Grammar:

P->prog DL SL end	DL->D DL|#
D->TY VL ;	TY->int|float
VL->id VL'	SL->S SL|#
S->ES|IS|WS|IOS	ES->id := E ;
IS->if BE then SL IS'	WS->while BE do SL end
IOS->print PE|scan id	PE->E|str
BE->AE BE'	AE->NE AE'
NE->not NE|{ BE }|RE	RE->E RE'
E->T E'	T->F T'
F->(E)|id|ic|fc	BE'->or AE BE'|#
AE'->and NE AE'|#	E'->+ T E'|- T E'|#
T'->* F T'|/ F T'|#	VL'->VL|#
IS'->end|else SL end	RE'->= E|< E|> E

Parsing Table:

It can be found in the file 'Parsing_table.txt' in Grammar2 folder. It is saved as a file as it cannot be visualized easily on command prompt due to size.

Input 1:

```
prog
int i;
int count;
sum := 0;
count := 5;
scan count
print sum
if count = 3
then sum:=2;
else
if count<4
then sum:=2.3;
```

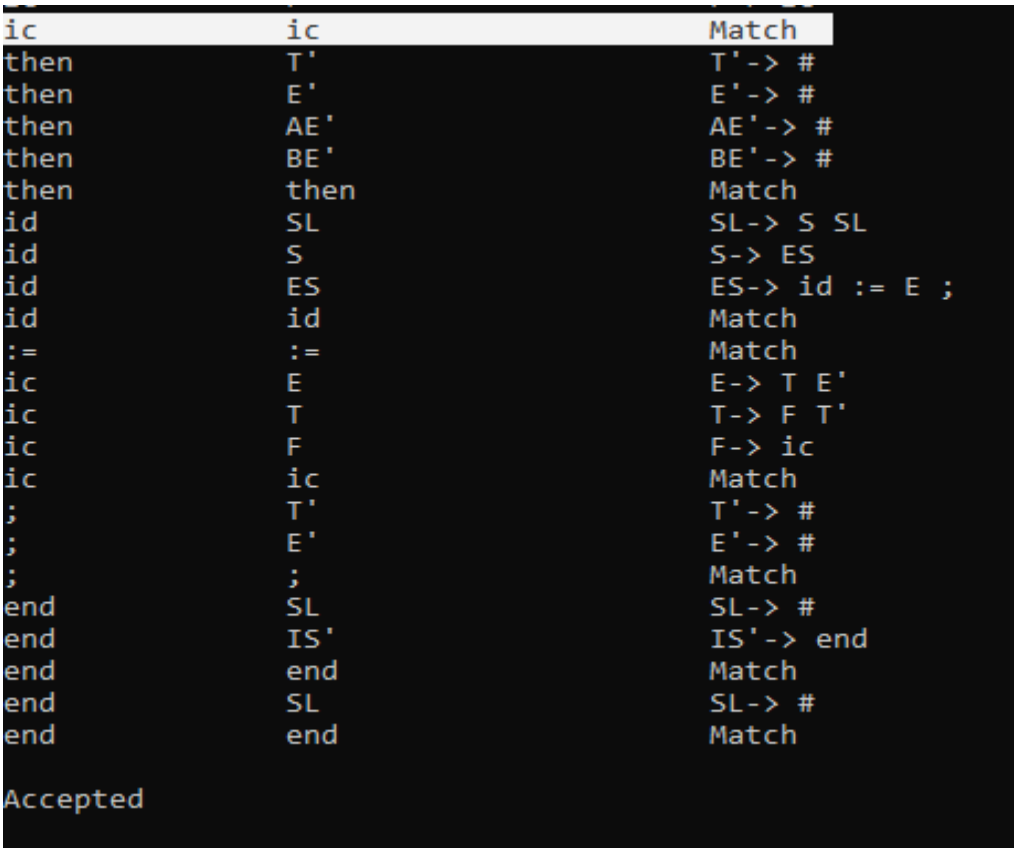
```
end
while count>2.3
do sum := sum – 1;
end
end
if count=3
then sum :=2;
end
end
```

