READ ME

1. Enter the Grammar in the LR(1)_Parser.py file to compute its corresponding Parse Table.
2. Once the Parse Table is computed we use that for building our AST(Attribute Syntax Tree) and computing TAC(Three Address Code).
3. AST.py file takes any Arithmetic Expression as input and generates its corresponding Attribute Syntax Tree and also calculates the value of the expression.
4. TAC.py file takes any Arithmetic Expression as input and generates its corresponding Three Address Code and also calculates the value of the expression.
5. DAG.py file uses network for plotting the graphs. Use pip install network, for installing networkx. DAG.py file takes expression as the input and finds its corresponding Direct Acyclic Graph.