1 Introduction to Turing Machine Language

In this section, you are given some programs in Turing Machine Language (TML). They will be used to explain the syntax of the programming language and how they can be run on tapes.

- `isDiv2`:

```
1   // checks whether a binary number is divisible by 2
2   alphabet = {0, 1}
3   module isDiv2 {
4       // move to the end
5       while 0, 1 {
6           move right
7       } if blank {
8           move left
9           // check last letter is 0
10          if 0 {
11              accept
12          } if 1, blank {
13              reject
14          }
15      }
16  }
```

- `isDiv2Rec`:

```
1   // checks whether a binary number is divisible by 2 recursively
2   alphabet = {0, 1}
3   module isDiv2Rec {
4       // recursive case: not at the end => move closer to the end
5       if 0, 1 {
6           move right
7           goto isDiv2Rec
8       }
9       // base case: at the end => check final letter 0
10      if blank {
11          move left
12          if 0 {
13              accept
14          } if 1, blank {
15              reject
16          }
17      }
18  }
```

Both `isDiv2` and `isDiv2Rec` correspond to the following Turing Machine (TM):
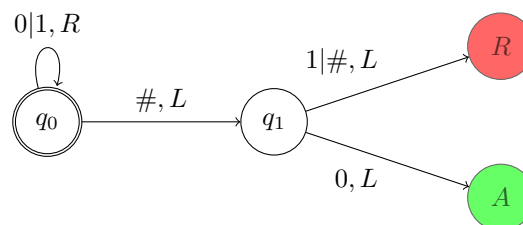
- palindrome:

```
1   // checks whether the given string is a palindrome, i.e. the string equals its reverse
2   alphabet = {a, b}
3   module palindrome {
4       if blank {
5           accept
6       }
7       // starts with a => check ends with a
8       if a {
9           changeto blank
10          move right
11          // move to the end
12          while a, b {
13              move right
14          } if blank {
15              move left
16              // cannot end with a b
17              if b {
18                  reject
19              } if a, blank {
20                  changeto blank
21                  move left
22                  goto restart
23              }
24          }
25      }
26      // starts with b => check ends with b
27      if b {
28          changeto blank
29          move right
30          // move to the end
31          while a, b {
32              move right
33          } if blank {
34              move left
35              // cannot end with an a
36              if a {
37                  reject
38              } if b, blank {
39                  changeto blank
40                  move left
41                  goto restart
42              }
43          }
44      }
45  }
46  // go to the start and restart
47  module restart {
48      while a, b {
49          move left
50      } if blank {
51          move right
52          goto palindrome
53      }
54  }
```

The program `palindrome` corresponds to the following TM:
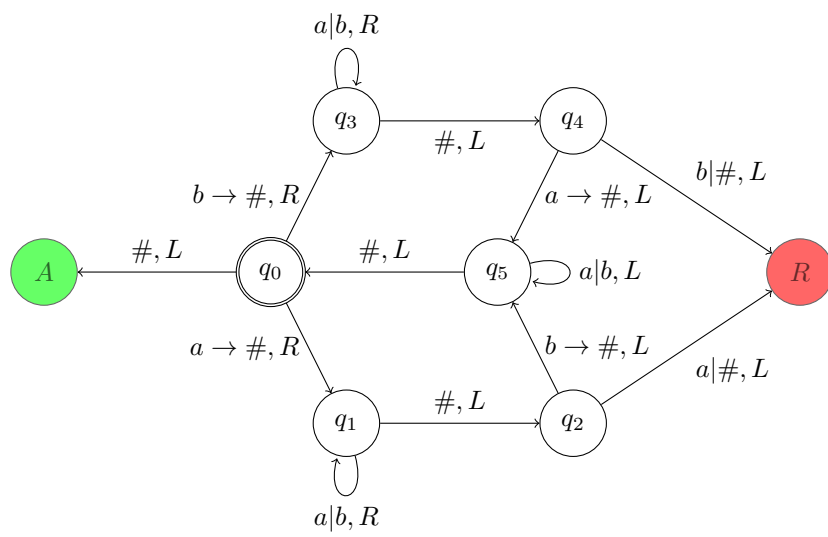
2 Identifying TML Programs

In this section, you are presented with TML programs. You will be given some tape values to run the program in and decode what values the program accepts. You are encouraged to use the website to try and solve this.

1. Consider the following TML Program:

```
1   alphabet = {0, 1}
2   module mystery1 {
3       while 0, 1 {
4           move right
5       } if blank {
6           move left
7           if blank, 0 {
8               reject
9           } if 1 {
10              move left
11              if blank, 1 {
12                  reject
13              } if 0 {
14                  accept
15              }
16          }
17      }
18  }
```

   (a) Does the program accept the values:

      i. $2 = 10$

      > **Solution:**

      ii. $1 = 1$

      > **Solution:**

      iii. $4 = 100$

      > **Solution:**

      iv. $5 = 101$

      > **Solution:**

      v. $6 = 110$

      > **Solution:**

   (b) Describe the values this program accepts.

