**Control Flow Graph**

**Method 1:**

BufferedReader open\_character\_stream(String fname) {

1=> BufferedReader br = null;

2=> if (fname == null) {

3=> br = new BufferedReader(new InputStreamReader(System.in));

4=> } else {

5=> try {

6=> FileReader fr = new FileReader(fname);

7=> br = new BufferedReader(fr);

8=> } catch (FileNotFoundException e) {

9=> System.out.print("The file " + fname +" doesn't exists\n");

10=> e.printStackTrace();

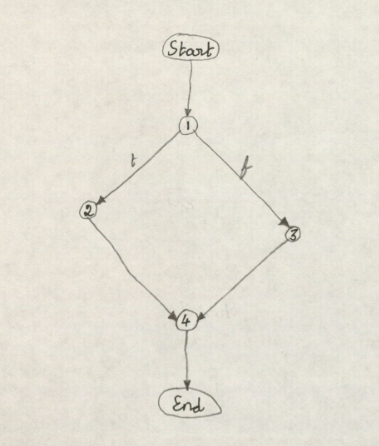
11=> }

12=> }

13=> return br;

}

|  |  |  |  |
| --- | --- | --- | --- |
| **Block** | **Lines** | **Entry** | **Exit** |
| 1 | 1, 2 | 1 | 2 |
| 2 | 3 | 3 | 3 |
| 3 | 4, 5, 6, 7 | 4 | 7 |
| 4 | 11, 12, 13 | 11 | 13 |



**Method 2:**

int get\_char(BufferedReader br){

1=> int ch = 0;

2=> try {

3=> br.mark(4);

4=> ch= br.read();

5=> } catch (IOException e) {

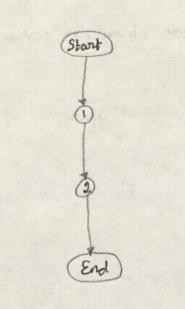
6=> e.printStackTrace();

7=> }

8=> return ch;

}

|  |  |  |  |
| --- | --- | --- | --- |
| **Block** | **Lines** | **Entry** | **Exit** |
| 1 | 1, 2, 3, 4 | 1 | 4 |
| 2 | 7, 8 | 7 | 8 |



**Method 3:**

char unget\_char (int ch,BufferedReader br) {

1=> try {

2=> br.reset();

3=> } catch (IOException e) {

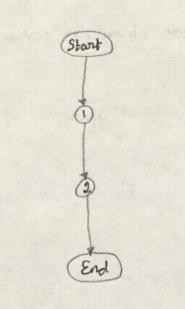
4=> e.printStackTrace();

5=> }

6=> return 0;

}

|  |  |  |  |
| --- | --- | --- | --- |
| **Block** | **Lines** | **Entry** | **Exit** |
| 1 | 1, 2 | 1 | 2 |
| 2 | 5, 6 | 5 | 6 |



**Method 4:**

BufferedReader open\_token\_stream(String fname)

1=> {

2=> BufferedReader br;

3=> if(fname.equals(""))

4=> br=open\_character\_stream(null);

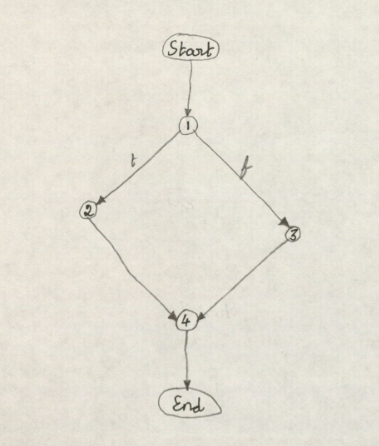
5=> else

6=> br=open\_character\_stream(fname);

7=> return br;

}

|  |  |  |  |
| --- | --- | --- | --- |
| **Block** | **Lines** | **Entry** | **Exit** |
| 1 | 1, 2, 3 | 1 | 3 |
| 2 | 4 | 4 | 4 |
| 3 | 5, 6 | 5 | 6 |
| 4 | 7 | 7 | 7 |



