



MACHAKOS UNIVERSITY

University Examinations for 2020/2021

SCHOOL OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF COMPUTING AND INFORMATION TECHNOLOGY

SECOND YEAR SPECIAL/SUPPLEMENTARY EXAMINATION FOR

BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)

SIT 211: INTRODUCTION TO LOGIC PROGRAMMING

DATE: 26/3/2021

TIME: 8.30-10.30 AM

INSTRUCTIONS: Answer question one and any other two questions

QUESTION ONE (30 MARKS)

- a) i Explain the term predicate as used in Prolog programming (2 marks)
- ii Give the Prolog predicate used to add a fact to the database. Give an example of its use. (2 marks)
- iii Give the Prolog predicate used to remove a fact from database. Give an example of its use. (2 marks)
- b) With an aid of an example in each case explain the following terms as used in Prolog programming
- i. Atom (2 marks)
- ii. Functor (2 marks)
- iii. Clause (2 marks)
- iv. Anonymous variable (2 marks)
- c) The *member* function is used in Prolog to determine if an item occurs in a list. Give a definition of the member function using Prolog (3 marks)
- d) Given that a database contains the following list of children, their names and age.
- aged(jacinter,12).
- aged(rose, 9).
- aged(peter,5).
- aged(daniel,9).

- i. Use the bagof predicate then return a list of children whose age is equal 9. (2 marks)
 - ii. Write a rule that returns sorted list of age without duplicates for all the children in the database (2 marks)
- e) For each of the queries below, state the answer that prolog would give (“yes”, “no”, or a variable binding). (4 marks)
- ?- foo(N,N)=foo(bar,X).
 ?- [a, b] = [X, b].
 ?- X = 25+4.
 ?- X is 25+4.
- f) Prolog will automatically backtrack if it's necessary to satisfy a goal. Automatic backtracking is useful because it relieves the programmer of the burden of programming backtracking explicitly. However, uncontrolled backtracking may cause inefficiency in a program. Discuss the method used to control or prevent backtracking. (5 marks)

QUESTION TWO (20 MARKS)

- a) Discuss declarative and procedural programming paradigms, giving examples of languages in each case. (4 marks)
- b) Suppose we have the following database:
 p(1).
 p(2) :- !.
 p(3).
 Write all of Prolog's answers to the following queries:
- i) ?- p(X). (2 marks)
 - ii) ?- p(X),p(Y). (2 marks)
 - iii) ?- p(X),!,p(Y). (2 marks)
- c) Distinguish between forward and backward chaining. (3 marks)
- d) Classify the following sequences of characters as atoms, variables, or neither? Giving reasons. (3 marks)
- vINCENT
 Footmessage
 Variable87
 _Jules
 '_Jules'
 2names

- e) Giving example in each case explain the built-in predicate *read* and *write* as used in prolog programming. (4 marks)

QUESTION THREE (20 MARKS)

- a) With an aid of a diagram explain the process of evaluating a goal in Prolog. (7 marks)
- b) Outline 3 advantages and 1 disadvantage of the goal CUT as used in Prolog programming. (5 marks)
- c) How many *facts, rules, clauses, and predicates* are there in the following knowledge base? What are the heads of the rules, and what are the goals they contain? (8 marks)

woman(mary).
woman(mia).
man(jules).
person(X) :- man(X); woman(X).
loves(X,Y) :- knows(Y,X).
father(Y,Z) :- man(Y), son(Z,Y).
father(Y,Z) :- man(Y), daughter(Z,Y).

QUESTION FOUR (20 MARKS)

- a) Given:
- Pete is the parent of Mike.
- Pete is the parent of Julie.
- Pete is the parent of Amanda.
- Mary is the parent of Julie.
- Mary is the parent of Amanda.
- Mary is female.
- Julie is female.
- Amanda is female.
- Mike is male.
- Pete is male.
- A male parent is a father.
- A female parent is a mother.
- A male sibling of someone is a brother of that person, unless the person in question is yourself.

A female sibling of someone is a sister of that person, unless the person in question is yourself.

Construct a family Knowledge base in prolog. (7 marks)

- b) Give the syntax for prolog comments (2 marks)
- c) Given a Prolog list, for example $L=[10, 20, 50, 6, 15]$. Write a recursive procedure that gets the sum of the items in such a list. (4 marks)
- d) Explain the following inbuilt predicates giving example of its use. (4 marks)
 - (i) reverse predicate
 - (ii) append predicate
- (c) Use the name predicate to define and test a predicate `remove_final` that removes any final spaces from a string. (3 marks)

QUESTION FIVE (20 MARKS)

- (a) Give the syntax for rules in Prolog and explain its elements (4 marks)
- (b) Given the following Prolog database, where the first argument is the company name, the second argument is the share price, and the third argument is the dividend paid out.
`share(equity,3.5,0.2).`
`share(safaricom,9,2.0).`
`share(kcb,48,1).`
`share(cooperative,18,1).`
`share(famiy,17,0.3).`
`share(toyota,24,3.5).`
 - i. Create a query to extract the names of companies which paid a dividend of two shilling and above. (3 marks)
 - ii. Use the fail predicate to create a rule to list the companies whose dividend is below 1 shilling. (2 marks)
 - iii. Create a query that returns a sorted Prolog list containing all the companies in the database. (3 marks)
 - iv. Prolog associates variables and values using a process known as unification. Outline the rules of unification hence give an example for unification. (3 marks)
- (c) Consider the following Prolog database:
`symptom(colds, sneezing).`

symptom(colds, running_nose).

symptom(flux, headache).

symptom(flux, shivering).

sympom(flux, fever).

sympom(brain_tumour, headache).

(i) Devise a query that will pick the symptoms of colds (2 marks)

(ii) Devise a query that will pick the symptoms of flux. (3 marks)