PROGRAMMING ASSIGNMENT #1

NAME	**LAVANYA SARAVANAN**
OU ID	**113443485**
COURSE NUMBER	**CS 4323**
COURSE NAME	**COMPILER CONSTRUCTION**

```
package Compiler;

import java.io.BufferedReader;
import java.util.*;
import java.util.regex.Pattern;
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.IOException;
public class Trumpscript
{

	static String file="",say1="";static char nfch = 0;
	static int flag;static char special=0;
	static Map<String,String> SYMTAB = new HashMap<String,String>();

//……BOOKKEEPER STORES TOKEN AND ITS ATTRIBUTES IN THE SYMBOL TABLE……

	static class BOOKKEEPER{
	public static void BOOKKEEPER1(String type,String token)
	{
		SYMTAB.put(type.toLowerCase(), token);
	}
	}

//..…………...SCANNER CONSUMES SYMBOL, FINDS TOKEN AND ITS ATTRIBUTE………….

	static class SCANNER{
	public static void SCANNER1(char ch)
	{
		if(ch=='#'|| flag==1)
		{
			if(ch!='\n')
			{flag=1;
			return;}
			else
			{flag=0;
			return;}
		}
		else
		{
		if(ch==' '||ch=='$'||ch=='\b'||ch=='\t'||ch=='\r'||ch=='\n')
		{
			if(file!=""&&file!=" ")
			{
					nfch=0;
					if(file.contains(",")==true)
					{
						nfch=',';
						String[] sp=file.split(",");
						for(String temp:sp)
							file=temp;
		}
		else if(file.contains(":")==true)
		{
			nfch=':';
			String[] sp=file.split(":");
			for(String temp:sp)
				file=temp;
		}
		else if(file.contains(";")==true)
		{
			nfch=';';
			String[] sp=file.split(";");
			for(String temp:sp)
				file=temp;
		}
		else if(file.contains("?")==true)
		{
			nfch='?';
			String[] sp=file.split("\\?");
			for(String temp:sp)
				file=temp;
		}
		else if(file.contains("!")==true)
		{
			nfch='!';
			String[] sp=file.split("!");
			for(String temp:sp)
				file=temp;
		}
		else if(file.contains("(")==true)
		{
			nfch='(';
			String[] sp=file.split("\\(");
			for(String temp:sp)
				file=temp;
		}
		else if(file.contains(")")==true)
		{
			nfch=')';
			String[] sp=file.split("\\)");
			for(String temp:sp)
				file=temp;
		}
		else if(file.contains("\""))
		{
			String check_double=Pattern.quote("\"");
			String[] check_double1=file.split(check_double,2);
	for(String temp:check_double1)
	{
		if(temp.contains("\""))
		{
```

```
                                        String[] say=temp.split("\"");
                                        for(String temp1:say)
                                                say1=temp1;
                                }
                                else
                                        file=temp;
                        }
                }
                }
```

//….IMPLEMENTATION OF THE DFA FOR FINDING KEYWORDS, IDENTIFIER,CONSTANT…..

