LUT University
Computational Engineering (LATE)
Prof. Heikki Kälviäinen

BM40A0102 Foundations of Information Processing

Exam example (Academic Year 2021-2022)

No written material is allowed in the exam. **No electronic devices** are allowed in the exam. The students in the international program must reply in English. The students in the Finnish-speaking program can reply either in English or in Finnish. **Please remember to explain your answers carefully in details.**

1. Concepts (10 p):
   (a) Illustrate by drawing how the merge sort algorithm sorts the following numbers 17, 1, 20, 23, 29, 7, 2, 13 in the increasing order.
   (b) What kind of a computational complexity is too difficult when solving a problem? Give an example.
   (c) Illustrate by drawing what is the difference between a tree structure and a graph structure?
   Show whether a tree can be a substructure of a graph.
   (d) How does the asymmetric encryption method differs from the symmetric encrption method?
   (e) Analyze lexically the following part of a program:
   IF x < 5 THEN x := x + 1 ENDIF

2. Functions (10 p):
   (a) Is the algorithm recursive or iterative? Justify your reply. It is not enough to just tell that the algorithm is iterative since it is not recursive.

```
MODULE factorial1(n) RETURNS n!
  IF n = 0 THEN
    RETURN 1
  ELSE
    RETURN n * factorial1(n - 1)
  ENDIF
ENDMODULE
```

   (b) Is the algorithm recursive or iterative? Justify your reply.

```
MODULE factorial2(n) RETURNS n!
  k:=1
  WHILE n>1 DO
    k:=k*n
    n:=n-1
  ENDWHILE
  RETURN k
ENDMODULE
```

   (c) Show all the calls of the function `factorial1` and the corresponding values returned by the function when it is called as `factorial1(4)`. For example, `factorial1(0)` returns `1` directly, but `factorial1(1)` does not give the value directly, Explain how the values are formed.

See the next page (the tasks to be continued)

3. Data encoding (10 p):

   The numbers represented in *the two's complement* notation are to be added and substracted as follows

0010_2	0010_2	1110_2	1110_2
$+\ 0011_2$	$-\ 0011_2$	$+\ 1101_2$	$-\ 1101_2$

   keeping them in the binary representation during computations.

   (a) Show the range of values represented in the two's complement (bit patterns of length four) needed for computations, i.e., all the values from the largest value to the smallest value.
   Compulsory subtask: no points from this task if this subtask has not been done successfully.

   (b) Calculate the given four computations in the two's complement notation, using the range of values defined.

   (c) Double check the results by changing the orignal numbers to the decimal system and performing the same computations.

4. Data compression using the Huffman coding (10 p):

   Let us assume that there are the following estimated probabilities of occurrence for 5 symbols: 0.40, 0.20, 0.20, 0.15 and 0.05.

   (a) Why does the Huffman coding compress the representation of data? Justify the benefit of the method.
   Compulsory subtask: no points from this task if this subtask has not been done successfully.

   (b) Apply the Huffman coding to generate the corresponding Huffman codes for each symbol. Tell step by step how your coding does work.

   (c) How many bits per symbol are needed on the average using your coding?

   (d) How many bits per symbol are needed on the average if there is no compression (so 5 symbols are represented directly without any coding for compression)? Justify your reply.

   (e) What would be a case where the Huffman coding cannot compress?

5. Logic and reasoning (10 p):

   Prove using the semantic method (the truth table) that c can be concluded from the following premises:

$$a \wedge b \rightarrow c$$
$$b \rightarrow a$$
$$b$$

   (a) How are the premisses used for proving?
   Hint: Show the proposition (statement) to be proved and tell how it is related to the premises. Do not yet present the truth table here.
   Compulsory subtask: no points from this task if this subtask has not been done successfully.

   (b) Present the truth table.

   (c) Explain what and how can be concluded from the truth table.

   (d) Why is the computational complexity of the truth table exponential? Explain how it can been seen in the truth table.

   Please remember to explain your answers carefully in details.