      > **Solution:**

2. Consider the following TML program:

```
1    alphabet = {a, b}
2    module mystery2 {
3        if blank {
4            accept
5        } if a {
6            changeto blank
7            move right
8            while a, b {
9                move right
10           } if blank {
11               move left
12               if a, blank {
13                   reject
14               } if b {
15                   changeto blank
16                   move left
17                   while a, b {
18                       move left
19                   } if blank {
20                       move right
21                       goto mystery2
22                   }
23               }
24           }
25       } if b {
26           changeto blank
27           move right
28           while a, b {
29               move right
30           } if blank {
31               move left
32               if b, blank {
33                   reject
34               } if a {
35                   changeto blank
36                   move left
37                   while a, b {
38                       move left
39                   } if blank {
40                       move right
41                       goto mystery2
42                   }
43               }
44           }
45       }
46   }
```

(a) Does the program accept the values:

    i. *ab*

    ┌───┐
    │ **Solution:**   │
    │   │
    └───┘

ii. *aab*

> **Solution:**

iii. *abba*

> **Solution:**

iv. *abab*

> **Solution:**

(b) Describe the values this program accepts.

> **Solution:**

3 Identifying TMs

In this section, you are presented with TMs. You will be given some tape values to run the program in and decode what values the program accepts. Since the website can only execute TML programs, you are also given the TML program for the code, but it is not comprehensible like the previous programs; you will likely find it easier to understand the TM than the program (which you should do!).

1. Consider the following TM FSM:
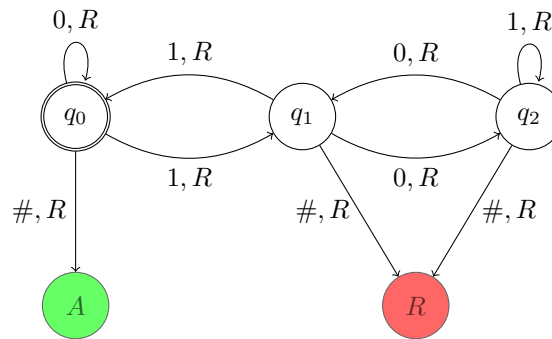
You are given a basic representation of this FSM as code in Teams.

(a) Does the TM accept the values:

  i. $3 = 11$

  > **Solution:**

  ii. $2 = 10$

  > **Solution:**

  iii. $1 = 1$

  > **Solution:**
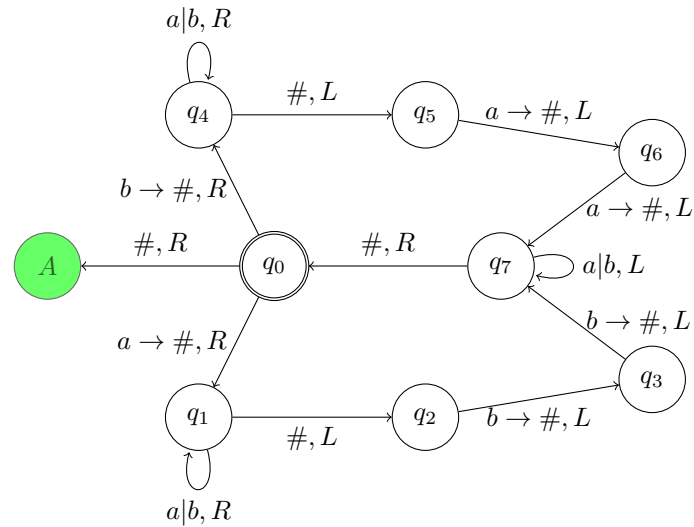
  iv. $6 = 110$

  > **Solution:**

  v. $9 = 1001$

  > **Solution:**

(b) Describe the values this program accepts.

  > **Solution:**

2. Consider the following TM FSM:

NOTE: The missing transitions go to the reject state, i.e. q_2, q_3 to $a|\#$ and q_5, q_6 to $b|\#$ are rejected. You are given a basic representation of this FSM as code in Teams.

(a) Does this TM accept the values:

    i. ab

    > **Solution:**

    ii. abb

    > **Solution:**

    iii. $aabb$

    > **Solution:**

    iv. $babbaa$

    > **Solution:**

    v. $abba$

    > **Solution:**

    vi. $abab$

    > **Solution:**

(b) Describe the values this program accepts.

    > **Solution:**

4   Writing TML Programs

Following a similar syntax to the code given above, write the following programs. You are free to use the website to check the accuracy of the program while writing the programs.

1. divisibility by 4 in binary iteratively [HINT: Go to the end and check for 2 zeros. Allow 0 as well.]

Solution:

2. divisibility by 4 in binary, recursively.

Solution:

3. check all a's come before the b's

Solution:

4. strings of the form $a^n b^n$ [HINT: this is closely related to palindrome.]

Solution:

5. HARD: check there are same number of a's and b's

Solution:

5 Appendix- Links

Below are the links to the website and the survey:

- Website: Link to Website

- Survey: Link to Survey