University of Technology

Department of Computer Sciences

Final Examination 2011-2012

Subject: Intelligent Systems & Applications — *Year: Fourth*
Division: Software & Computer Security — *Time: 3 Hours*
Examiner: Dr. Hasanen S. Abdullah — *Date: 2/ June/2012*

*Answer **Five** Questions Only*

Q1) Suppose that you have the following natural language sentences:

- I feel that you can solve these questions.
- You must think before answering question.

1. Built a suitable Context Free Grammar **(CFG)** and **Case Grammar** for the above sentences. **(3 marks)**
2. Write a prolog program to achieve the **syntax** and **semantic** (thematic) analysis for the given sentences by using the grammars that you built in step**1**. **(7 marks)**

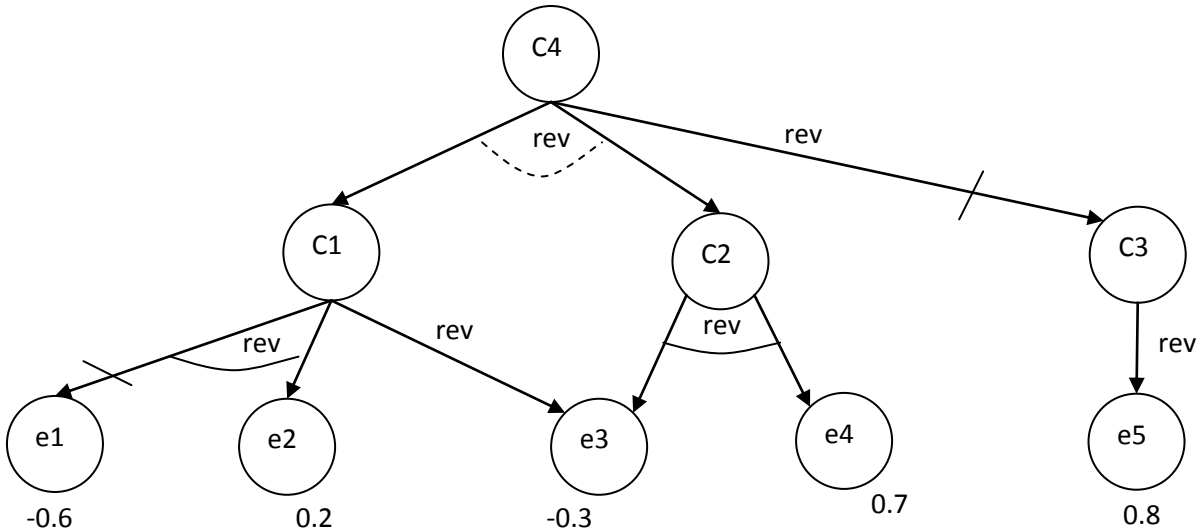
Q2) Design an expert system to classify computers (three types only) by using backward chaining according to the following properties: (processor speed, memory capacity, mother board type, and video card quality). **(10 marks)**

Q3) Consider the inference network below and then give the results of the following items:

1- Find the certainty factor for the node **C4 (the implication value of each node is equal to 0.5)**. **(6 marks)**

2- Show the contents of **WHY** stack when the system asks the user about the certainty value of node **e2**. **(2 marks)**

3- Describe the **HOW** explanation mechanism when the user asks **H C2**. What is the system response? **(2 marks)**



إقلب الصفحة لتكملة الأسئلة

Q4) What is the **main goal** of use the following adaptive parameters? **(Choose <u>five</u> only)**

(Fitness function, Reproduction, Momentum Term, Bias, Mutation, Activation Function, genetic Programming). **<u>(10 marks, 2 marks for each)</u>**

Q5) Choose **<u>one</u>** branch to solve:

A) A BAM artificial neural network is trained using the following input and output patterns:

A1 = (1 1 0 0) B1 = (0 0 1)

A2 = (1 0 1 0) B2 = (0 1 1)

A3 = (1 0 0 1) B3 = (0 1 0) then,

1- Build the weight matrix. **<u>(4 marks)</u>**

2-Apply an input vector A_3 = (1 0 0 1) to test the net to remember A_3 **(In one iteration only)**. **<u>(6 marks)</u>**

B) Consider the following samples that are stored in a Hopfield artificial neural network:

(1 1 0 0) (1 0 1 0) (1 0 0 1), then,

1. Calculate the thresholds for all units of the input pattern. **<u>(5 marks)</u>**
2. Build the weight matrix. **<u>(5 marks)</u>**

Q6) In **two iterations** only, by using genetic algorithm approach and operators (crossover, mutation, and reproduction), try to solve the **8-puzzle** problem with any initial population that you suggest.

