

ALGORITHMS

Assignment 6 Report

Group: 6

Student ID: 202020681

Student ID: 202020673

Student ID: 201921166

name: 성민규

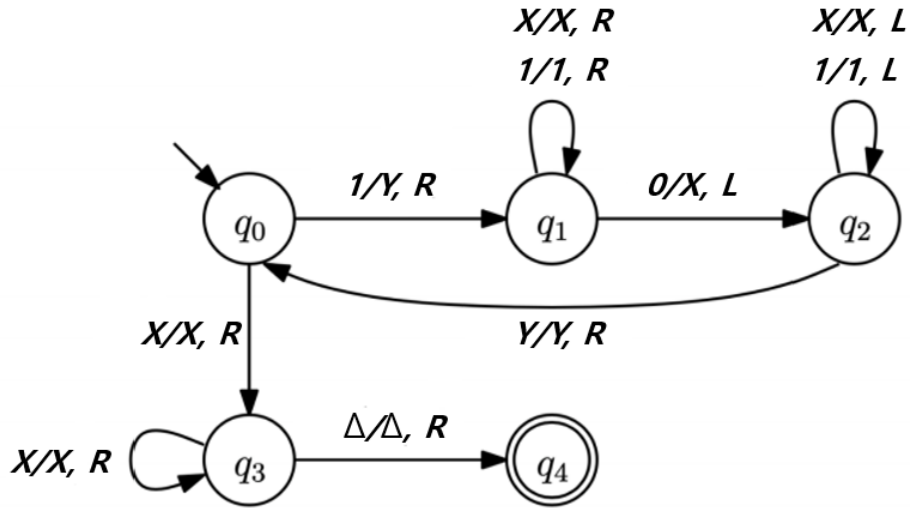
name: 안관우

name: 정의철

June 11, 2021

1 Exercise 1

1.1 Turing Machine (Total 20 point)



2 Exercise 2

A Turing machine M is defined as follows: $M = (Q, \Sigma, \delta, \Gamma, q_0, F, B)$, where

$Q = \{q_0, q_1, q_2, q_3, q_4, q_5, q_6, q_7\}$,

$\Sigma = \{0, 1\}$,

$\Gamma = \{0, 1, X, Y, Z, B\}$,

q_0 is the initial state,

$F = q_6$ is the set of final states,

B is the Blank symbol, and

δ is the transition function

$\delta(q_0, 0) = (q_1, X, R)$

$\delta(q_0, Y) = (q_4, Y, R)$

$\delta(q_1, 0) = (q_1, 0, R)$

$\delta(q_1, Y) = (q_1, Y, R)$

$\delta(q_1, 1) = (q_2, Y, R)$

$\delta(q_1, B) = (q_7, B, R)$

$\delta(q_2, 1) = (q_2, 1, R)$

$\delta(q_2, Z) = (q_2, Z, R)$

$\delta(q_2, 2) = (q_3, Z, L)$

$\delta(q_3, 0) = (q_3, 0, L)$

$$\begin{aligned} \delta(q_3, 1) &= (q_3, 1, L) \\ \delta(q_3, Y) &= (q_3, Y, L) \\ \delta(q_3, Z) &= (q_3, Z, L) \\ \delta(q_3, X) &= (q_0, X, R) \\ \delta(q_4, Y) &= (q_4, Y, R) \\ \delta(q_4, Z) &= (q_5, Z, R) \\ \delta(q_5, Z) &= (q_5, Z, R) \\ \delta(q_5, B) &= (q_6, B, R) \\ \delta(q_7, B) &= (q_7, B, R) \end{aligned}$$

3 Exercise 3

프로그램 P 가 입력값 i 를 받았을 때 halt 하는지 알려주는 프로그램 $Halt$ 를 만들 수 있다고 가정한다. 프로그램 $Halt$ 는 P 가 infinite loop에 빠졌을 경우 $False$ 를 반환하고, halt 할 경우 $True$ 를 반환한다. 이 때, P 는 아래의 동작을 수행한다.

3.1

```
def P(i):
    if Halt(P, i) == True:
        while True:
            do something ...
    else:
        return 0
```

$Halt(P, P)$ 를 실행한다고 하자. 그렇다면, 이 프로그램은 $P(P)$ 가 halt할 경우 $True$ 를 반환할 것이고, $P(P)$ 가 infinite loop에 빠질 경우 $False$ 를 반환할 것이다.

1. $Halt(P, P)$ 가 $True$ 를 반환할 경우

$Halt(P, P)$ 가 $True$ 를 리턴하므로 $P(P)$ 가 halt해야 한다. 하지만 $P(i)$ 프로그램의 if문에서 $Halt(P, P)$ 가 $True$ 조건에 해당하므로 infinite loop에 빠지게 된다. 이럴 경우 $P(P)$ 는 halt하지 않고, infinite looping 하므로 $Halt(P, P)$ 의 반환값은 전제에 모순된다.

2. $Halt(P, P)$ 가 $False$ 를 반환할 경우

$Halt(P, P)$ 가 $False$ 를 리턴하므로 $P(P)$ 가 infinite looping 해야 한다. 하지만 $P(i)$ 프로그램의 if문에서 $Halt(P, P)$ 가 $False$ 조건에 해당하므로 반환값 0과 함께 halt 하게 된다. 이럴 경우 $P(P)$ 는 infinite looping하지 않고, halt 하므로 $Halt(P, P)$ 의 반환값은 전제에 모순된다.

위 두가지 모두에서 전제가 모순되므로, 프로그램 P 가 halt하는지 알려주는 프로그램 $Halt$ 를 만들 수 있다는 가정은 기각된다.

강의 내용에서는 $\text{Halt}(P, P)$ 가 True를 반환할 경우 Test가 infinite looping 한다고 가정하였습니다. 그 반대의 경우에는 Test가 halt 한다고 가정하였으나, 실제 Test 프로그램의 코드는 두 경우에서 모두 다항시간 내에 True 또는 False를 반환하도록 작성되어 있었습니다. 이 부분이 잘 이해가 되지 않아서, $P(i)$ 프로그램이 while True를 통해서 infinite looping을 하도록 코드를 수정하였습니다. 반대의 경우에는 다항시간 내에 0을 반환하도록 하여, 제가 이해한 대로 문제를 풀이하였습니다.