```
                char [] iptoken= file.toCharArray();
                int iptokenl= iptoken.length;
                int i=0;
                if(iptokenl!=0)
                {
                if(iptoken[0]!=','&&iptoken[0]!=';'&&iptoken[0]!=':'&&iptoken[0]!='('
                        &&iptoken[0]!=')'&&iptoken[0]!='?'&&iptoken[0]!='!'&&iptokenl==1)
                        check_special(i,iptoken);
                else
                {
                        switch(iptoken[i])
                        {
                        case 'm': case 'M':
                                switch(iptoken[i+1])
                                {
                                case 'a': case'A':
                                        switch(iptoken[i+2])
                                        {
                                        case 'k':case'K':
                                                if((iptoken[i+3]=='e'||iptoken[i+3]=='E')&&
                                                (iptokenl==4))
                                                {
                                                        System.out.println("keyword "+file);}
                                                else
                                                        check_special(i+3,iptoken);
                                                break;
                                                        default: check_special(i+2,iptoken);
                                                                break;
                                                }
                                        break;
                                        case 'o':case'O':
                                                switch(iptoken[i+2])
                                                {
                                                        case 'r':case'R':
                                                        if((iptoken[i+3]=='e'||iptoken[i+3]=='E')&& (
                                                        iptokenl==4))
                                                        {
                                                                System.out.println("keyword "+file);
                                                        }
                                        else
                                                    check_special(i+3,iptoken);
                              break;
                              default: check_special(i+2,iptoken);
                                          break;
                              }
                              break;
                              default: check_special(i+1,iptoken);
                                          break;
                    }
                    break;
          case 'p':case'P':
                    switch(iptoken[i+1])
                    {
                    case 'r':case'R':
                              switch(iptoken[i+2])
                              {
                              case 'o':case'O':
                                        switch(iptoken[i+3])
                                        {
                                        case 'g':case'G':
                                                  switch(iptoken[i+4])
                                                  {
                                                  case 'r':case'R':
                                                            switch(iptoken[i+5])
                                                            {
                                                            case 'a':case'A':
                                                                      switch(iptoken[i+6])
                                                                      {
                                                                      case 'm':case'M':
                                                                                switch(iptoken[i+7])
                                                                                {
                                                                                case 'm':case'M':
                                                                                switch(iptoken[i+8])
                                                                                {
                                                                                          case 'i':case'I':
                                                                                switch(iptoken[i+9])
                                                                                {
                                                                                case 'n':case'N':
                                                                      if((iptoken[i+10]=='g'||
                                                                                iptoken[i+10]=='G')
                                                                                &&iptokenl==11)
                                                                      {
                                                                                System.out.println
                                                                                ("keyword "+file);
                                                                      }
                                                                      else
                                                                      check_special(i+10,iptoken);
                                                                      break;
                                                            default: check_special(i+9,iptoken);
                                                                      break;
                                                            }
                                                            break;
                                                    default: check_special(i+8,iptoken);
                                                              break;
                                                    }
                                                    break;
                                                    default: check_special(i+7,iptoken);
                                                              break;
                                            }
                                            break;
                                            default: check_special(i+6,iptoken);
                                                      break;
                                    }
                                    break;
                                    default: check_special(i+5,iptoken);
                                              break;
                            }
                            break;
                            default: check_special(i+4,iptoken);
                                      break;
                    }
                    break;
                    default: check_special(i+3,iptoken);
                              break;
            }
            break;
            default: check_special(i+2,iptoken);
                      break;
            }
            break;
            case 'l':case'L':
                    switch(iptoken[i+2])
                    {
                            case 'u':case'U':  if((iptoken[i+3]=='s'||iptoken[i+3]=='S')&& iptokenl==4)
                                            {
                                                    System.out.println("keyword "+file);
                                            }
                                            else
                                                    check_special(i+3,iptoken);
                                            break;
                            default: check_special(i+2,iptoken);
                                    break;
                                    }
                            break;
                            default: check_special(i+1,iptoken);
