PAPER CODE NO.
COMP309

THE UNIVERSITY
of LIVERPOOL

JANUARY 2004 EXAMINATIONS

Degree of Bachelor of Arts : Year 3
Degree of Bachelor of Science : Year 3

EFFICIENT SEQUENTIAL ALGORITHMS

TIME ALLOWED: Two and a half Hours

INSTRUCTIONS TO CANDIDATES

Credits will be given for the best <u>four</u> answers only. Each question is marked out of 25.

THE UNIVERSITY
of LIVERPOOL

1. Huffman codes.

   (a) Compute the frequency of each character in the following text.

   a a b c a d b b a b a b a c b c b b b a a b a a b a b a c

   (3 marks)

   (b) Compute the Huffman code $\mathcal{C}_1$ associated with the previous sequence. **(7 marks)**

   (c) Compute the frequency of each character in the following text.

   a a c c a d b b a c a c a c b c b b b a a b a a b a b a c b c a c a a b a c a c c b a b a
   b a b c b c a c a c a a c d a

   (3 marks)

   (d) Compute the Huffman code $\mathcal{C}_2$ associated with the sequence in the previous question. **(7 marks)**

   (e) Encode the sequence in question (b) using each of the codes $\mathcal{C}_1$ and $\mathcal{C}_2$ and compute the length of the encoded text in each case. **(5 marks)**

2. (a) Give a pseudo-code description of the brute-force pattern matching algorithm in which the comparison between the pattern P and a portion of the text T is perfomed in a loop in which individual characters are compared. **(6 marks)**

   (b) Run the brute-force pattern matching algorithm obtained as an answer to the previous question on the text $T \equiv$ ABBBBABBBBABBBBCBB and pattern $P \equiv$ ABBBBC. How many comparisons are performed during this process? **(7 marks)**

   (c) Run the Knuth-Morris-Pratt pattern matching algorithm on the same T and P as above. How many comparisons are performed during this process? (Hints: compute the prefix function first; account for comparisons in the function computing the prefix function too). **(12 marks)**

3. Consider the following list of line segments:

segment	starting point	ending point
s_1	(25.5, 27)	(42, 21)
s_2	(28.5, 13.5)	(42, 16.5)
s_3	(24, 24)	(28.5, 24)
s_4	(24, 10.5)	(28.5, 16.5)
s_5	(39, 15)	(45, 30)

   (a) Give a definition of bounding box of a geometric object. **(5 marks)**

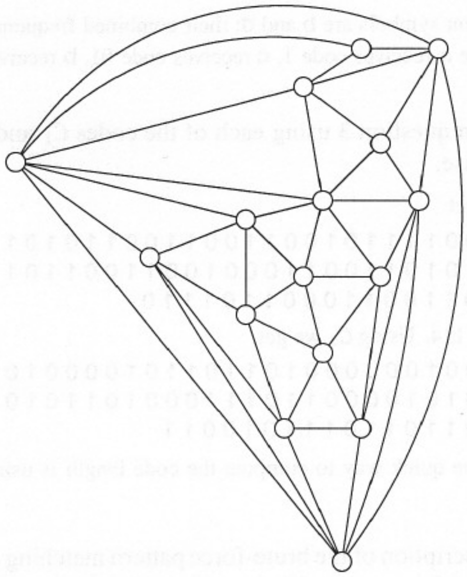
   (b) Determine the event point schedule for the ANY-SEGMENT-INTERSECT algorithm and the set of segments given in the table above. **(11 marks)**

   (c) Simulate ANY-SEGMENT-INTERSECT and find out if there exists any pair of intersecting segments. **(9 marks)**

4. (a) State the definition of absolute approximation algorithm. **(3 marks)**

(b) State and prove the absolute approximation result for vertex-colouring planar graphs. **(7 marks)**

(c) Run the algorithm implicit in the proof of the result above on the following graph:



(15 marks)

(Hints: label the vertices of the graph in the order they are considered by the recursive algorithm; mark the vertices with a label from the set of colours.)

