1. Users of the Tiger programming language have suggested that it could be modified to make a useful scripting language for World-Wide Web browsers. The modified language, called TigerScript, would offer improved support for reading and writing files using URLs to specify the file location, as shown in the following two examples.

```
print (file:/home/stg/tiger.out, "writing to a file\n");
getchar (http://www.dcs.ed.ac.uk/home/cs3/index.html);
```

Only restricted forms of general URLs are allowed in the TigerScript language. These are made up of a *protocol* which is either `file:` or `http:`. In the latter case, this is followed by two forward slashes and a *machine name* which is either *symbolic* (as in `www.dcs.ed.ac.uk`) or *numeric* (as in `129.215.212.98`). Next comes a *path name* made up of alphanumeric strings prefixed by forward slashes. Finally, the URL is finished with an *extension* which has a dot which is followed by an alphanumeric.

(a) One possible implementation technique is for the TigerScript interpreter to treat each URL as a single token, whose semantic value is the string of characters from the URL itself. Show how the definition of TigerScript URLs would be written as the regular-expression based input to a lexical analyser generator. [**Hint:** Use the regular-expression abbreviation mechanism to structure your answer by defining symbolic machine name, numeric machine name, path name and extension. Introduce other definitions as needed.] [*7 marks*]

(b) An alternative to describing TigerScript URLs by regular expressions is to use a context free grammar instead. The resulting grammar could be used as input to a parser generator. Provide such a grammar which builds up its definition from the tokens **file**, **http**, **alphanum**, **num**, **colon**, **slash** and **dot**. Use additional definitions as necessary to structure your answer. [*7 marks*]

(c) i. What difference would the two forms of definition of a URL (via regular expressions or via context free grammars) make in practice to the TigerScript programmer? [*2 marks*]

ii. Two important classes of grammars for compilation are LL(1) and LALR(1). Explain concisely the difference between the implementation techniques used in parsers for these classes of grammars. [*3 marks*]

iii. It has been suggested that it would be advantageous to have an LL(1) grammar for TigerScript. Do you agree? If so, why? [*3 marks*]

iv. Is your grammar for TigerScript URLs LL(1)? If not, explain where it fails to be LL(1) and state which transformations on grammars could be applied to make it LL(1). [*3 marks*]