                                      break;
                            }
                            break;
                            case 'g':case'G':
```

```
switch(iptoken[i+1])
{
        case 'r':case'R':
        switch(iptoken[i+2])
        {
                case 'e':case'E':
                switch(iptoken[i+3])
                                {
                                case 'a':case'A':
                        if((iptoken[i+4]=='t'||iptoken[i+4]=='T')&&iptokenl==5)
                        {
                                System.out.println("keyword "+file);
                        }
                        else
                                check_special(i+4,iptoken);
                                break;
                        default: check_special(i+3,iptoken);
                                }
                        break;
                        default: check_special(i+2,iptoken);
                                 break;
                }
                break;
                default: check_special(i+1,iptoken);
                         break;
                }
        break;
        case 'a':case'A':
                switch(iptoken[i+1])
                        {
                        case 'g':case'G':
                                switch(iptoken[i+2])
                                {
                                case 'a':case'A':
                                        switch(iptoken[i+3])
                                        {
                                        case 'i':case'I':
                if((iptoken[i+4]=='n'||iptoken[i+4]=='N')&&iptokenl==5)
                {
                        System.out.println("keyword "+file);
                }
                else
                        check_special(i+4,iptoken);
                        break;
                        default: check_special(i+3,iptoken);
                                 break;
                }
                break;
                default: check_special(i+2,iptoken);
                         break;
        }
```

```
                break;
                case 'm':case'M':
                        switch(iptoken[i+2])
                        {
                        case'e':case'E':
                                switch(iptoken[i+3])
                                {
                                case 'r':case'R':
                                        switch(iptoken[i+4])
                                        {
                                        case 'i':case'I':
                                                switch(iptoken[i+5])
                                                {
                                                case'c':case'C':
                                if((iptoken[i+6]=='a'||iptoken[i+6]=='a')&&iptokenl==7)
                                {
                                        System.out.println("keyword "+file);
                                }
                                else
                                        check_special(i+6,iptoken);
                                        break;
                                        default: check_special(i+5,iptoken);
                                                 break;
                                }
                                break;
                                default: check_special(i+4,iptoken);
                                         break;
                        }
                        break;
                        default: check_special(i+3,iptoken);
                                 break;
                }
                break;
                default: check_special(i+2,iptoken);
                         break;
        }
        break;
        case'n':case'N':  if((iptoken[i+2]=='d'||iptoken[i+2]=='D')&&iptokenl==3)
                        {
                                System.out.println("keyword "+file);
                        }
                        else
                                check_special(i+2,iptoken);
                                break;
                                case 's':case'S':
                                        if(iptokenl==2)
                        {
                                System.out.println("keyword "+file);
                        }
                        else
                                check_special(i+2,iptoken);
                break;
                default: check_special(i+1,iptoken);
                         break;
}
break;
case'i':case'I':
        if((iptoken[i+1]=='s'||iptoken[i+1]=='S')&&iptokenl==2)
        {
                System.out.println("keyword "+file);
        }
        else if((iptoken[i+1]=='f'||iptoken[i+1]=='F')&&iptokenl==2)
        {
                System.out.println("keyword "+file);
        }
        else
                check_special(i+1,iptoken);
        break;
        case'e':case'E':
                switch(iptoken[i+1])
                {
                case'l':case'L':
                        switch(iptoken[i+2])
                        {
                        case 's':case'S':
                if((iptoken[i+3]=='e'||iptoken[i+3]=='E')&&iptokenl==4)
                {
                        System.out.println("keyword "+file);
                }
                else
                        check_special(i+3,iptoken);
                        break;
                        default: check_special(i+2,iptoken);
                                 break;
        }
        break;
        default: check_special(i+1,iptoken);
                 break;
}
break;
case 'n':case'N':
        switch(iptoken[i+1])
        {
case'u':case'U':
        switch(iptoken[i+2]) {
        case'm':case'M':
                switch(iptoken[i+3])
                {
                case'b':case'B':
                        switch(iptoken[i+4])
                        {
```

```
                                        case 'e':case'E':
                                if((iptoken[i+5]=='r'||iptoken[i+5]=='R')&&iptokenl==6)
                                {
                                        System.out.println("keyword "+file);
                                }
                                else
                                        check_special(i+5,iptoken);
                                        break;
                                        default: check_special(i+4,iptoken);
                                                 break;
                                }
                                break;
                                default: check_special(i+3,iptoken);
                                         break;
                        }
                        break;
                        default: check_special(i+2,iptoken);
                                 break;
                }
                break;
                case'o':case'O':  if((iptoken[i+2]=='t'||iptoken[i+2]=='R')&&iptokenl==3)
                                {
                                        System.out.println("keyword "+file);
                                }
                                else
                                        check_special(i+2,iptoken);
                                        break;
                                        default: check_special(i+1,iptoken);
                                                         break;
                                        }
                                break;
                                case 'b':case'B':
                                        switch(iptoken[i+1])
                                        {
                                        case'o':case'O':
                                                switch(iptoken[i+2])
                                                {
                                                case 'o':case'O':
                                                        switch(iptoken[i+3])
                                                        {
                                                        case'l':case'L':