5. (a) Define the MULTIPROCESS SCHEDULING problem. Your definition should express the fact that such problem is an optimisation problem (i.e. all four components in the definition of optimisation problem must be defined in the specific case). **(5 marks)**

(b) Run the LIST SCHEDULING and LPT heuristic on the system formed by two processors and 8 jobs with the following computation times:

$$p_1 = 6, p_2 = 2, p_3 = 5, p_4 = p_5 = 1, p_6 = 6, p_7 = 2, p_8 = 0.5$$

(17 marks)

(c) Find an optimal solution for the instance above. **(3 marks)**

COMP309: Marking Scheme and Model Answers

1. Huffman codes.

   (a) Compute the frequency of each character in the following text.

   a a b c a d b b a b a b a c b c b b b a a b a a b a b a c

   $f(a) = 12/29$, $f(b) = 12/29$, $f(c) = 4/29$, $f(d) = 1/29$.

   (b) Compute the Huffman code $\mathcal{C}_1$ associated with the previous sequence.

   $C(a) = 0$, $C(b) = 10$, $C(c) = 110$, $C(d) = 111$.

   (c) Compute the frequency of each character in the following text.

   a a c c a d b b a c a c a c b c b b b a a b a a b a b a c b c a c a a b a c a c c b a b a
   b a b c b c a c a c a a c d a

   Easy. $f(\mathsf{a}) = 5/12$, $f(\mathsf{c}) = 17/60$, $f(\mathsf{b}) = 16/60$, $f(\mathsf{d}) = 1/30$.

   (d) Compute the Huffman code $\mathcal{C}_2$ associated with the sequence in question 3.

   The two least frequent symbols are b and d; their combined frequency is larger that $f(\mathsf{c})$ (but smaller than $f(\mathsf{a})$. Therefore a receives code 1, c receives code 01, b receives code 000, and d receives code 001.

   (e) Encode the sequence in question 3 using each of the codes $\mathcal{C}_1$ and $\mathcal{C}_2$ and compute the length of the encoded text in each case.

   Easy. Using $\mathcal{C}_1$ we get

   0 0 1 1 0 1 1 0 0 1 1 1 1 0 1 0 0 1 1 0 0 1 1 0 0 1 1 0 1 0 1 1 0 1 0 1 0 1 0 0 0 1 0 0 0
   1 0 0 1 0 0 1 1 0 1 0 1 1 0 0 1 1 0 0 0 1 0 0 1 1 0 0 1 1 0 1 1 0 1 0 0 1 0 0 1 0 0 1 0 1
   1 0 1 0 1 1 0 0 1 1 0 0 1 1 0 0 0 1 1 0 1 1 1 0

   with length equal to 114. Using $\mathcal{C}_2$ we get

   1 1 0 1 0 1 1 0 0 1 0 0 0 0 0 0 1 0 1 1 0 1 1 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 1 0 0 0 1
   1 0 0 0 1 0 0 0 1 0 1 0 0 0 0 1 1 0 1 1 1 0 0 0 1 0 1 1 0 1 0 1 0 0 0 1 0 0 0 1 0 0 0 1 0
   0 0 0 1 0 0 0 0 1 1 0 1 1 0 1 1 1 0 1 0 0 1 1

   with length 113. The quick way to compute the code length is using the frequencies of the various symbols.

2. (a) Give a pseudo-code description of the brute-force pattern matching algorithm in which the comparison between the pattern P and a portion of the text T is perfomed in a loop in which individual characters are compared.

   The following pseudo-code gets full marks.

```
BRUTE-MATCHING (T, P)
    n ← length(T)
    m ← length(P)
    for s ← 0 to n − m
        set a boolean variable match to TRUE
        for i ← 1 to m
            if P[i] ≠ T[s + i]
                match ← FALSE
        if match
            print "pattern occurs with shift s"
```

   An informal English description will get 65%.

   (b) Run the brute-force pattern matching algorithm obtained as an answer to the previous question on the text $T \equiv$ ABBBBABBBBABBBBCBB and pattern $P \equiv$ ABBBBC. How many comparisons are performed during this process?

   6 comparisons when the shift is 0, 5 or 10, 1 comparison when the shift is in $\{1, \ldots, 4, 6, \ldots, 9, 11, 12\}$, for a grand total of 28.

