IUBAT – International University of Business Agriculture and Technology

Assignment No: 2

Title of Assignment:

Program for PDA capable of recognizing the

language w#w^R where w ∈ {0, 1}* and ∑={0, 1, #}

Course Name: Theory of Computation
Course Code: CSC 397

Submitted To:

Dr. Ehteshamul Haque
Faculty, Dept. of CSE

Submitted By:

Naimul Ferdous
ID: 13303042
Sec-C, CSC397
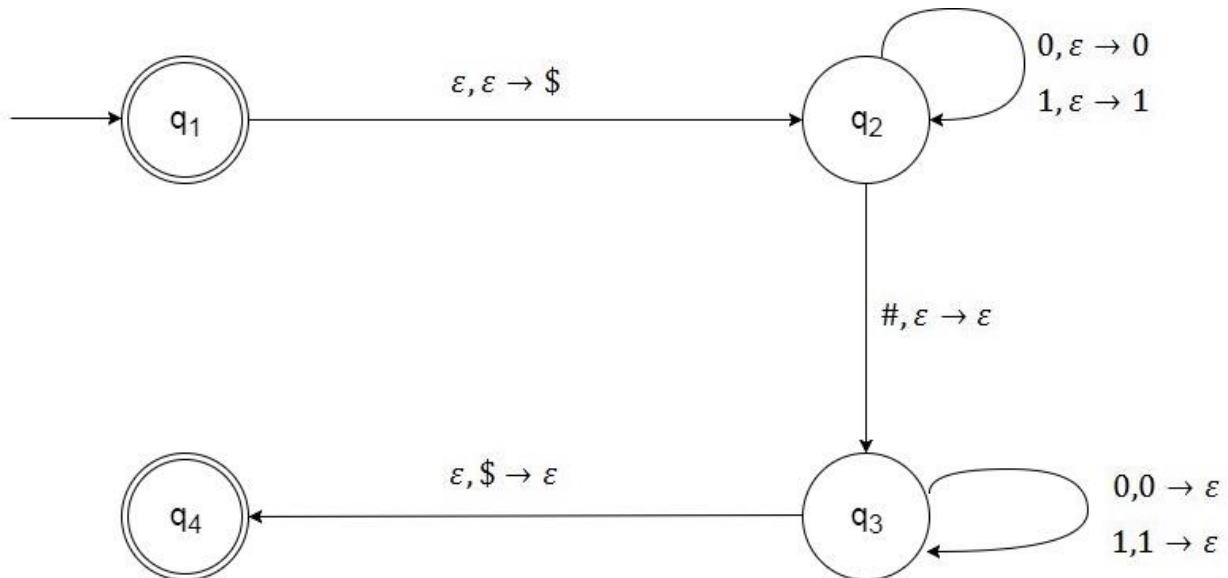
Submission Date: 21.11.2017

Formal definition of pushdown automaton (PDA)

A pushdown automaton is a 6-tuple $(Q, \Sigma, \Gamma, \delta, q_0, F)$, where Q, Σ, Γ, F are all finite sets, and

1. Q is the set of states,
2. Σ is the input alphabet,
3. Γ is the stack alphabet,
4. $\delta: Q \times \Sigma_\varepsilon \times \Gamma_\varepsilon \to P(Q \times \Gamma_\varepsilon)$ is the transition function,
5. $q_0 \in Q$ is the start state, and
6. $F \subseteq Q$ is the set of accept states.

- Alphabets $\Sigma_\varepsilon = \Sigma \cup \{\varepsilon\}$ and $\Gamma_\varepsilon = \Gamma \cup \{\varepsilon\}$

- P( ) accounts for the non-determinism

State Diagram for w#w^R

Here,

P = {Q, Σ, Γ, δ, q_0, F}, where

1. Q = {q_1, q_2, q_3, q_4},
2. Σ = {0, 1, #},
3. Γ = {0, 1, $}$,
4. q_s = q_1
5. F = {q_l, q_4},

δ is given by the following transition table, cell entries are new state and TOS symbol pair, blank entries are $\varnothing$.

Inputs are current state, input string and TOS symbols. Cell entries are new state and TOS symbol pair. Blank entries are $\varnothing$.

$$\boxed{\begin{array}{l} \text{Input } \Sigma \text{ U } \varepsilon \\ \text{Stack } \Gamma \text{ U } \varepsilon \end{array}}$$

Transition Table:

Input:	0				1				#				ε			
Stack:	**0**	**1**	**$**	ε	**0**	**1**	**$**	ε	**0**	**1**	**$**	ε	**0**	**1**	**$**	ε
q_1																{q_2,$}
q_2				{q_2,0}				{q_2,1}				{q_3,ε}				
q_3	{q_3,ε}				{q_3,ε}										{q_4,ε}	
q_4																

Program

```c
#include<stdio.h>
#define EOS '\0'
#define SIZE 101

char Stck[SIZE];
int top=-1;

void push(char symb)
{
    Stck[++top]= symb;
}

char pop(void)
{
    return  Stck[top--];
}

int main()
{
    char c,inpstr[SIZE];
    int q,i;
    scanf("%s",inpstr);
    printf("\n");
    printf("%s ",inpstr);
    q=1;
    i=0;
    c=inpstr[i];
    while(true)
    {
        if(q==1)
        {
            q=2;
            push('$');
        }
        if(q==2)
        {
            if(c=='0')
            {
                q=2;
                push(c);
            }
            else  if(c=='1')
```

```
                {
                    q=2;
                    push(c);
                }
                else  if(c=='#')
                {
                    q=3;
                }
                else
                {
                    q=5;
                    break;
                }
            }
        else  if(q==3)
        {
            if(c=='0'  &&  Stck[top]=='0')
            {
                q=3;
                pop();
            }
            else if(c=='1'  &&  Stck[top]=='1')
            {
                q=3;
                pop();
            }
            else if(c==EOS  &&  Stck[top]=='$')
            {
                q=4;
                break;
            }
            else
            {
                q=5;
                break;
            }
        }
        else
        {
            q=5;
            break;
        }
        i++;
        c=inpstr[i];
    }
```

```
  if(q==5)
    printf("   Rejected");
  if(q==4)
    printf("   Accepted");

  printf("\n");

  return  0;
}
```

I/O:

Input: **101011#110101**
Output: **Accepted**

Input: **1010#1010**
Output: **Rejected**

Input: **11#00**
Output: **Rejected**