Pure Python LL1 Complier. Run python3 custom.txt <code.txt>

P3

Connor Gillis

```
<Block>
                                                                                                # <Var Decl>
                                                           <Assignment Statement>
 j = 4
                                                                                                # # [TYPE_INT]
                                                            # [j]
 string c
                                                                                                # # [q]
                                                            # [4]
                                                                                                # <Var Decl>
 c = false
                                                                                                  # [TYPE INT]
                                                            <Var Decl>
 string e
                                                            # [TYPE_STR]
 e = "connor"
                                                                                                  <Assignment Statement>
                                                            # [c]
                                                                                                  # [b]
 boolean g
                                                            <Assignment Statement>
                                                                                                  # [1]
 q = false
                                                                                                  <Assignment Statement>
                                                            # [c]
 int m
                                                            # [False]
 int a
                                                            <Var Decl>
                                                                                                # <Var Decl>
 a = x
                                                            # [TYPE_STR]
                                                                                                # # [TYPE_INT]
 int b
                                                            # [e]
                                                                                                  <Assignment Statement>
                                                            <Assignment Statement>
                                                                                                  # [w]
   int q
                                                           # [e]
                                                                                                # # [1]
                                                                                                  <Block>
   int q
                                                            # ["connor"]
                                                                                                   # <Print Statement>
   b = 1
                                                            <Var Decl>
                                                            # [TYPE_BOOL]
   d = 3
                          SC => AST
                                                                                                  # <Print Statement>
                                                            # [g]
                                                                                                  # # [m]
   int w
                                                                                                   # <Var Decl>
                                                            <Assignment Statement>
   w = 1
                                                                                                  # # [TYPE_INT]
                                                            # [g]
                                                            # [False]
                                                                                                   # <Assignment Statement>
     print(p)
                                                            <Var Decl>
                                                                                                   # # [1]
     print(m)
                                                            # [TYPE_INT]
     int l
                                                                                                   # <Print Statement>
                                                            # [m]
     l = 5
                                                                                                     # [1]
                                                            <Var Decl>
                                                                                                   # <Var Decl>
     print(l)
                                                            # [TYPE_INT]
                                                                                                   # # [TYPE_INT]
     int r
                                                            # [a]
                                                                                                   # <Assignment Statement>
     r = 2
                                                            <Assignment Statement>
                                                                                                     # [r]
     if (w == r){
                                                            # [a]
                                                                                                     # [2]
                                                            # [x]
                                                                                                     <If Statement>
                                                                                                     # <isEq>
                                                           <Var Decl>
                                                                                                     # # [w]
                                                            # [TYPE_INT]
                                                                                                   # # # [r]
                                                              [b]
}$
                                                                                                  # # [Block]
```

<Block>

```
j = 4
string c
c = false
string e
e = "connor"
boolean q
q = false
int m
int a
a = x
int b
  int q
  int q
  b = 1
  d = 3
  int w
 w = 1
    print(p)
    print(m)
    int l
    l = 5
    print(l)
    int r
    r = 2
    if (w == r){
```

```
ERROR SA - TYPEINT [q] redeclared in same scope.
ERROR SA - Will not add to symbol table
```

```
j = 4
          INFO SA - Attempt Assignment: [c] <= [False]</pre>
string c
c = false
          ERROR SA - Type Mismatch: TYPESTR [c] cannot be assigned int boolean: False
string e
e = "connor"
boolean q
g = false
                          Type mismatch on assign, don't set init to true
int m
int a
a = x
int b
                                              Scope Name: 1
 int q
 int a
                                              - Scope Symbols
 b = 1
 d = 3
                                              *** ID: c
 int w
 w = 1
                                              *** TYPE: TYPESTR
  print(p)
  print(m)
                                              *** VAL: typemismatch
  int 1
  1 = 5
                                              *** INIT: False
  print(l)
  int r
                                              *** USED: False
  r = 2
  if (w == r){
```

*** USEDCOUNT: 0

```
j = 4
string c
c = false
           INFO SA - Attempt Assignment: [e] <= [connor]</pre>
string e
          INFO SA - Type Checking... TYPESTR with connor
e = "connor"
          INFO SA - Successful Assignment: ID e found in scope1 assigned val connor
boolean q
g = false
int m
int a
a = x
int b
                                       ID: e
 int q
                                       TYPE: TYPESTR
 int q
 b = 1
                                       VAL: connor
 d = 3
 int w
 w = 1
                                       INIT: True
  print(p)
                                       USED: False
  print(m)
  int l
                                *** USEDCOUNT: 0
  1 = 5
  print(l)
  int r
```

No type mismatch on assign, set init to true

r = 2

if (w == r){

```
j = 4
string c
c = false
string e
e = "connor"
boolean q
q = false
int m
int a
a = x
int b
 int q
            INFO SA - Attempt Assignment: [b] <= [1]</pre>
 int q
 b = 1
            INFO SA - Type Checking... TYPEINT with 1
 d = 3
            INFO SA - Successful Assignment: ID b found in scope2 assigned val 1
 int w
 w = 1
  print(p)
                                                                 *** ID: b
  print(m)
                                                                  *** TYPE: TYPEINT
  int l
  1 = 5
                                                                  *** VAL: 1
  print(l)
  int r
                                                                  *** INIT: True
  r = 2
                                                                  *** USED: False
  if (w == r){
                                                                  *** USEDCOUNT: 0
```

```
j = 4
string c
c = false
                                               Check the children on print statement (error)
string e
e = "connor"
boolean g
g = false
int m
int a
a = x
int b
 int q
 int q
 b = 1
 d = 3
 int w
 w = 1
   print(p)
             WARNING SA - Uninitialized Variable. (Prev Scope) Unable to PRINT( m )
   print(m)
   int l
   l = 5
   print(l)
   int r
   r = 2
   if (w == r){
```

```
j = 4
string c
c = false
                                           Check the children on print statement (good)
string e
e = "connor"
boolean g
g = false
int m
int a
a = x
int b
 int q
 int q
 b = 1
 d = 3
 int w
 w = 1
   print(p)
   print(m)
   int l
          INFO SA - Var Declared and Initialized. Able to PRINT( 1 )
   l = 5
   print(l)
   int r
   r = 2
   if (w == r){
```

```
j = 4
string c
c = false
string e
e = "connor"
boolean q
g = false
int m
                                Check type compatibility for isEq and isNotEq
int a
                                children
a = x
int b
 int q
 int q
 b = 1
 d = 3
 int w
 w = 1
   print(p)
   print(m)
   int l
   l = 5
               INFO SA - Compare Check .. Var Declared and Initialized (Prev. Scope): w
   print(l)
               TYPEINT
   int r
               INFO SA - Compare Check .. Var Declared and Initialized: r
   r = 2
   if (w == r){
               TYPEINT
               INFO SA - Compare Check PASS TYPEINT and TYPEINT
```