Output 1:

```
Parsing:
Current Input   Current stack top       Next Action
prog            P                       P-> prog DL SL end
prog            prog                    Match
int             DL                      DL-> D DL
int             D                       D-> TY VL ;
int             TY                      TY-> int
int             int                     Match
id              VL                      VL-> id VL'
id              id                      Match
;               VL'                     VL'-> #
;               ;                       Match
int             DL                      DL-> D DL
int             D                       D-> TY VL ;
int             TY                      TY-> int
int             int                     Match
id              VL                      VL-> id VL'
id              id                      Match
;               VL'                     VL'-> #
;               ;                       Match
id              DL                      DL-> #
id              SL                      SL-> S SL
id              S                       S-> ES
id              ES                      ES-> id := E ;
id              id                      Match
:=              :=                      Match
ic              E                       E-> T E'
ic              T                       T-> F T'
ic              F                       F-> ic
ic              ic                      Match
;               T'                      T'-> #
;               E'                      E'-> #
;               ;                       Match
id              SL                      SL-> S SL
id              S                       S-> ES
id              ES                      ES-> id := E ;
id              id                      Match
:=              :=                      Match
ic              E                       E-> T E'
ic              T                       T-> F T'
ic              F                       F-> ic
ic              ic                      Match
```

```
ic                  ic                  Match
;                   T'                  T'-> #
;                   E'                  E'-> #
;                   ;                   Match
scan                SL                  SL-> S SL
scan                S                   S-> IOS
scan                IOS                 IOS-> scan id
scan                scan                Match
id                  id                  Match
print               SL                  SL-> S SL
print               S                   S-> IOS
print               IOS                 IOS-> print PE
print               print               Match
id                  PE                  PE-> E
id                  E                   E-> T E'
id                  T                   T-> F T'
id                  F                   F-> id
id                  id                  Match
if                  T'                  T'-> #
if                  E'                  E'-> #
if                  SL                  SL-> S SL
if                  S                   S-> IS
if                  IS                  IS-> if BE then SL IS'
if                  if                  Match
id                  BE                  BE-> AE BE'
id                  AE                  AE-> NE AE'
id                  NE                  NE-> RE
id                  RE                  RE-> E RE'
id                  E                   E-> T E'
id                  T                   T-> F T'
id                  F                   F-> id
id                  id                  Match
-                   T'                  T'-> #
-                   E'                  E'-> #
-                   RE'                 RE'-> - E
-                   -                   Match
ic                  E                   E-> T E'
ic                  T                   T-> F T'
ic                  F                   F-> ic
ic                  ic                  Match
then                T'                  T'-> #
then                E'                  E'-> #
then                E'                  E'-> #
then                AE'                 AE'-> #
then                BE'                 BE'-> #
then                then                Match
id                  SL                  SL-> S SL
id                  S                   S-> ES
id                  ES                  ES-> id := E ;
id                  id                  Match
:=                  :=                  Match
ic                  E                   E-> T E'
ic                  T                   T-> F T'
ic                  F                   F-> ic
ic                  ic                  Match
;                   T'                  T'-> #
;                   E'                  E'-> #
;                   ;                   Match
else                SL                  SL-> #
else                IS'                 IS'-> else SL end
else                else                Match
if                  SL                  SL-> S SL
if                  S                   S-> IS
if                  IS                  IS-> if BE then SL IS'
if                  if                  Match
id                  BE                  BE-> AE BE'
id                  AE                  AE-> NE AE'
id                  NE                  NE-> RE
id                  RE                  RE-> E RE'
id                  E                   E-> T E'
id                  T                   T-> F T'
id                  F                   F-> id
id                  id                  Match
<                   T'                  T'-> #
<                   E'                  E'-> #
<                   RE'                 RE'-> < E
<                   <                   Match
ic                  E                   E-> T E'
ic                  T                   T-> F T'
ic                  F                   F-> ic
ic                  ic                  Match
then                T'                  T'-> #
then                E'                  E'-> #
then                AE'                 AE'-> #
then                BE'                 BE'-> #
```

```
then                BE'                 BE'-> #
then                then                Match
id                  SL                  SL-> S SL
id                  S                   S-> ES
id                  ES                  ES-> id := E ;
id                  id                  Match
:=                  :=                  Match
fc                  E                   E-> T E'
fc                  T                   T-> F T'
fc                  F                   F-> fc
fc                  fc                  Match
;                   T'                  T'-> #
;                   E'                  E'-> #
;                   ;                   Match
end                 SL                  SL-> #
end                 IS'                 IS'-> end
end                 end                 Match
while               SL                  SL-> S SL
while               S                   S-> WS
while               WS                  WS-> while BE do SL end
while               while               Match
id                  BE                  BE-> AE BE'
id                  AE                  AE-> NE AE'
id                  NE                  NE-> RE
id                  RE                  RE-> E RE'
id                  E                   E-> T E'
id                  T                   T-> F T'
id                  F                   F-> id
id                  id                  Match
>                   T'                  T'-> #
>                   E'                  E'-> #
>                   RE'                 RE'-> > E
>                   >                   Match
fc                  E                   E-> T E'
fc                  T                   T-> F T'
fc                  F                   F-> fc
fc                  fc                  Match
do                  T'                  T'-> #
do                  E'                  E'-> #
do                  AE'                 AE'-> #
do                  BE'                 BE'-> #
do                  do                  Match
do                  do                  Match
id                  SL                  SL-> S SL
id                  S                   S-> ES
id                  ES                  ES-> id := E ;
id                  id                  Match
:=                  :=                  Match
id                  E                   E-> T E'
id                  T                   T-> F T'
id                  F                   F-> id
id                  id                  Match
-                   T'                  T'-> #
-                   E'                  E'-> - T E'
-                   -                   Match
ic                  T                   T-> F T'
ic                  F                   F-> ic
ic                  ic                  Match
;                   T'                  T'-> #
;                   E'                  E'-> #
;                   ;                   Match
end                 SL                  SL-> #
end                 end                 Match
end                 SL                  SL-> #
end                 end                 Match
if                  SL                  SL-> S SL
if                  S                   S-> IS
if                  IS                  IS-> if BE then SL IS'
if                  if                  Match
id                  BE                  BE-> AE BE'
id                  AE                  AE-> NE AE'
id                  NE                  NE-> RE
id                  RE                  RE-> E RE'
id                  E                   E-> T E'
id                  T                   T-> F T'
id                  F                   F-> id
id                  id                  Match
=                   T'                  T'-> #
=                   E'                  E'-> #
=                   RE'                 RE'-> = E
=                   =                   Match
ic                  E                   E-> T E'
ic                  T                   T-> F T'
ic                  F                   F-> ic
ic                  ic                  Match
```



