

Sample success scenario

```
Token: PROGRAM, Value: program
Token: VAR, Value: var
Token: ident, Value: SMALLER
Token: AS, Value: as
Token: INT, Value: int
Token: SC, Value: ;
Token: VAR, Value: var
Token: ident, Value: BIGGER
Token: AS, Value: as
Token: INT, Value: int
Token: SC, Value: ;
Token: VAR, Value: var
Token: ident, Value: TEMP
Token: AS, Value: as
Token: INT, Value: int
Token: SC, Value: ;
Token: BEGIN, Value: begin
Token: ident, Value: BIGGER
Token: ASGN, Value: :=
Token: READINT, Value: readInt
Token: SC, Value: ;
Token: ident, Value: SMALLER
Token: ASGN, Value: :=
Token: READINT, Value: readInt
Token: SC, Value: ;
Token: IF, Value: if
Token: ident, Value: SMALLER
Token: OP4, Value: >
Token: ident, Value: BIGGER
Token: THEN, Value: then
Token: ident, Value: TEMP
Token: ASGN, Value: :=
Token: ident, Value: SMALLER
Token: SC, Value: ;
Token: ident, Value: TEMP1
Token: ASGN, Value: :=
Token: num, Value: 2147483648
Token: SC, Value: ;
Token: ident, Value: SMALLER
Token: ASGN, Value: :=
Token: ident, Value: BIGGER
Token: SC, Value: ;
Token: ident, Value: BIGGER
Token: ASGN, Value: :=
```

```
Token: ident, Value: TEMP
Token: SC, Value: ;
Token: END, Value: end
Token: SC, Value: ;
Token: WHILE, Value: while
Token: ident, Value: SMALLER
Token: OP4, Value: >
Token: num, Value: 0
Token: DO, Value: do
Token: ident, Value: BIGGER
Token: ASGN, Value: :=
Token: ident, Value: BIGGER
Token: OP3, Value: -
Token: ident, Value: SMALLER
Token: SC, Value: ;
Token: IF, Value: if
Token: ident, Value: SMALLER
Token: OP4, Value: >
Token: ident, Value: BIGGER
Token: THEN, Value: then
Token: ident, Value: TEMP
Token: ASGN, Value: :=
Token: ident, Value: SMALLER
Token: SC, Value: ;
Token: ident, Value: SMALLER
Token: ASGN, Value: :=
Token: ident, Value: BIGGER
Token: SC, Value: ;
Token: ident, Value: BIGGER
Token: ASGN, Value: :=
Token: ident, Value: TEMP
Token: SC, Value: ;
Token: END, Value: end
Token: SC, Value: ;
Token: END, Value: end
Token: SC, Value: ;
Token: WRITEINT, Value: writeInt
Token: ident, Value: BIGGER
Token: SC, Value: ;
Token: END, Value: end
```

Lexical analysis successful. No errors found.

SUCCESS

Press enter to end program...

{cslinux1:~/cs4386}

Example failure scenario

```
Token: PROGRAM, Value: program
Token: VAR, Value: var
Token: ident, Value: SMALLER
Token: AS, Value: as
Invalid token: $
Token: INT, Value: int
Token: SC, Value: ;
Token: VAR, Value: var
Token: ident, Value: BIGGER
Token: AS, Value: as
Token: INT, Value: int
Token: SC, Value: ;
Token: VAR, Value: var
Token: ident, Value: TEMP
Token: AS, Value: as
Token: INT, Value: int
Token: SC, Value: ;
Token: BEGIN, Value: begin
Token: ident, Value: BIGGER
Invalid token: :
Token: ASGN, Value: :=
Token: READINT, Value: readInt
Token: SC, Value: ;
Token: ident, Value: SMALLER
Token: ASGN, Value: :=
Token: READINT, Value: readInt
Token: SC, Value: ;
Token: IF, Value: if
Token: ident, Value: SMALLER
Token: OP4, Value: >
Token: ident, Value: BIGGER
Token: THEN, Value: then
Token: ident, Value: TEMP
Token: ASGN, Value: :=
Token: ident, Value: SMALLER
Token: SC, Value: ;
Token: ident, Value: TEMP1
Token: ASGN, Value: :=
Token: num, Value: 2147483648
Token: SC, Value: ;
Token: ident, Value: SMALLER
Token: ASGN, Value: :=
Token: ident, Value: BIGGER
Token: SC, Value: ;
```

```
Token: ident, Value: BIGGER
Token: ASGN, Value: :=
Token: ident, Value: TEMP
Token: SC, Value: ;
Token: END, Value: end
Token: SC, Value: ;
Token: WHILE, Value: while
Token: ident, Value: SMALLER
Token: OP4, Value: >
Token: num, Value: 0
Token: DO, Value: do
Token: ident, Value: BIGGER
Token: ASGN, Value: :=
Token: ident, Value: BIGGER
Token: OP3, Value: -
Token: ident, Value: SMALLER
Token: SC, Value: ;
Token: IF, Value: if
Token: ident, Value: SMALLER
Token: OP4, Value: >
Token: ident, Value: BIGGER
Token: THEN, Value: then
Token: ident, Value: TEMP
Token: ASGN, Value: :=
Token: ident, Value: SMALLER
Token: SC, Value: ;
Token: ident, Value: SMALLER
Token: ASGN, Value: :=
Token: ident, Value: BIGGER
Token: SC, Value: ;
Token: ident, Value: BIGGER
Token: ASGN, Value: :=
Token: ident, Value: TEMP
Token: SC, Value: ;
Token: END, Value: end
Token: SC, Value: ;
Token: END, Value: end
Token: SC, Value: ;
Token: WRITEINT, Value: writeInt
Token: ident, Value: BIGGER
Token: SC, Value: ;
Token: END, Value: end
```

There were 2 errors.

Press enter to end program...

cslinux1:~/cs4386