CS4100/5100 COMPILER PROJECT
Part 1: Foundations Grade Sheet

Main Program should be a simple, modular, calls to other functions: BuildQuads, BuildSymbolTable, InterpretQuads(Quadtable,SymbolTable,True), InterpretQuads(Quadtable,SymbolTable,False); as specified	15	_____
ADT: QuadTable– indexed list of entries, containing opcode and 3 operand fields; Required Methods: Initialize, NextQuad, AddQuad, GetQuad, SetQuad, GetMnemonic, PrintQuadTable	15	_____
Create and use a Quad Table instance and fill it with the Quad codes for in-class score summation and 10-factorial algorithms	10	_____
ADT: SymbolTable– indexed list of entries, containing name field, kind, data type field, and value fields for int, double, String; Methods: AddSymbol (3 data types), LookupSymbol, GetSymbol, UpdateSymbol, PrintSymbolTable	15	_____
Create and use a Symbol table to accommodate the variables and constants needed to execute the Quad codes correctly	10	_____
ADT: ReserveTable- Methods: Init, Add, LookupName, LookupCode, PrintReserveTable	10	_____
Create and use a Reserved Words lookup table structure for mnemonics in QuadTable	10	_____
Interpreter method- Has parameters: Quads, Symbol, Trace, complete and correctly implements all defined opcodes	10	_____
Good Software Engineering throughout, encapsulated, functions as needed, documented with comments	10	_____
Turn-in:		
All Source in a single .txt file NEATLY collected and presented	10	_____
Run the Quad codes 4 times- (2 algorithms, with and without trace) using the interpreter algorithm given, printing the final answers for the problem		
Runs of each algorithm (sum scores, factorial) with **TRACE ON**, clear, neat, understandable	15	_____
Same runs (sum scores, factorial) with **TRACE OFF**, clear, neat, understandable	15	_____
TOTAL	**145**	_____

NAME: