

## QUESTIONS ON LEXICAL ANALYSIS

a) Write a **regular expression** that recognises the same string as the following C-like code:

```
ch=getchar();
if (ch=='a') {
    ch=getchar();
    while (ch=='a' || ch=='b') {
        ch=getchar();
        do {
            ch=getchar();
        } while (ch=='c');
    }
    if (ch=='d') accept();
} else {
    if (ch=='e') accept();
}
```

(4 marks)

b) Consider the following **regular expression**

$(0|\epsilon)(0|1)^*0$

i) Construct an **NFA** for the regular expression above using **Thomson's construction**.

(4 marks)

ii) Convert the **NFA to a DFA** using the **subset construction** algorithm. Feel free to use a shortened version of the NFA for this conversion, which does not include unimportant  $\epsilon$ -transitions.

(8 marks)

iii) Use the algorithm for state minimisation to construct a **minimised DFA**.

(4 marks)

c) Draw the **DFA** for the following transition table (s1 is the start state; s4 is the end state)

	a	b
s1	s2	s3
s2	s4	s2
s3	s3	s4
s4	-	-

(4 marks)

d) Consider the alphabet  $V=\{0, 1, \dots, 9\}$  and the language L, which **consists of all strings** of V, which represent all integers that are **greater than** 798 (for example, the strings 799, 890, 2345, 777777 belong to the language L, whereas the strings 1, 42, 711, 798 do not). Provide a regular expression that generates all strings of the language L.

(4 marks)