**Method 5:**

String get\_token(BufferedReader br)

1=> {

2=> int i=0,j;

3=> int id=0;

4=> int res = 0;

5=> char ch = '\0';

6=> StringBuilder sb = new StringBuilder();

7=> try {

8=> res = get\_char(br);

9=> if (res == -1) {

10=> return null;

11=> }

12=> ch = (char)res;

13=> while(ch==' '||ch=='\n' || ch == '\r')

14=> {

15=> res = get\_char(br);

16=> ch = (char)res;

17=> }

18=> if(res == -1)

19=> return null;

20=> sb.append(ch);

21=> if(is\_spec\_symbol(ch)==true)

22=> return sb.toString();

23=> if(ch =='"')

24=> id=1;

25=> if(ch ==59)

26=> id=0;

27=> res = get\_char(br);

28=> if (res == -1) {

29=> unget\_char(ch,br);

30=> return sb.toString();

31=> }

32=> ch = (char)res;

33=> while (is\_token\_end(id,res) == false)

34=> {

35=> sb.append(ch);

36=> br.mark(4);

37=> res = get\_char(br);

38=> if (res == -1) {

39=> break;

40=> }

41=> ch = (char)res;

42=> }

43=> if(res == -1)

44=> { unget\_char(ch,br);

45=> return sb.toString();

46=> }

47=> if(is\_spec\_symbol(ch)==true)

48=> { unget\_char(ch,br);

49=> return sb.toString();

50=> }

51=> if(id==1)

52=> {

53=> sb.append(ch);

54=> return sb.toString();

55=> }

56=> if(id==0 && ch==59)

57=> { unget\_char(ch,br);

58=> return sb.toString();

59=> }

60=> } catch (IOException e) {

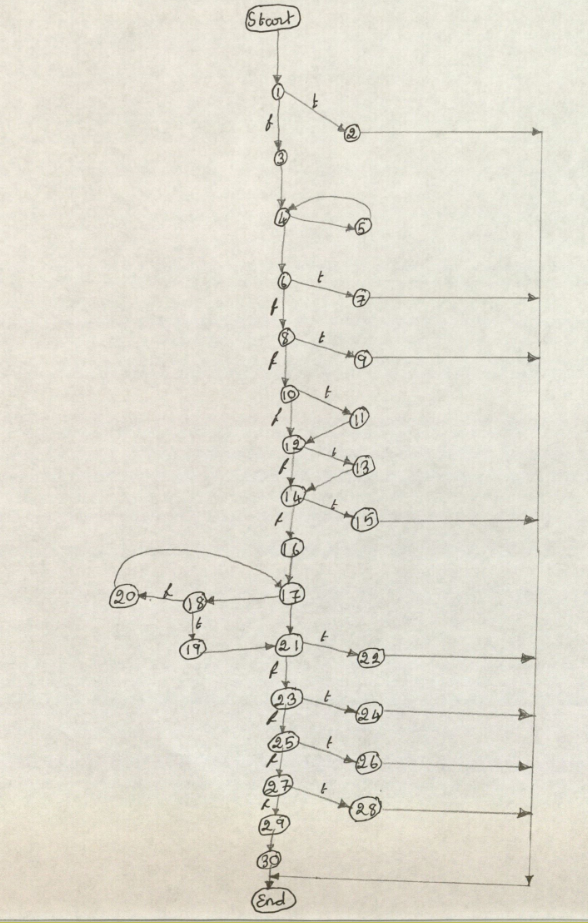
61=> e.printStackTrace();

62=> }

63=> return sb.toString();