                                        switch(iptoken[i+4])
                                                {
                                                case'e':case'E':
                                                        switch(iptoken[i+5])
                                                        {
                                                        case 'a':case'A':
if((iptoken[i+6]=='n'||iptoken[i+6]=='N')&&iptokenl==7)
{
        System.out.println("keyword "+file);
}
```

```
            else
                        check_special(i+6,iptoken);
                        break;
                        default: check_special(i+5,iptoken);
                                    break;
            }
            break;
            default: check_special(i+4,iptoken);
                        break;
}
break;
default: check_special(i+3,iptoken);
            break;
}
break;
default: check_special(i+2,iptoken);
            break;
}
break;
default: check_special(i+1,iptoken);
            break;
}
break;
            case 'l':case'L':
                        switch(iptoken[i+1])
                        {
                                    case'e':case'E':
                                    switch(iptoken[i+2])
                                    {
                                                case 's':case'S':
                                                if((iptoken[i+3]=='s'||iptoken[i+3]=='S')&&iptokenl==4)
                                                {
                                                            System.out.println("keyword "+file);
                                                }
                                                else
                                                            check_special(i+3,iptoken);
                                                            break;
                                                default: check_special(i+2,iptoken);
                                                            break;
                                                }
                                                break;
                                                case'o':case'O':
                                                            switch(iptoken[i+2])
                                                            {
                                                            case 'n':case'N':
                                                if((iptoken[i+3]=='g'||iptoken[i+3]=='G')&&iptokenl==4
                                                {
                                                            System.out.println("keyword "+file);
                                                }
                                                else
                                                            check_special(i+3,iptoken);
```

```
                    break;
                    default: check_special(i+2,iptoken);
                               break;
}
break;
case 'i':case'I': if((iptoken[i+2]=='e'||iptoken[i+2]=='E')&&iptokenl==3)
                 {
                              System.out.println("keyword "+file);
                    }
                    else
                              check_special(i+2,iptoken);
                              break;
                              default: check_special(i+1,iptoken);
                                         break;
                    }
          break;
          case 't':case'T':
                    switch(iptoken[i+1])
                    {
                    case'e':case'E':
                              switch(iptoken[i+2])
                              {
                              case 'l':case'L':
                    if((iptoken[i+3]=='l'||iptoken[i+3]=='L')&&iptokenl==4)
                    {
                              System.out.println("keyword "+file);
                    }
                    else
                              check_special(i+3,iptoken);
                              break;
                              default: check_special(i+2,iptoken);
                                                   break;
          }
          break;
          case'i':case'I':
                    switch(iptoken[i+2])
          {
          case'm':case'M':
                    switch(iptoken[i+3])
                    {
                    case'e':case'E':
          if((iptoken[i+4]=='s'||iptoken[i+4]=='S')&&iptokenl==5)
          {
                    System.out.println("keyword "+file);
          }
          else
                    check_special(i+4,iptoken);
                    break;
                    default: check_special(i+3,iptoken);
                               break;
}
```

```
                break;
                default: check_special(i+2,iptoken);
                         break;
        }
        break;
        default: check_special(i+1,iptoken);
                 break;
}
break;
case's':case'S':
        switch(iptoken[i+1])
        {
                case 'a':case'A': if((iptoken[i+2]=='y'||iptoken[i+2]=='Y')&&iptokenl==3)
                                {
                                        System.out.println("keyword "+file);
                                }
                                else
                                        check_special(i+2,iptoken);
                                        break;
                                        default: check_special(i+1,iptoken);
                                                 break;
        }
        break;
        case'f':case'F':
                switch(iptoken[i+1])
                {
                case 'a':case'A':
                        switch(iptoken[i+2])
                        {
                        case'c':case'C':
                        If((iptoken[i+3]=='t'||iptoken[i+3]=='T')&&iptokenl==4)
                        {
                                System.out.println("keyword "+file);
                        }
                        else
                                check_special(i+3,iptoken);
                                break;
                                default: check_special(i+2,iptoken);
                                         break;
                }
                break;
                default: check_special(i+1,iptoken);
                         break;
        }
        break;
        case'o':case'O': if((iptoken[i+1]=='r'||iptoken[i+1]=='R')&&iptokenl==2)
                        {
                                System.out.println("keyword "+file);
                        }
                        else
                                check_special(i+1,iptoken);
```

```
                                        break;
                                        case '1':case '2':case '3':case '4':case '5':
                                             case '6':case '7':case '8':case '9':
                                        int d=0;
                                        for(int n=0;n<iptokenl;n++)
                                        {
                                             if(!Character.isDigit(iptoken[n]))
                                                       d=1;
                                        }
                                        if(d==1)
                                          ERRORHANDLER.ERRORHANDLER1(file,2);
                                        if(d==0)
                                        {
                                             int num=Integer.parseInt(file);
                                             if(num>1000000)
                                                  {
                                                  BOOKKEEPER.BOOKKEEPER1
                                                       (file,"CONSTANT");
                                                  System.out.println("constant "+file);
                                                  }
                                             else
                                                  ERRORHANDLER.
