Mini Project 1 Scanner

Compilers

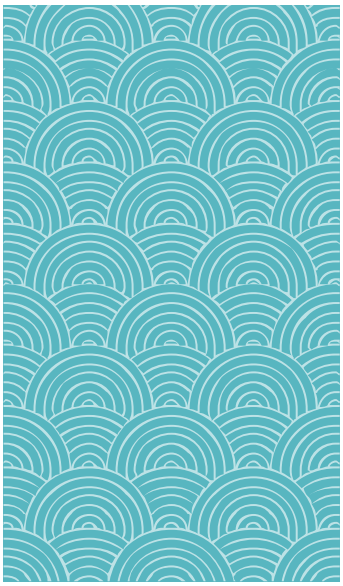
By:

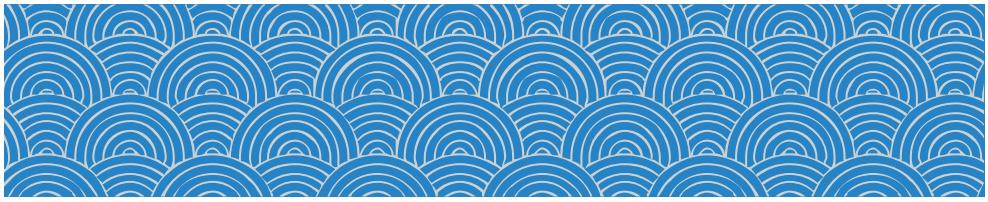
Amira Muhammad Fareed

Basma Saeed Ragab

Section: 1

CONTENTS





PRE CONDITIONS:

- Put the Input in the same directory of the Scanner.exe and name it "input.txt"

POST CONDITIONS:

- The output will be generated to a csv file called "output.csv"

STEPS:

1. Click on Scanner.exe
2. Check "output.csv" to see the output

EXAMPLE:

Input:

```
read x;   {input an integer }
if  0 < x
then    {  don't compute if x <= 0 }
fact  := 1;
repeat
fact  := fact *  x;
x  := x  -  1
until  x  =  0;
write  fact   {  output  factorial of x }
end
```

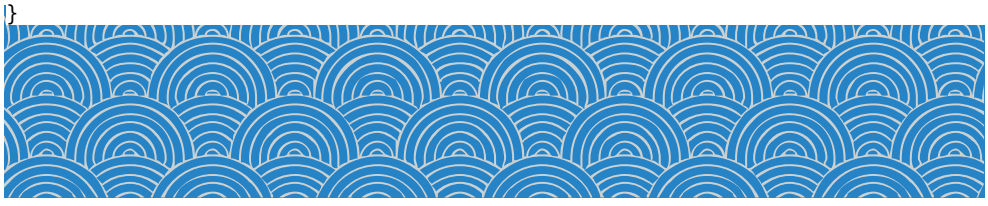
Output:

Token Value	Token Type	Details
read	Reserved Word	
x	Identifier	
;	Special Symbol	Semicolon
if	Reserved Word	
0	Number	
<	Special Symbol	Less than
x	Identifier	
then	Reserved Word	
fact	Identifier	
:=	Special Symbol	assign
1	Number	
;	Special Symbol	Semicolon
repeat	Reserved Word	
fact	Identifier	
:=	Special Symbol	assign
fact	Identifier	
*	Special Symbol	Asterisk
x	Identifier	
;	Special Symbol	Semicolon
x	Identifier	
:=	Special Symbol	assign
x	Identifier	
-	Special Symbol	Minus
1	Number	
until	Reserved Word	
x	Identifier	
=	Special Symbol	Equal sign
0	Number	
;	Special Symbol	Semicolon
write	Reserved Word	
fact	Identifier	
end	Reserved Word	

CODE:

```
#include <iostream>
#include <algorithm>
#include <vector>
#include <string>
#include <fstream>
#include <sstream>

using namespace std;

void Trim(string& s)
{
	for (int i = 0; i < s.length(); i++)
	{
		if (s[i] == ' ' && s[i + 1] == ' ')
		{
			s.erase(i, 1);
			i--;
		}
	}
}

void EditNewLineCharacter(string& s)
{
	for (int i = 0 ; i<s.length() ;i++)
	{
		if (s[i] == '\n')
		{
			s[i]= ' ';
		}

		if (s[i] == ';'||s[i] == '+' || s[i] == '-' || s[i] == '*'||s[i] == '/'
			|| s[i] == '>' || s[i] == '<' || s[i] == '(' || s[i] == ')' )
		{
			s.insert(i+1," ");
			s.insert(i," ");
			i+=2;
		}
		if(s[i] == '=')
		{
			if(s[i-1] == ':')
			{
				s.insert(i+1," ");
				s.insert(i-1," ");
				i+=2;
			}
			else
			{
				s.insert(i+1," ");
				s.insert(i," ");
				i+=2;
			}
		}
	}
}
void RemoveComments(string& s)
{
	int end ;
	for (int i = 0 ; i<s.length() ;i++)
	{
		if(s[i] == '{')
		{
			end = i;
			while(s[end] != '}')
			{
				end++;
			}
			s.erase(i ,end-i+1);
			i--;
		}
	}
}
```

```
void split(string &s , vector<string> &elems) {
    stringstream ss(s);
    string item;
    while (getline(ss, item, ' ')) {
        elems.push_back(item);
    }
}

bool checkReservedWord(string word)
{
        string ReservedWords[8]
={"if" ,"then","else","end","repeat","until" ,"read","write"};
        for(int i = 0 ; i<8; i++)
        {
                if(word == ReservedWords[i])
                        return 1 ;
        }
        return 0 ;
}

string checkSpecialSymbols(string word)
{
        if(word == "+")
                return "Plus";

        else if (word == "-")
                return "Minus";

        else if (word == "*")
                return "Asterisk";

        else if (word == "/")
                return "Slash";

        else if (word == "=")
                return "Equal sign";

        else if (word == ">")
                return "Greater than";

        else if (word == "<")
                return "Less than";

        else if (word == "(")
                return "Minus";

        else if (word == ")")
                return "Minus";

        else if (word == ";")
                return "Semicolon";

        else if (word == ":=")
                return "assign";

        else
                return "none";
}
```



```
bool IsNumber(string word)
{
	int i =0 ;
	while(isdigit(word[i]))
		i++;
	if (i>0)
		return 1;
	return 0;
}

int main()
{
	vector<string> words ;
	ifstream ip ("input.txt");
	ofstream op ("output.csv");
	string txtIp((istreambuf_iterator<char>(ip)),istreambuf_iterator<char>());
	RemoveComments(txtIp);
	EditNewLineCharacter(txtIp);
	Trim(txtIp);
	split(txtIp,words);
	op <<"Token Value ,Token Type ,Details \n" ;
	for(int i =0 ; i<words.size() ;i++)
	{
		if(checkReservedWord(words[i]))
			op << words[i] <<", Reserved Word \n" ;

		else if(checkSpecialSymbols(words[i]) != "none")
			op << words[i] <<", Special Symbol, "
	<< checkSpecialSymbols(words[i]) << endl;

		else if (IsNumber(words[i]))
			op << words[i] << ", Number \n" ;
		else
			op << words[i] << ", Identifier \n";

	}
	system("pause");
	return 0;
}
```