}

|  |  |  |  |
| --- | --- | --- | --- |
| **Block** | **Lines** | **Entry** | **Exit** |
| 1 | 1, 2, 3, 4, 5, 6, 7, 8, 9 | 1 | 8 |
| 2 | 10 | 10 | 10 |
| 3 | 11, 12 | 11 | 12 |
| 4 | 13 | 13 | 13 |
| 5 | 14, 15, 16, 17 | 14 | 17 |
| 6 | 18 | 18 | 18 |
| 7 | 19 | 19 | 19 |
| 8 | 20, 21 | 20 | 21 |
| 9 | 22 | 22 | 22 |
| 10 | 23 | 23 | 23 |
| 11 | 24 | 24 | 24 |
| 12 | 25 | 25 | 25 |
| 13 | 26 | 26 | 26 |
| 14 | 27, 28 | 27 | 28 |
| 15 | 29, 30 | 29 | 30 |
| 16 | 31, 32 | 31 | 32 |
| 17 | 33 | 33 | 33 |
| 18 | 34, 35, 36, 37, 38 | 34 | 38 |
| 19 | 39 | 39 | 39 |
| 20 | 40, 41, 42 | 40 | 42 |
| 21 | 43 | 43 | 43 |
| 22 | 44, 45 | 44 | 45 |
| 23 | 46, 47 | 46 | 47 |
| 24 | 48, 49 | 48 | 49 |
| 25 | 50, 51 | 50 | 51 |
| 26 | 52, 53, 54 | 52 | 54 |
| 27 | 55, 56 | 55 | 56 |
| 28 | 57, 58 | 57 | 58 |
| 29 | 59 | 59 | 59 |
| 30 | 62, 63 | 62 | 63 |



**Method 6:**

static boolean is\_token\_end(int str\_com\_id, int res)

1=> {

2=> if(res==-1)

3=> return(true);

4=> char ch = (char)res;

5=> if(str\_com\_id==1)

6=> { if(ch=='"' | ch=='\n' || ch == '\r')

7=> return true;

8=> else

9=> return false;

10=> }

11=> if(str\_com\_id==2) /\* is comment token \*/

12=> { if(ch=='\n' || ch == '\r' || ch==' ') /\* for comment until meet end of line \*/

13=> return true;

14=> else

15=> return false;

16=> }

17=> if(is\_spec\_symbol(ch)==true)

18=> return true;

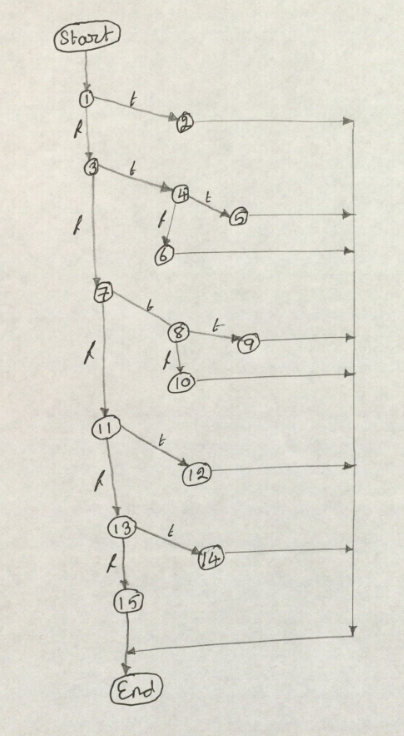
19=> if(ch ==' ' || ch=='\n'|| ch=='\r' || ch==59)

20=> return true;

21=> return false;

}

|  |  |  |  |
| --- | --- | --- | --- |
| **Block** | **Lines** | **Entry** | **Exit** |
| 1 | 1, 2 | 1 | 2 |
| 2 | 3 | 3 | 3 |
| 3 | 4, 5 | 4 | 5 |
| 4 | 6 | 6 | 6 |
| 5 | 7 | 7 | 7 |
| 6 | 8, 9 | 8 | 9 |
| 7 | 10, 11 | 10 | 11 |
| 8 | 12 | 12 | 12 |
| 9 | 13 | 13 | 13 |
| 10 | 14, 15 | 14 | 15 |
| 11 | 16, 17 | 16 | 17 |
| 12 | 18 | 18 | 18 |
| 13 | 19 | 19 | 19 |
| 14 | 20 | 20 | 20 |
| 15 | 21 | 21 | 21 |