(c) Run the Knuth-Morris-Pratt pattern matching algorithm on the same T and P as above. How many comparisons are performed during this process? (Hints: compute the prefix function first; account for comparisons in the function computing the prefix function too).

The correct, 12 mark, answer goes as follows. The prefix function is identically equal to zero. Here's a trace of KMP algorithm on T and P showing, in the last row, the number of comparisons[1] performed during iteration i:

i		1		2		3		4		5		6		7		8		9		10		11		12		13		14		15	
q	0		1		2		3		4		5		0-1		2		3		4		5		0-1		2		3		4		5
T_i		A		B		B		B		B		A		B		B		B		B		A		B		B		B		B	
comp.		1		2		2		2		2		2		2		2		2		2		2		2		2		2		2	

The total number of comparisons is 33 plus the initial 5, which is actually larger than those of the brute-force algorithm.

If they simulate the algorithm well but, perhaps have some typos on the counting, they get 10 marks.

3. Consider the following list of segments:

segment	starting point	ending point
s_1	$(25.5, 27)$	$(42, 21)$
s_2	$(28.5, 13.5)$	$(42, 16.5)$
s_3	$(24, 24)$	$(28.5, 24)$
s_4	$(24, 10.5)$	$(28.5, 16.5)$
s_5	$(39, 15)$	$(45, 30)$

(a) Give a definition of bounding box of a geometric object.

The *bounding box* of a figure is the smallest rectangle that contains the figure. Figure 1 shows a line segment and its enclosing bounding box.

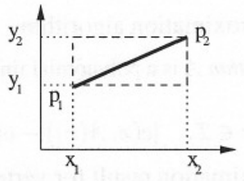


Figure 1: Line segment $\overline{p_1 p_2}$ and its enclosing boundin g box.

(b) Determine the event point schedule for the ANY-SEGMENT-INTERSECT algorithm and the set of segments given in the table above.

The points are ordered as follows:

$$(10.5, 12.7, s_3, S), (16.5, 12, s_2, S), (16.5, 27, s_0, S), (24, 10.5, s_6, S),$$
$$(24, 24, s_5, S), (24, 9, s_3, E), (25.5, 27, s_1, S), (28.5, 13.5, s_4, S),$$
$$(28.5, 16.5, s_6, E), (28.5, 19.5, s_0, E), (28.5, 24, s_5, E), (39, 15, s_7, S),$$
$$(42, 16.5, s_4, E), (42, 21, s_1, E), (42, 27, s_2, E), (45, 30, s_7, E).$$

Let S be the resulting ordered list.

(c) Simulate ANY-SEGMENT-INTERSECT and find out which segments are intersecting.

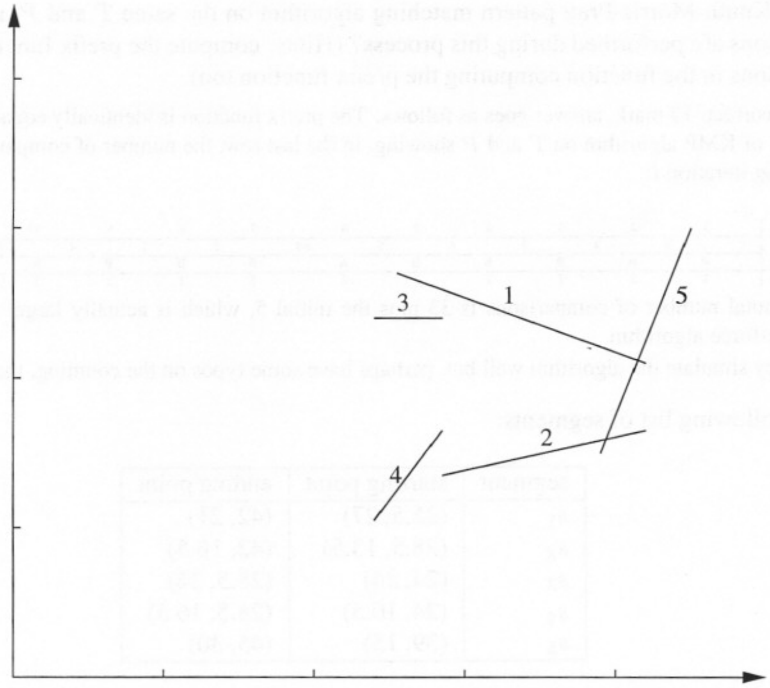
The input can be represented graphically as follows.

