordered_map` 的键和值。同时由于只能使用 C++14，所以没有办法使用 `optional`，所以自定义了 `QueryResult` 类，用于表示可能失败的查询结果。
2. `Parser`: 用于解析 PDA/TM 文件，将其转换为 `Context` 对象。具体而言，`Parser` 会读取文件，进行语法和语义检查，然后将其转换为 `Context` 对象。在转换过程中，`Parser` 会检查语法错误，而不会检查语义错误。
3. `Emulator`: 用于模拟 PDA/TM 的运行。具体而言，`Emulator` 会接受一个输入串，然后根据状态转移函数进行状态转移，直到达到终止状态。在模拟过程中，`Emulator` 会检查输入串是否合法，即输入串中的符号是否在字母表中。`Emulator` 同时会在运行前检查自动机是否合法，也就是语义错误。PDA 的模拟器运用一个栈来模拟栈操作，而 TM 的模拟器则使用自定义的 `MTape` 类来模拟磁带操作。

部分结果展示

PDA

```

1  $ ./bin/fla pda/anbn.pda aabbb
2  false
3
4  $ ./bin/fla -v pda/anbn.pda aabbb
5  Input: aabbb
6  ===== RUN =====
7  Step : 0
8  State: q0
9  Stack: z || (Stack Bottom)
10 -----
11 Step : 1
12 State: q1
13 Stack: 1 z || (Stack Bottom)
14 -----
15 Step : 2
16 State: q1
17 Stack: 1 1 z || (Stack Bottom)
18 -----
19 Step : 3
20 State: q2
21 Stack: 1 z || (Stack Bottom)
22 -----
23 Step : 4
24 State: q2
25 Stack: z || (Stack Bottom)
26 -----
27 Step : 5
28 State: accept
29 Stack: || (Stack Bottom)
30 -----
31 Result: false
32 ===== END =====
33
34 $ ./bin/fla -v pda/anbn.pda aabbc
35 Input: aabbc
36 ===== ERR =====
37 error: 'c' was not declared in the set of input symbols
38 Input: aabbc
39      ^
40 ===== END =====
41
42 $ ./bin/fla pda/anbn.pda aabbc
43 illegal input
44
45 $ ./bin/fla pda/anbn.pda aabb
46 true
47
48 $ ./bin/fla pda/test.pda aabb # 一个有语法错误的 PDA 文件
49 syntax error
50

```

```

51 $ ./bin/fla -v pda/test.pda aabb
52 syntax error
53 Syntax error at line 4: #Q = {q0,q1,q2,accept
54 Location: {q0,q1,q2,accept
55 Description: Expected {}.
```

TM

```

1 $ ./bin/fla -v tm/palindrome_detector_2tapes.tm 10001B10001
2 Input: 10001B10001
3 ===== ERR =====
4 error: 'B' was not declared in the set of input symbols
5 Input: 10001B10001
6      ^
7 ===== END =====
8
9 $ ./bin/fla tm/palindrome_detector_2tapes.tm 10001B1001
10 illegal input
11
12 $ ./bin/fla -v tm/palindrome_detector_2tapes.tm 1001001
13 Input: 1001001
14 ===== RUN =====
15 Step   : 0
16 State  : 0
17 Index0 : 0 1 2 3 4 5 6
18 Tape0   : 1 0 0 1 0 0 1
19 Head0   : ^
20 Index1 : 0
21 Tape1   : _
22 Head1   : ^
23 -----
24 Step   : 1
25 State  : cp
26 Index0 : 0 1 2 3 4 5 6
27 Tape0   : 1 0 0 1 0 0 1
28 Head0   : ^
29 Index1 : 0
30 Tape1   : _
31 Head1   : ^
32 -----
33 Step   : 2
34 State  : cp
35 Index0 : 0 1 2 3 4 5 6
36 Tape0   : 1 0 0 1 0 0 1
37 Head0   : ^
38 Index1 : 0 1
39 Tape1   : 1 _
40 Head1   : ^
41 -----
```

```

42 Step : 3
43 State : cp
44 Index0 : 0 1 2 3 4 5 6
45 Tape0 : 1 0 0 1 0 0 1
46 Head0 :      ^
47 Index1 : 0 1 2
48 Tape1 : 1 0 _
49 Head1 :      ^
50 -----
51 Step : 4
52 State : cp
53 Index0 : 0 1 2 3 4 5 6
54 Tape0 : 1 0 0 1 0 0 1
55 Head0 :      ^
56 Index1 : 0 1 2 3
57 Tape1 : 1 0 0 _
58 Head1 :      ^
59 -----
60 Step : 5
61 State : cp
62 Index0 : 0 1 2 3 4 5 6
63 Tape0 : 1 0 0 1 0 0 1
64 Head0 :      ^
65 Index1 : 0 1 2 3 4
66 Tape1 : 1 0 0 1 _
67 Head1 :      ^
68 -----
69 .....
70 -----
71 Step : 28
72 State : accept4
73 Index0 : 7 8 9 10
74 Tape0 : t r u _
75 Head0 :      ^
76 Index1 : 1
77 Tape1 : _
78 Head1 : ^
79 -----
80 Step : 29
81 State : halt_accept
82 Index0 : 7 8 9 10
83 Tape0 : t r u e
84 Head0 :      ^
85 Index1 : 1
86 Tape1 : _
87 Head1 : ^
88 -----
89 Result: true
90 ===== END =====
91
92 $ ./bin/fla tm/palindrome_detector_2tapes.tm 1001001

```

```
93 | true
94 |
95 | $ ./bin/fla tm/palindrome_detector_2tapes.tm 100100
96 | false
```

总结与感想

总体而言，这个实验还是比较简单的。可以感受到因为 PDA TM 的定义比较简洁，带来的结果是编程非常受限，但是模拟他们却非常好实现。实验的难点在于实现状态转移函数，其他则是按定义编写就可以了。这个实验即让我感受到了自动机的强大，也让我感受到了自动机的局限性。自动机的定义非常简洁，但是却能够解决很多问题，这是非常有趣的。