**Method 7:**

static int token\_type(String tok)

1=> {

2=> if(is\_keyword(tok))

3=> return(keyword);

4=> if(is\_spec\_symbol(tok.charAt(0)))

5=> return(spec\_symbol);

6=> if(is\_identifier(tok))

7=> return(identifier);

8=> if(is\_num\_constant(tok))

9=> return(num\_constant);

10=> if(is\_str\_constant(tok))

11=> return(str\_constant);

12=> if(is\_char\_constant(tok))

13=> return(char\_constant);

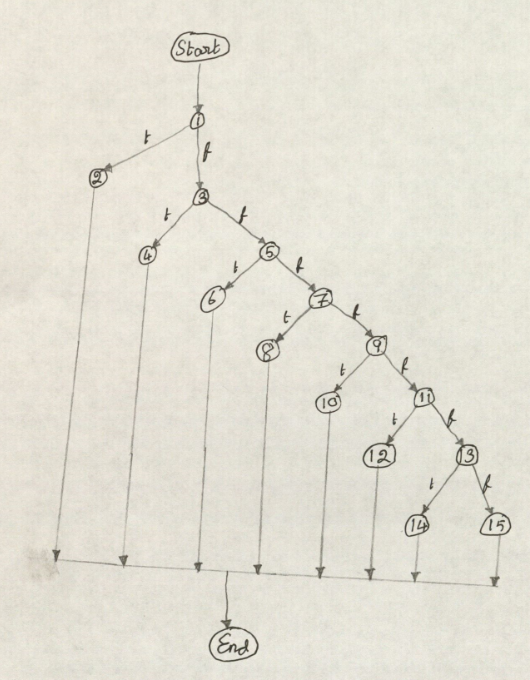
14=> if(is\_comment(tok))

15=> return(comment);

16=> return(error);

}

|  |  |  |  |
| --- | --- | --- | --- |
| **Block** | **Lines** | **Entry** | **Exit** |
| 1 | 1, 2 | 1 | 2 |
| 2 | 3 | 3 | 3 |
| 3 | 4 | 4 | 4 |
| 4 | 5 | 5 | 5 |
| 5 | 6 | 6 | 6 |
| 6 | 7 | 7 | 7 |
| 7 | 8 | 8 | 8 |
| 8 | 9 | 9 | 9 |
| 9 | 10 | 10 | 10 |
| 10 | 11 | 11 | 11 |
| 11 | 12 | 12 | 12 |
| 12 | 13 | 13 | 13 |
| 13 | 14 | 14 | 14 |
| 14 | 15 | 15 | 15 |
| 15 | 16 | 16 | 16 |



**Method 8:**

void print\_token(String tok)

1=> { int type;

2=> type=token\_type(tok);

3=> if(type==error)

4=> {

5=> System.out.print("error,\"" + tok + "\".\n");

6=> }

7=> if(type==keyword)

8=> {

9=> System.out.print("keyword,\"" + tok + "\".\n");

10=> }

11=> if(type==spec\_symbol)

12=> print\_spec\_symbol(tok);

13=> if(type==identifier)

14=> {

15=> System.out.print("identifier,\"" + tok + "\".\n");

16=> }

17=> if(type==num\_constant)

18=> {

19=> System.out.print("numeric," + tok + ".\n");

20=> }

21=> if(type==str\_constant)

22=> {

23=> System.out.print("string," + tok + ".\n");

24=> }

25=> if(type==char\_constant)

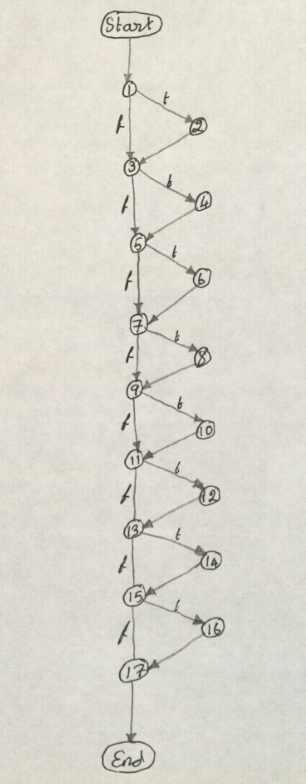
26=> {

27=> System.out.print("character,\"" + tok.charAt(1) + "\".\n");

28=> }

}

|  |  |  |  |
| --- | --- | --- | --- |
| **Block** | **Lines** | **Entry** | **Exit** |
| 1 | 1, 2, 3 | 1 | 3 |
| 2 | 4, 5 | 4 | 5 |
| 3 | 6, 7 | 6 | 7 |
| 4 | 8, 9 | 8 | 9 |
| 5 | 10, 11 | 10 | 11 |
| 6 | 12 | 12 | 12 |
| 7 | 10, 11 | 10 | 11 |
| 8 | 12 | 12 | 12 |
| 9 | 13 | 13 | 13 |
| 10 | 14, 15 | 14 | 15 |
| 11 | 16, 17 | 16 | 17 |
| 12 | 18, 19 | 18 | 19 |
| 13 | 20, 21 | 20 | 21 |
| 14 | 22, 23 | 22 | 23 |
| 15 | 24, 25 | 24 | 25 |
| 16 | 26, 27 | 26 | 27 |
| 17 | 28 | 28 | 28 |



**Method 9:**

static boolean is\_comment(String ident)

1=> {

2=> if( ident.charAt(0) ==59 )

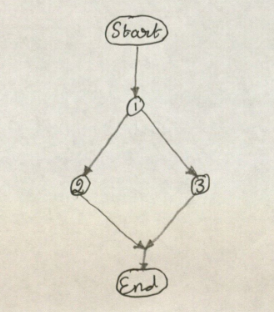
3=> return true;

4=> else

5=> return false;

}

|  |  |  |  |
| --- | --- | --- | --- |
| **Block** | **Lines** | **Entry** | **Exit** |
| 1 | 1, 2 | 1 | 2 |
| 2 | 3 | 3 | 3 |
| 3 | 4, 5 | 4 | 5 |



**Method 10:**

static boolean is\_keyword(String str)

1=> {

2=> if (str.equals("and") || str.equals("or") || str.equals("if") ||

str.equals("xor")||str.equals("lambda")||str.equals("=>"))

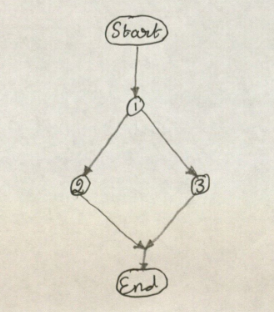
3=> return true;

4=> else

5=> return false;

}

|  |  |  |  |
| --- | --- | --- | --- |
| **Block** | **Lines** | **Entry** | **Exit** |
| 1 | 1, 2 | 1 | 2 |
| 2 | 3 | 3 | 3 |
| 3 | 4, 5 | 4 | 5 |



**Method 11:**

static boolean is\_char\_constant(String str)

1=> {

2=> if (str.length() >= 2 && str.charAt(0)=='#' && Character.isLetter(str.charAt(1)))

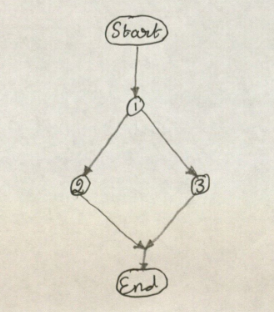
3=> return true;

4=> else

5=> return false;