                                                       ERRORHANDLER1(file,2);
                                        }
                                        break;
                                   default:
                                        int c=0,f=0;
                                        if(iptoken[0]==','||iptoken[0]==';'||iptoken[0]==':'||
                                             iptoken[0]=='('||iptoken[0]==')'||iptoken[0]=='?'||
                                                  iptoken[0]=='!'||iptokenl==1)
                                        {
                                             f=1;
                                             break;
                                        }
if(f==0)
{
     for(int n=0;n<iptokenl;n++)
     {
          if((!Character.isDigit(iptoken[n]))&& (!Character.isAlphabetic(iptoken[n])))
               c=1;
     }
     if(c==1)
          ERRORHANDLER.ERRORHANDLER1(file,1);
     else
          {
               BOOKKEEPER.BOOKKEEPER1(file,"IDENTIFIER");
               System.out.println("identifier "+file);
          }
     }
     break;
     }
```

```
//……………………PRINTS THE SPECIAL SYMBOLS IN THE PROGRAM…………………….

                switch(nfch)
                {
                        case',': System.out.println("Special symbol "+nfch);
                                break;
                        case':': System.out.println("Special symbol "+nfch);
                                break;
                        case';': System.out.println("Special symbol "+nfch);
                                break;
                        case'?': System.out.println("Special symbol "+nfch);
                                break;
                        case'(': System.out.println("Special symbol "+nfch);
                                        break;
                        case')': System.out.println("Special symbol "+nfch);
                                break;
                        case'!': System.out.println("Special symbol "+nfch);
                                break;
                        default: break;
                }
                if(say1!="")
                {
                        System.out.println("String "+say1);
                        BOOKKEEPER.BOOKKEEPER1(say1,"STRING");
                        say1="";
                }}
        file="";
        }}
        else
                file=file+Character.toString(ch);
        }
        public static void check_special(int index, char[] word)
        {
        int flag1=0;
        for(int j=index;j<word.length;j++){
                if(word[j]==','||word[j]=='?'||word[j]==':'||word[j]==';'||word[j]=='('||word[j]==')')
                flag1=1;
        }
        if(flag1==1)
                ERRORHANDLER.ERRORHANDLER1(String.copyValueOf(word),2);
        else
        {System.out.println(file+"Identifier");
        BOOKKEEPER.BOOKKEEPER1(file,"IDENTIFIER");}
        }
}

//…………ERRORHANDLER WILL FIND THE TYPE OF THE ERROR AND PRINT IT……………

static class ERRORHANDLER{
public static void ERRORHANDLER1(String err,int type)
```

```
{
if(type==1)
        System.out.println("["+err+"]"+"error: This is a country where we speak English.");
else if(type==2)
        System.out.println("["+err+"]"+"error: I'm really rich, part of the beauty of me is I'm very rich.");
else
        System.out.println("Trump does not want to hear.");
}
}
public static void main(String args[]) throws IOException
{
        // ………………………………DISPLAYING THE INPUT FILE............................................

