Q1) Consider the following grammar

E → E+T | T
T → F*T|F
F→ id

a) What is the operation with the highest precedence? (1 grade)
b) What is the associativity of the * operation? (1 grade)
c) Rewrite the grammar to reverse the precedence of operations and associativity of the * operation. (1 grade)
d) Rewrite the original grammar to make it suitable for recursive descent parsing. (1 grade)

e) Rewrite the original grammar to make it suitable for LL(1) parsing. (1 grades)

Please scan your answers and upload the file.

Consider the following inference rule that is used by a type system __(3 grades

$$\frac{\vdash e_1 : T \quad \vdash e_2 : T \quad T \text{ is a primitive type}}{\vdash e_1 == e_2 : \texttt{bool}}$$

Suggest the minimal (smallest) possible modification to the type system so that we can compare integers with floats.

Please scan your answer and upload your answer.

Q3) Consider the following grammar

E → E+T | T
T → F*T |F
F→ id

a) Design an NFA that recognizes all viable prefixes of the grammar. (2 grades)

b) Convert the NFA into a DFA. (3 grades)

Please scan your answers and upload the file.

Q4) For each of the following expressions, replace d1 and d2 by the two rightmost digits in your student number. For example, if your student number is 40345678 then replace d1 with 7 and d2 with 8. Then generate the code for the expressions. In other words, write down the output that would be produced by cgen if it was called to generate the code for the above expression.

a) *d1+d2* (4 grades)

b) if d1=d2 then d1+d2 else d1-d2 (3 grades)

Please scan your answers and upload your answer.