CARDIFF UNIVERSITY
EXAMINATION PAPER

Academic Year:	2018/2019
Examination Period:	Spring
Examination Paper Number:	CMT304
Examination Paper Title:	Programming Paradigms
Duration:	2 hours

Do not turn this page over until instructed to do so by the Senior Invigilator.

Structure of Examination Paper:

There are 4 pages.
There are 3 questions in total.
There are no appendices.

The maximum mark for the examination paper is 60 and the mark obtainable for a question or part of a question is shown in brackets alongside the question.

Students to be provided with:

The following items of stationery are to be provided:
ONE answer book.

Instructions to Students:

Answer **all three** questions.

Students are permitted to introduce to this examination any textbook, any printed or handwritten notes, and other similar materials. These may be annotated, highlighted and bookmarked as desired.
The use of translation dictionaries between English or Welsh and a foreign language is permitted in this examination.
The use of electronic devices is not permitted.

Q1. (a) Describe the characteristics of a problem for which Answer Set Programming is the best paradigm to use for addressing it. [4]

(b) Consider the following Answer Set Program:

```
1 c(1..n).
2 r(1..n).
3 {q(I,J) :- r(I), c(J)}.
4 :- not n {q(I,J)} n.
5 :- q(I, J), q(II, JJ), (I,J) != (II,JJ) , I - J == II - JJ.
6 :- q(I, J), q(I, JJ), J != JJ.
7 :- q(I, J), q(II, JJ), (I,J) != (II,JJ), I + J == II + JJ.
8 :- q(I, J), q(II, J), I != II.
```

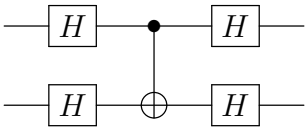
Explain each line. [8]

If this is used in a large program (i.e. with a large `n`), it can be inefficient. Explain why. [4]

Describe in plain text and in code a change to this program to increase its efficiency. [4]

Q2. (a) Explain the difference between the `Functor`, `Applicative` and `Monad` typeclass in Haskell. Provide an instance definition for some data structure for each typeclass as example. [9]

(b) Consider the following quantum circuit:



Show that this circuit can be simplified to a single (standard) gate and explain the operation of this gate. [7]

(c) Is quantum computing part of the functional programming paradigm? Provide in total two arguments, where either argument can be either for or against the proposition. [4]

Q3. (a) For each of the following regular expressions (Perl syntax), describe the language matched by the regular expression in words and give an example for a string matched by the expression:

(i) `:-[()]`

(ii) `[$%@]\w[\w\d]+`

(iii) `[A-Z]\w+.`

[6]

(b) For each of the following problems, state whether it is best solved by script programming, genetic algorithms, or machine programming. Provide a reason for each answer.

(i) Allocate 30000 students to 2000 personal tutors, taking into account staff availability, staff language requirements, and student preferences.

(ii) Compute a cryptographic hash on a very low-powered embedded device.

(iii) Compute statistical measures for file access times on a Unix server.

(iv) Find the optimal strategy for malware to hide itself from detection.

[4]

(c) Assume you have a processor with 5 registers, referred to as R1, R2, R3, R4, and R5, and the following instruction set:

`ADD Rx, Ry`: Add the contents of the registers Rx and Ry and store the result in Rx.
`SUB Rx, Ry`: Subtract the content of Ry from Rx and store the result in Rx.
`DEC Rx`: Decrease the value in register Rx by 1.
`INC Rx`: Increase the value in register Rx by 1.
`LOAD Rx, Z`: Direct load into register Rx using address Z.
`LOAD Rx, @Ry`: Register indirect load into Rx using register Ry
`LOAD Rx, #Z`: Immediate load of Z into register Rx.
`JMP @LABEL`: Unconditional jump to the address of LABEL.
`JZ @LABEL`: Jump to the address of LABEL if the last operation has resulted in a value of zero.

QUESTION CONTINUES ON NEXT PAGE

You are given the following program:

```
1          LOAD R1, #0
2          LOAD R2, 1
3          LOAD R5, #1
4          LOAD R3, 2
5          JZ   @E
6          @L:
7            LOAD R4, @R2
8            LOAD R5, R5
9            JZ @D1
10           JMP @D2
11          @D1:
12            SUB R1, R4
13            INC R5
14            JMP @D3
15          @D2:
16            ADD  R1, R4
17            DEC R5
18          @D3:
19           INC  R2
20           DEC  R3
21           JZ   @E
22           JMP  @L
23         @E:
```

Assume the memory contains the following values (starting from address 1):

```
3 6 1 2 3 4 5 6 7
```

(i) Identify two high-level control-flow constructs in this program. Indicate the type and start and end line for each.

(ii) Which function does the program compute in R1? Describe the general function of the program (you may give a mathematical formula).

(iii) Give the value in R1 after the program terminates for the given memory contents.

[5]

(d) Provide minimal valid example Perl code for the following tasks:

(i) Assign the numerical value 3 to a scalar variable named ‘x’.

(ii) Assign the number of elements in the array ‘a’ to the scalar variable ‘x’.

(iii) Split a string named ‘s’ into individual characters.

(iv) Replace all occurrences of the character ‘a’ in the string named ‘s’ with the character ‘b’.

(v) Assign the string “a” to the key ‘b’ in a hash named ‘test’.

[5]