        System.out.println("INPUT PROGRAM");
        BufferedReader br1=new BufferedReader(new FileReader("D:\\spring 18\\CC\\program.txt"));
        String program=null;
        while((program = br1.readLine()) != null)
                System.out.println(program);
        int read;
        BufferedReader br2=new BufferedReader(new FileReader("D:\\spring 18\\CC\\program.txt"));
        System.out.println("TOKEN\t\tTYPE");

        //……….CALLING THE SCANNER FOR EACH SYMBOL IN THE PROGRAM…………

        while((read=br2.read()) !=-1)
        {
                SCANNER.SCANNER1((char)read);
        }
        SCANNER.SCANNER1('$');
        br1.close();
        br2.close();
        System.out.println("----------------------------------------");
        System.out.println("\tSYMBOL TABLE");
        System.out.println("----------------------------------------");
        System.out.println("|TOKEN\t\t|ATTRIBUTE\t|");
        System.out.println("----------------------------------------");
        for(String key: SYMTAB.keySet())
                System.out.println("|"+key + "\t\t|" + SYMTAB.get(key)+"\t|");
}
}
```

OUTPUT:

```
INPUT PROGRAM
Make programming great again
# main body begins
Make x number make  y1 z2zz 1w numbers
make a b Boolean
X is 1000000 y1 is 2000000 z is 123456789
A is fact b  is lie
As long as, fact or lie ;
:
Tell x y1 z2zz say"continue"
If , x plus (y ) times 2000000 more z? ; : tell a b say "stop" ! else : make
c boolean !
C is not not not fact and x less z ? or lie
Tell a b c x y z
Say"done" # say done
!
America is great

TOKEN       TYPE
Make        Keyword
programming Keyword
great       Keyword
again       Keyword
Make        Keyword
X           Identifier
number      Keyword
make        Keyword
y1          Identifier
z2zz        Identifier
[1w]error: I'm really rich, part of the beauty of me is I'm very rich.
Numbers     Identifier
make        Keyword
a           Identifier
b           Identifier
Boolean     Keyword
X           Identifier
is          Keyword
[1000000]error: I'm really rich, part of the beauty of me is I'm very rich.
y1          Identifier
is          Keyword
2000000     Constant
Z           Identifier
is          Keyword
123456789   Constant
A           Identifier
is          Keyword
fact        Keyword
b           Identifier
is          Keyword
```

```
lie         Keyword
As          Keyword
long        Keyword
as          Keyword
,           Special symbol
fact        Keyword
or          Keyword
lie         Keyword
;           Special symbol
:           Special symbol
Tell        Keyword
X           Identifier
y1          Identifier
z2zz        Identifier
say         Keyword
continue    String
If          Keyword
,           Special symbol
X           Identifier
plus        Keyword
(           Special symbol
y           Identifier
)           Special symbol
times       Keyword
2000000     Constant
more        Keyword
z           Identifier
?           Special symbol
;           Special symbol
:           Special symbol
tell        Keyword
a           Identifier
b           Identifier
say         Keyword
stop        String
!           Special symbol
else        Keyword
:           Special symbol
make        Keyword
c           Identifier
boolean     Keyword
!           Special symbol
C           Identifier
is          Keyword
not         Keyword
not         Keyword
not         Keyword
fact        Keyword
and         Keyword
x           Identifier
```

```
less        Keyword
z           Identifier
?           Special symbol
or          Keyword
lie         Keyword
Tell        Keyword
a           Identifier
b           Identifier
c           Identifier
x           Identifier
y           Identifier
z           Identifier
Say         Keyword
done        String
!           Special symbol
America     Keyword
is          Keyword
great       Keyword

-----------------------------
      SYMBOL TABLE
-----------------------------
|TOKEN            |ATTRIBUTE
-----------------------------
|z2zz             |IDENTIFIER|
|a                |IDENTIFIER|
|b                |IDENTIFIER|
|c                |IDENTIFIER|
|123456789        |CONSTANT  |
|numbers          |IDENTIFIER|
|2000000          |CONSTANT  |
|done             |STRING    |
|stop             |STRING    |
|continue         |STRING    |
|x                |IDENTIFIER|
|y1               |IDENTIFIER|
|y                |IDENTIFIER|
|z                |IDENTIFIER|
------------------------------
```