Lecture 20B Worksheet

Question 1

You are writing a tokeniser. The target language contains a “/” of class “symbol” and single line comments that start with “//” that must be ignored. Complete the following part of the next_token function’s switch statement:

```
case ‘/’:
        nextch() ;
```

Question 2

You are writing a parser and the LL(1) grammar contains the following rule and associated code generation rule, leave room to add extra code between lines for question 3.

    whileStatement ::= 'while' '(' expression ')' '{' statements '}'

```
void parseWhileStatement()              // "while" has already been read
{







}
```

Question 3

Add code generation to the parseWhileStatement function based on this code generation rule:

```
whileStatement ::=   label WHILE_EXPRn
                     expression
                     not
                     if-goto WHILE_ENDn
                     statements
                     goto WHILE_EXPRn
```