}

|  |  |  |  |
| --- | --- | --- | --- |
| **Block** | **Lines** | **Entry** | **Exit** |
| 1 | 1, 2 | 1 | 2 |
| 2 | 3 | 3 | 3 |
| 3 | 4, 5 | 4 | 5 |



**Method 12:**

static boolean is\_num\_constant(String str)

1=> {

2=> int i=1;

3=> if ( Character.isDigit(str.charAt(0)))

4=> {

5=> while ( i < str.length() && str.charAt(i) != '\0' )

6=> {

7=> if(Character.isDigit(str.charAt(i+1)))

8=> i++;

9=> else

10=> return false;

11=> }

12=> return true;

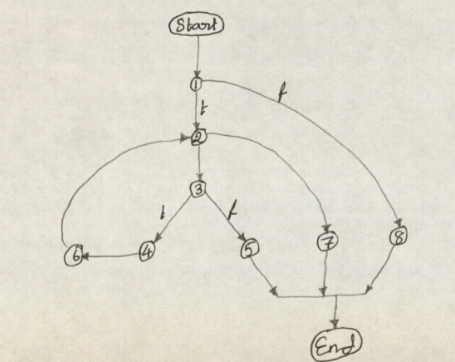
13=> }

14=> else

15=> return false;

}

|  |  |  |  |
| --- | --- | --- | --- |
| **Block** | **Lines** | **Entry** | **Exit** |
| 1 | 1, 2, 3 | 1 | 3 |
| 2 | 4, 5 | 4 | 5 |
| 3 | 6, 7 | 6 | 7 |
| 4 | 8 | 8 | 8 |
| 5 | 9, 10 | 9 | 10 |
| 6 | 11 | 11 | 11 |
| 7 | 12 | 12 | 12 |
| 8 | 13, 14, 15 | 13 | 15 |



**Method 13:**

static boolean is\_str\_constant(String str)

1=> {

2=> int i=1;

3=> if ( str.charAt(0) =='"')

4=> { while (i < str.length() && str.charAt(0)!='\0')

5=> { if(str.charAt(i)=='"')

6=> return true;

7=> else

8=> i++;

9=> }

10=> return true;

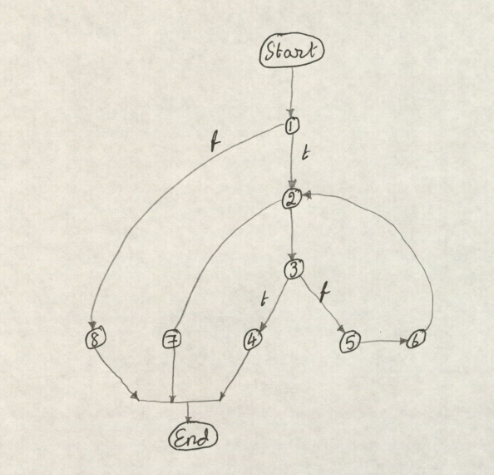
11=> }

12=> else

13=> return false;

}

|  |  |  |  |
| --- | --- | --- | --- |
| **Block** | **Lines** | **Entry** | **Exit** |
| 1 | 1, 2, 3 | 1 | 3 |
| 2 | 4 | 4 | 4 |
| 3 | 5 | 5 | 5 |
| 4 | 6 | 6 | 6 |
| 5 | 7, 8 | 7 | 8 |
| 6 | 9 | 9 | 9 |
| 7 | 10 | 10 | 10 |
| 8 | 11, 12, 13 | 11 | 13 |



**Method 14:**

static boolean is\_identifier(String str)

1=> {

2=> int i=1;

3=> if ( Character.isLetter(str.charAt(0)) )

4=> {

5=> while(i < str.length() && str.charAt(i) !='\0' )

6=> {

7=> if(Character.isLetter(str.charAt(i)) || Character.isDigit(str.charAt(i)))

8=> i++;

9=> else

10=> return false;

11=> }

12=> return true;

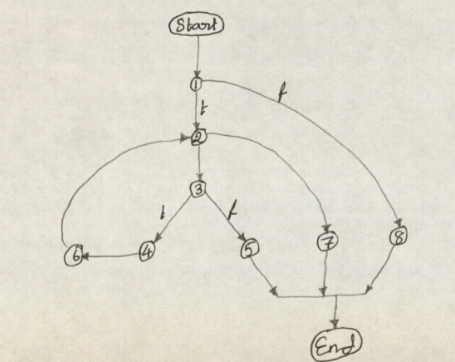
13=> }

14=> else

15=> return false;

}

|  |  |  |  |
| --- | --- | --- | --- |
| **Block** | **Lines** | **Entry** | **Exit** |
| 1 | 1, 2, 3 | 1 | 3 |
| 2 | 4, 5 | 4 | 5 |
| 3 | 6, 7 | 6 | 7 |
| 4 | 8 | 8 | 8 |
| 5 | 9, 10 | 9 | 10 |
| 6 | 11 | 11 | 11 |
| 7 | 12, 13 | 12 | 13 |
| 8 | 14, 15 | 13 | 15 |



**Method 15:**

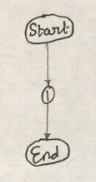
static void unget\_error(BufferedReader br)

1=> {

2=> System.out.print("It can not get charcter\n");

}

|  |  |  |  |
| --- | --- | --- | --- |
| **Block** | **Lines** | **Entry** | **Exit** |
| 1 | 1, 2 | 1 | 2 |



**Method 16:**

static void print\_spec\_symbol(String str)

1=> {

2=> if (str.equals("("))

3=> {

4=> System.out.print("lparen.\n");

5=> return;

6=> }

7=> if (str.equals(")"))

8=> {

9=> System.out.print("rparen.\n");

10=> return;

11=> }

12=> if (str.equals("["))

13=> {

14=> System.out.print("lsquare.\n");

15=> return;

16=> }

17=> if (str.equals("]"))

18=> {

19=> System.out.print("rsquare.\n");

20=> return;

21=> }

22=> if (str.equals("'"))

23=> {

24=> System.out.print("quote.\n");

25=> return;

26=> }

27=> if (str.equals("`"))

28=> {

29=> System.out.print("bquote.\n");

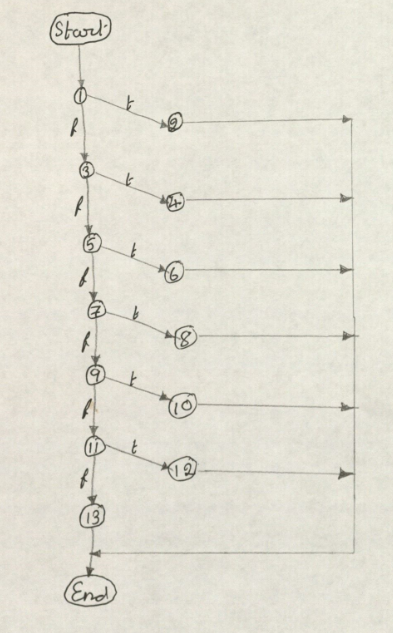
30=> return;

31=> }

32=> System.out.print("comma.\n");

}

|  |  |  |  |
| --- | --- | --- | --- |
| **Block** | **Lines** | **Entry** | **Exit** |
| 1 | 1, 2 | 1 | 2 |
| 2 | 3, 4, 5 | 3 | 5 |
| 3 | 6, 7 | 6 | 7 |
| 4 | 8, 9, 10 | 8 | 10 |
| 5 | 11, 12 | 11 | 12 |
| 6 | 13, 14, 15 | 13 | 15 |
| 7 | 16, 17 | 16 | 17 |
| 8 | 18, 19, 20 | 18 | 20 |
| 9 | 21, 22 | 21 | 22 |
| 10 | 23, 24, 25 | 23 | 25 |
| 11 | 26, 27 | 26 | 27 |
| 12 | 28, 29, 30 | 28 | 30 |
| 13 | 31, 32 | 31 | 32 |



