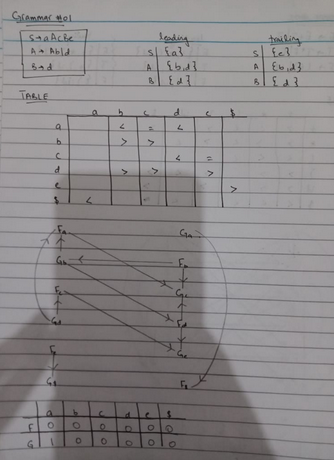
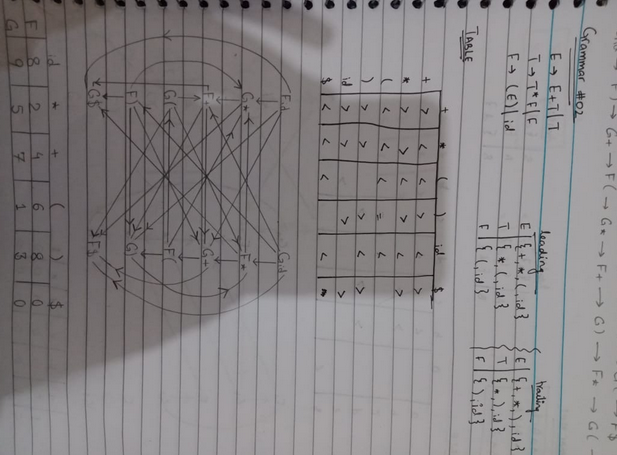
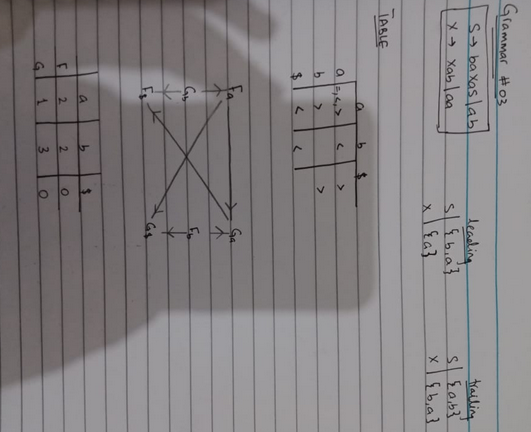
Compiler Construction

Lab 10







# CODE:

#include<iostream>

#include<string> #include<deque> using namespace std; int n,n1,n2; int getPosition(string arr[], string q, int size)

{ for(int i=0;i<size;i++)

{ if(q == arr[i]) return i;

}

return -1;

}

int main()

{

string prods[10],leads[10],trails[10],nonterms[10],terms[10]; char op\_table[20][20] = {}; cout<<"Enter the number of productions : "; cin>>n; cin.ignore(); cout<<"Enter the productions"<<endl; for(int i=0;i<n;i++)

{

getline(cin,prods[i]);

}

cout<<"Enter the number of Terminals : "; cin>>n2; cin.ignore(); cout<<"Enter the Terminals"<<endl; for(int i=0;i<n2;i++)

{

cin>>terms[i];

}

terms[n2] = "$"; n2++; cout<<"Enter the number of Non-Terminals : "; cin>>n1; cin.ignore(); for(int i=0;i<n1;i++)

{

cout<<"Enter Non-Terminal : "; getline(cin,nonterms[i]); cout<<"Enter Leads of "<<nonterms[i]<<" : "; getline(cin,leads[i]); cout<<"Enter Trails of "<<nonterms[i]<<" : "; getline(cin,trails[i]);

}

cout<<"Enter the Rules (exit to stop)"<<endl; string rule = ""; while(rule != "exit")

{

getline(cin,rule); if(rule[0] == '1')

{

int row = getPosition(terms,rule.substr(2,1),n2); int column = getPosition(terms,rule.substr(4,1),n2); op\_table[row][column] = '=';

}

if(rule[0] == '2')

{

int ntp = getPosition(nonterms,rule.substr(4,1),n1); int row = getPosition(terms,rule.substr(2,1),n2); for(int j=0;j<leads[ntp].size();j++)

{

int col = getPosition(terms,leads[ntp].substr(j,1),n2); op\_table[row][col] = '<';

}

}

if(rule[0] == '3')

{

int col = getPosition(terms,rule.substr(4,1),n2); int ntp = getPosition(nonterms,rule.substr(2,1),n1); for(int j=0;j<trails[ntp].size();j++)

{

int row = getPosition(terms,trails[ntp].substr(j,1),n2); op\_table[row][col] = '>';

}

}

}

for(int j=0;j<leads[0].size();j++)

{

int col = getPosition(terms,leads[0].substr(j,1),n2); op\_table[n2-1][col] = '<';

}

for(int j=0;j<trails[0].size();j++)

{

int row = getPosition(terms,trails[0].substr(j,1),n2); op\_table[row][n2-1] = '>';

}

cout<<endl; cout<<"Grammar"<<endl;

for(int i=0;i<n;i++)

{

cout<<prods[i]<<endl;

}

//Display Table for(int j=0;j<n2;j++) cout<<"\t"<<terms[j]; cout<<endl; for(int i=0;i<n2;i++)

{

cout<<terms[i]<<"\t"; for(int j=0;j<n2;j++)

{

cout<<op\_table[i][j]<<"\t";

}

cout<<endl;

}

return 0;