ic	ic	Match
then	T'	T'-> #
then	E'	E'-> #
then	AE'	AE'-> #
then	BE'	BE'-> #
then	then	Match
id	SL	SL-> S SL
id	S	S-> ES
id	ES	ES-> id := E ;
id	id	Match
:=	:=	Match
ic	E	E-> T E'
ic	T	T-> F T'
ic	F	F-> ic
ic	ic	Match
;	T'	T'-> #
;	E'	E'-> #
;	;	Match
end	SL	SL-> #
end	IS'	IS'-> end
end	end	Match
end	SL	SL-> #
end	end	Match

Accepted

Input 2:

Prog
if x = 5
then y = 2.3 ;
else
y := 2 ;
end
end

(Note = operator is used instead of := in line 3 i.e 8th input word.)

Output 2:

```
Parsing:
Current Input   Current stack top       Next Action
prog            P                       P-> prog DL SL end
prog            prog                    Match
if              DL                      DL-> #
if              SL                      SL-> S SL
if              S                       S-> IS
if              IS                      IS-> if BE then SL IS'
if              if                      Match
id              BE                      BE-> AE BE'
id              AE                      AE-> NE AE'
id              NE                      NE-> RE
id              RE                      RE-> E RE'
id              E                       E-> T E'
id              T                       T-> F T'
id              F                       F-> id
id              id                      Match
=               T'                      T'-> #
=               E'                      E'-> #
=               RE'                     RE'-> = E
=               =                       Match
ic              E                       E-> T E'
ic              T                       T-> F T'
ic              F                       F-> ic
ic              ic                      Match
then            T'                      T'-> #
then            E'                      E'-> #
then            AE'                     AE'-> #
then            BE'                     BE'-> #
then            then                    Match
id              SL                      SL-> S SL
id              S                       S-> ES
id              ES                      ES-> id := E ;
id              id                      Match
Not Accepted
Some syntax error has occured near word number 8
```

Input 3:

Prog
if x = 5
then y := 2.3 :
else
y := 2 ;
end
end

(Note undefined symbol : is used instead of ; in line 3 i.e 10th input word.)

Output 3:

```
Parsing:
Current Input   Current stack top       Next Action
prog            P                       P-> prog DL SL end
prog            prog                    Match
if              DL                      DL-> #
if              SL                      SL-> S SL
if              S                       S-> IS
if              IS                      IS-> if BE then SL IS'
if              if                      Match
id              BE                      BE-> AE BE'
id              AE                      AE-> NE AE'
id              NE                      NE-> RE
id              RE                      RE-> E RE'
id              E                       E-> T E'
id              T                       T-> F T'
id              F                       F-> id
id              id                      Match
=               T'                      T'-> #
=               E'                      E'-> #
=               RE'                     RE'-> = E
=               =                       Match
ic              E                       E-> T E'
ic              T                       T-> F T'
ic              F                       F-> ic
ic              ic                      Match
then            T'                      T'-> #
then            E'                      E'-> #
then            AE'                     AE'-> #
then            BE'                     BE'-> #
then            then                    Match
id              SL                      SL-> S SL
id              S                       S-> ES
id              ES                      ES-> id := E ;
id              id                      Match
:=              :=                      Match
fc              E                       E-> T E'
fc              T                       T-> F T'
fc              F                       F-> fc
fc              fc                      Match
Not Accepted

Encountered lexical error
```