The data structure T evolves as shown in the table below

```
(s_4,S)
(s_4,S) (s_3,S)                          check s_3 and s_4 intersect
(s_4,S) (s_3,S) (s_1,S)                  check s_1 and s_3 intersect
(s_2,S) (s_4,S) (s_3,S) (s_1,S)          check s_2 and s_4 intersect
(s_2,S) (s_4,S) (s_3,S) (s_1,S)          check s_2 and s_4 intersect
                                         check s_3 and s_2 intersect
(s_2,S) (s_3,S) (s_1,S)                  (delete s_4)
                                         check s_2 and s_1 intersect

TRUE!
```

[1]We are not counting the comparison in the while loop unless it is actually performed (i.e. if $q = 0$ the comparison is NOT performed!).



Full marks should be given to a description showing the use of cross-product to find segment intersection and to insert segments in the status data structure. Second classs mark to a(n essentially) correct solution not showing the details mentioned above, perhaps referring to a graphical representation of the input. Smaller marks to solutions containing conceptual errors.

4. (a) State the definition of absolute approximation algorithm.

An *absolute approximation algorithm* A is a polynomial time approximation algorithm for Π such that for some constant $k > 0$,

$$\forall x \in \mathcal{I}, \quad |c(x, A(x)) - \text{opt}(x)| \leq k$$

(b) State and prove the absolute approximation result for vertex-colouring planar graphs.

A statement along the following lines gets 3 marks:
There is an absolute approximation algorithm for the planar graph colouring problem.
The proof (worth 4 more marks) is by INDUCTION. The student gets full marks if (s)he mentions induction, and then uses either the 5-colouring or the (much simpler) 6-colouring algorithm to prove the result. Assuming the 6-colouring algorithm is used, the student should state that any planar graph on at most 6 vertices is obviously 6-colourable and any planar graph has (by Euler formula) a vertex of degree at most 5. The result really follows from these facts.

(c) Run the colouring algorithm implicit in the proof of the result above on the given graph:

Hints: label the vertices of the graph in the order they are considered by the recursive algorithm; mark the vertices with a label from the set of colours $\mathcal{C} = \{G, B, Y, O, W\}$.

5. (a) Define the MULTIPROCESS SCHEDULING problem.

The following definition (from the lecture slides) gets full marks.
The input consists of n jobs. Each job has a corresponding runtime $p_1, \ldots, p_n$, where each p_i is assumed to be rational. The jobs are to be scheduled on m identical machines or processors so as to minimise the finish time. The finish time is defined to be the maximum over all processors of the total run-time of the jobs assigned to that processor.

A second class solution should mention the fact that a number of jobs must be assigned to a bunch of machines to minimise processing time.

(b) Run the LIST SCHEDULING and LPT heuristic on the a system formed by two processor and 8 jobs with the following computation times:

$$p_1 = 6, p_2 = 2, p_3 = 5, p_4 = p_5 = 1, p_6 = 9, p_7 = 2, p_8 = 3$$

Eight point for a correct simulation of the LIST SCHEDULING heuristic and nine point for a correct simulation of the LPT heuristic. The resulting allocations are:

```
List Scheduling:

P1 -------------
       ||| |  |
       145 7  8
P2 ----------------
    |    |         |
    2    3         6

MAXSPAN: 16

LPT: order =>   6 1 3 8 2 7 4 5

P1 --------------
           |  | |
           6  8 7
P2 ---------------
       |     | |||
       1     3 245

MAXSPAN: 15
```

(c) Find an optimal solution for the instance above.

Full mark to an answer along the following lines. The total work that has to be done is 29. So the average work per processor is 14.5 The LPT solution is therefore optimal in this case!
2.3 marks for a correct answer obtained from an example.