<u>(10 marks)</u>

إقلب الصفحة لبداية الأسئلة

Best wishes

The Solution of the Questions

Q1) Solution

1.

Context Free Grammar

S → Np, Vp, Pp

Np → Pro_noun / Np, Vp, Np /det noun / noun, noun.

Vp → verb / helping verb

Pp → preposition, Np

Case Grammar

Sentence 1

agent case

verb case

object case

Sentence 2

agent case

verb case

time case

2.

```
run:- readln(S), str_to_list(S, L), parse(L), case(L, L0).
parse(L):- append(A1, A2, A3, L),
           np(A1),
           vp(A2),
           pp(A3).

np(X):- append(Y1, Y2, _, X),
        det(Y1),
        noun(Y2).
np(X):- append(Y1, Y2, _, X),
        noun(Y1),
        noun(Y2).
np(X):- append(Y1, Y2, Y3, X),
        np(Y1),
        vp(Y2),
        pp(Y3).
np(X):- pro_noun(X).

vp(Z):- append(Y1, Y2, _, X),
        h.verb(Y1),
        verb(Y2).
vp(Z):- verb(Z).
```

```
pp(M):- append(W1, W2, _, M),
        preposition(W1),
        np(W2).

/ * set of Facts */
det(["these"]).
pro_noun(["I"]).        pro_noun(["you"]).
noun(["questions"]).    noun(["question"]).    noun(["answering"]).
verb(["solve"]).        verb(["think"]).    verb(["feel"]).
h.verb(["can"]).        h.verb(["must"]).
preposition(["that"]).        preposition(["before"]).

case(S,S0):- agent(S,S1), backparta(S1, S0).
backparta(S,S0):- verb(S,S1), object(S1,S0).

case(S,S0):- agent(S,S1), backpartb(S1, S0).
backpartb(S,S0):- verb(S,S1), time(S1,S0).
```

Q2) Solution

```
database

db_confirm(symbol, symbol)

db_denied(symbol, symbol)


clauses

guess_computer :- identify(X), write("Your computer is a(n) ",X),!.


identify(type1) :-

                    it_is(video card quality),

                    not(confirm(has, old mother board)),

                    confirm(has, big RAM capacity),

                    it_is(processor version),!.


identify(type2) :-

                    it_is(processor version),

                    confirm(has, normal RAM capacity),
```

```
                    confirm(has, a compatible mother board),
                    it_is(video card quality)!.

identify(type3) :-
                    it_is(video card quality),
                    it_is(processor version),
                    confirm(has, small RAM capacity),
                    confirm(has, old mother board),!.

it_is(video card quality):-
                        confirm(has, high video card quality),!.
it_is(video card quality):-
                        confirm(has, normal video card quality),!.

it_is(processor version):-
                        confirm(has, new version processor),!.
it_is(processor version):-
                        confirm(has, modern version processor),!.

confirm(X,Y):- db_confirm(X,Y),!.
confirm(X,Y):- not(denied(X,Y)),!, check(X,Y).

denied(X,Y):- db-denied(X,Y),!.

Check(X,Y):- write(X, " it ", Y, \ "n"), readln(Reply), remember(X, Y, Reply).

remember(X, Y, yes):- asserta(db_confirm(X, Y)).
remember(X, y, no):- assereta(db_denied(X, Y)), fail.
```

Q3) Solution

1-

$Ct_1(C_1) = 0.2 * 0.5$
$= 0.1$
$Ct_2(C_1) = -0.2 * 0.5$
$= -0.15$
$Ct(C_1) = (0.1 – 0.15) / (1 – 0.1)$
$= -0.05$

$Ct(C_2) = -0.3 * 0.5$
$= -0.15$

$Ct(C_3) = 0.8 * 0.5$
$= 0.4$

$Ct_1(C_4) = -0.05 * 0.5$
$= -0.025$
$Ct_2(C_4) = -0.4 * 0.5$
$= -0.2$
$Ct (C_4) = (-0.025 – 0.2) + (-0.025 * - 0.2)$
$= -0.22$

2-

The contents of why stack about W e_2

db_imp(imp(a, r, C_1, pos, e_1, pos, e_2, o.5))
db_imp(imp(o, r, C_4, neg, C_1, neg, C_2, o.5))

3-

The system response of How facility about H C_2

How describer of the system
S: type h(how) node name, or c to continue
U: h C_2

S: Concluded C_2 with certainty of -0.15 from
e_3 and e_4 → C_2
the rule is reversible
certainty of e_3 is -0.3
certainty of e_4 is 0.7
certainty of the implication is 0.5

Q4) Solution

1.Fitness Function

A function that can give a decision to choose the best individuals in the next population through the genetic algorithm learning.