**Method 17:**

static boolean is\_spec\_symbol(char c)

1=> {

2=> if (c == '(')

3=> {

4=> return true;

5=> }

6=> if (c == ')')

7=> {

8=> return true;

9=> }

10=> if (c == '[')

11=> {

12=> return true;

13=> }

14=> if (c == ']')

15=> {

16=> return true;

17=> }

18=> if (c == '\'')

19=> {

20=> return true;

21=> }

22=> if (c == '`')

23=> {

24=> return true;

25=> }

26=> if (c == ',')

27=> {

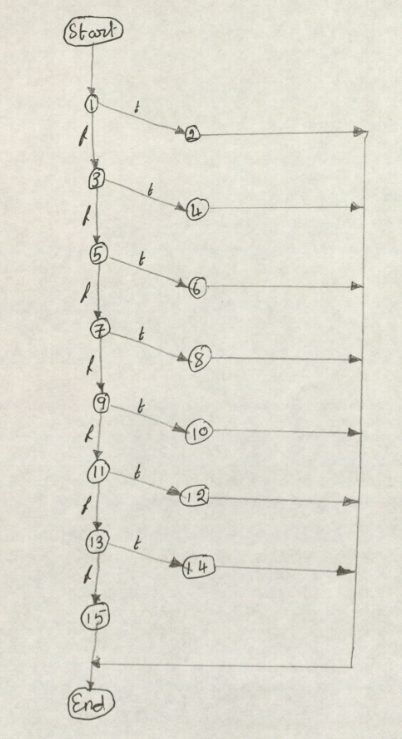
28=> return true;

29=> }

30=> return false;

}

|  |  |  |  |
| --- | --- | --- | --- |
| **Block** | **Lines** | **Entry** | **Exit** |
| 1 | 1, 2 | 1 | 2 |
| 2 | 3, 4 | 3 | 4 |
| 3 | 5, 6 | 5 | 6 |
| 4 | 7, 8 | 7 | 8 |
| 5 | 9, 10 | 9 | 10 |
| 6 | 11, 12 | 11 | 12 |
| 7 | 13, 14 | 13 | 14 |
| 8 | 15, 16 | 15 | 16 |
| 9 | 17, 18 | 17 | 18 |
| 10 | 19, 20 | 19 | 20 |
| 11 | 21, 22 | 21 | 22 |
| 12 | 23, 24 | 23 | 24 |
| 13 | 25, 26 | 25 | 26 |
| 14 | 27, 28 | 27 | 28 |
| 15 | 29, 30 | 29 | 30 |



**Main Method**

public static void main(String[] args) {

1=> String fname = null;

2=> if (args.length == 0) {

3=> fname = new String();

4=> } else if (args.length == 1) {

5=> fname = args[0];

6=> } else {

7=> System.out.print("Error!,please give the token stream\n");

8=> System.exit(0);

9=> }

10=> Printtokens2 t = new Printtokens2();

11=> BufferedReader br = t.open\_token\_stream(fname);

12=> String tok = t.get\_token(br);

13=> while (tok != null) {

14=> t.print\_token(tok);

15=> tok = t.get\_token(br);

16=> }

17=> System.exit(0);

}

|  |  |  |  |
| --- | --- | --- | --- |
| **Block** | **Lines** | **Entry** | **Exit** |
| 1 | 1, 2 | 1 | 2 |
| 2 | 3 | 3 | 3 |
| 3 | 4 | 4 | 4 |
| 4 | 5 | 5 | 5 |
| 5 | 6, 7, 8 | 6 | 8 |
| 6 | 9, 10, 11, 12 | 9 | 12 |
| 7 | 13 | 13 | 13 |
| 8 | 14, 15 | 14 | 15 |
| 9 | 16, 17 | 16 | 17 |

