Assignment-4

(Assignment on Language Translation)

Submission Deadline: 16/11/2018

1. Write a translator to convert PASCAL declarations to corresponding C style declarations, as specified below

   Syntax of PASCAL declarations:

   Variables are declared in a *var* section. In Pascal, every variable has a data type already known at compile-time. A variable is declared by a tuple, separated by a colon. Pascal declarations must be terminated by a semicolon.

   Example

Pascal	C

`var`
`  a,b,c : integer;`
`  x : integer[0..15,0..17,0..21];`
`  foo : record`
`  begin;`
`  x,y,z : integer;`
`  p,r : char;`
`  end;`
	`int a,b,c;`
	`int x[15][17][21];`
	`struct`
	`{`
	`  int x,y,z;;`
	`  char p,r;`
	`} foo;`

`  a,b,c : integer`	`// syntax error`

   The input file would not contain the var declaration, just one or more variable or record declarations. The data types will be integer or character.

   The translator must parse each declaration and identify if they are valid or not.
   If the Pascal declaration is valid the corresponding C style declaration must be produced
   Otherwise, a commented error message is to be printed.

2. Write a translator to convert a for in C to a while loop, as specified below

   The input for loop will not have skipped sections and will contain assignments or a nested for loop (same constraints)

```
for( <id>=<const val>; <id><relation><const val>; <id><inc/dec> )
{
  Assignment statement
  for (...)
  {
      ...
  }
}
```

```
while( <id><relation><const val>)
{
  <id>=<const val>;
  Assignment statement
  while (...)
  {
      ...
  }
  <id><inc/dec>
}
```

   The input file will contain one for loop with one or more nested for loops. The nested loops will not contain further nested loops.

   The translator must parse the declaration and identify if they are valid or not.
   If the input is valid the corresponding while loop is produced
   Otherwise, an error message along with the line number is to be printed.