Exam Performance Feedback Form

COMP10042—Part 1

2009/2010

The following only concerns Section A of this exam.

General remarks: 142 students answered questions from this part. The average mark for this part was 10.5 marks out of twenty. 37 students received a mark of 7 or fewer; of these two had 0 marks, the seven had 1 mark, and 4 students had 2 marks—these candidates seem to have done very little to prepare for the exam. 44 students received a mark of 14 or better, 3 of these got 20 out of 20.

Question 1. 111 students attempted this question. The average mark was 5.0 out of 10. 25 students received 2 marks or fewer, 27 students got 8 marks or higher.

Many students provided automata, regular expressions or grammars which only covered some of the words required. An infinite number of words was typically missed out, often:

- b, bb, bbb, ...
- abb, $abbb$, $abbbb$, ...
- cbc, $cbbc$, $cbbbc$, ... or any words in which the two c generated together had other letters between them,
- words containing more than two cs.

a) Most automata missed out at least one of the groups of words mentioned above, many several. Relatively few allowed words that shouldn't have been allowed, that is those containing an odd number of cs or those containing an a that was not immediately followed by b.

   The smallest automaton possible has four states. It is the product of the smallest automaton for the set of all words over $\{a, b, c\}$ that contain an even number of cs and that for the set of all words over the same alphabet where every a is immediately followed by b.

   Alternatively one can think of what needs to be remembered about any word seen so far, namely whether the number of cs is even or odd, and whether the last letter seen is a. The combination of these lead to four states and the same automaton as above.

b) Again, many patterns given here failed to match one or more of the groups given above. Students whose automaton from a) was very complicated and who tried to apply Algorithm 2 did not manage to do so correctly.

   The easiest way of generating the desired regular expression is to realise that 'letters other than c' can come as ab or as b, leading to

   $$(ab|b)^*(c(ab|b)^*c(ab|b)^*)^*.$$

   There are a number of correct alternatives.

c) Some students had difficulties giving any context-free grammar, and some used non-terminal symbols without really understanding what they were doing. Typically the words generated by the given grammar missed some of the groups mentioned above.

   Given an automaton there is a way of generating a context-free grammar for it, and that is the most straightforward answer here if the automaton is correct. Alternatively, it is sufficient for the start symbol S to allow us to construct $ScScS$, abS, bS and the empty word. A number of students submitted answers where $bcbc$ could be generated but $cbcb$ couldn't.

Question 2. This question was attempted by 98 students. The average mark was 5.7 out of ten. 16 students received a mark of 2 or lower, while 25 students got a mark of eight or more.

It was no accident that I asked for a regular expression before wanting a DFA—this is one example where it is easier to read off the expression from the NFA.

a) Most students got something very close to the correct answer, which is $(abb^*aa^*|acc^*aa^*)^*$ or, equivalently, $(a(bb^*|cc^*)aa^*)^*$. A lot of students forgot the outermost star, losing a mark, and a number failed to realize that it is not sufficient to replace bb^* by b^* (or similarly for aa^* or cc^*). A very few wrote $(b|c)(b^*|c^*)$ for the middle part, which allows mixing bs and cs and is therefore incorrect. Also few wrote $(aa)^*a$ for the last part, which only allows an odd number of as in that position, which is also incorrect. Some students did b) first and then tried to apply Algorithm 2 but ended up with expressions so long that they could not finish the calculation.

b) This is asking for Algorithm 1, and all students who started that correctly got at least partial marks. Some students drew another NFA here (some seemed to do so at random) and got no marks. Mistakes were frequently made in carrying out the algorithm, leading to fewer states than required, or missing some transitions. The result needs to be symmetric in b and c, and some people missed that. The correct automaton has 8 states.

Question 3. This question was attempted by 69 students. It had an average mark of 5.3. 9 students received a mark of at most 2, while 9 managed 8 or more.

a) Almost all students correctly found that three of the four words were generated by the grammar and got two marks.

b) Most students recognized that the language defined by the grammar is that of all words over the alphabet $\{a, b\}$ which contain at least one a and got full marks.

c) Many students did this part correctly but a number could not give properly formed parse trees. If they still had discernibly the right idea they got one mark. The shortest word with two parse trees is aa, but any correct word got full marks.

d) There were few right answers here. The fundamental problem with the grammar given is that the a that is required to exist in the word, and which is generated in the first step, can be matched against any a in a given word. One solution is to ensure that it is the leftmost (or rightmost) a that is matched here. A number of solutions offered did not describe the same language, typically not allowing words like bab to be created. A number of solutions given were still ambiguous, and some seemed to be only random alternations carried out on the given rules. No marks were given for those, but one or two marks were available where it was discernible that the student had something of a right idea.