2.Reproduction

To add new genetic information to the population of the chromosomes by combining strong parents with strong children, the hope is to obtain new more fit children**.**

3.Momentum Term

To increase the learning rate without leading to oscillations is to modify the GDR (Generalized Delta Rule) to include momentum term.

4.Bias

It is a constant value that is added to the input of the neural network so to enhance the learning process.

5.Mutation

A way to introduce new information is by changing the content of some genes. Mutation can be applied to :-

1- Chromosomes selected from the MP.
2- Chromosomes that have already subject to crossover.

6.Activation Function

A function that used in neural network to give the final output of each neuron, it is two types Linear functions and non-Linear functions according to the application.

7.Genetic Programming

It is one of the central goals of computer science namely automatic programming. The goal of automatic programming is to create, in an automated way, a computer program that enables a computer to solve a problem.

Q5) Solution

A) BAM ANN

1-

(A_i) Vectors Transpose * (B_i) = Weight Matrix (W)

1	1	1		-1	-1	1		-3	1	1
1	-1	-1	*	-1	1	1	=	1	-3	1
-1	-1	1		-1	1	-1		1	1	1
								1	1	-3

2-

Forward Phase

A_3 * W

$$(1\ \ -1\ \ -1\ \ 1) * \begin{matrix} -3 & 1 & 1 \\ 1 & -3 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & -3 \end{matrix} = (-4\ \ 4\ \ -4) = B_3(0\ \ 1\ \ 0)$$

Backward Phase

B_3 * W^T

$$(-1\ \ 1\ \ -1) \begin{matrix} -3 & 1 & 1 & 1 \\ 1 & -3 & 1 & 1 \\ 1 & 1 & 1 & -3 \end{matrix} = (3\ \ -5\ \ -1\ \ 3) = A_3(1\ \ 0\ \ 0\ \ 1)$$

The BAM ANN able to remember each vector in each pair.

B) The Hopfield ANN

1- The Thresholds for all units of input pattern

The vectors

(1 1 -1 -1) (1 - 1 1 -1) (1 -1 -1 1)

The matrix

$$\begin{matrix} 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{matrix}$$

The transpose of matrix

$$\begin{matrix} 1 & 1 & 1 \\ 1 & -1 & -1 \\ -1 & 1 & -1 \\ -1 & -1 & 1 \end{matrix} * \begin{matrix} 1 \\ 1 \\ 1 \\ 1 \end{matrix} = \begin{matrix} 3 \\ -1 \\ -1 \\ -1 \end{matrix} = \mathbf{W_0}$$

$$\mathbf{T_j = -W_0 =} \begin{matrix} \mathbf{-3\ (T_1)} \\ \mathbf{1\ \ (T_2)} \\ \mathbf{1\ \ (T_3)} \\ \mathbf{1\ \ (T_4)} \end{matrix}$$

2- The weight matrix

The vectors

(1 1 -1 -1) (1 - 1 1 -1) (1 -1 -1 1)

After applying the Heb rule, the weight matrix is

$$\begin{matrix} 0 & -1 & -1 & -1 \\ -1 & 0 & -1 & -1 \\ -1 & -1 & 0 & -1 \\ -1 & -1 & -1 & 0 \end{matrix}$$

Q6) Solution

Suppose the following initial population of 80Puzzle problem

<u>Initial Population</u>

X1= [2 3 1 0 4 6 5 7 8] **X2= [4 3 2 1 0 5 6 7 8]**

X3= [0 8 7 6 4 3 2 1 5] **X4= [8 7 6 5 3 4 1 0 2]**

<u>The Target Gaol</u>

[1 2 3 4 0 5 6 7 8]

The initial population with fitness value for each chromosome (number of tiles in place according to the target goal chromosome)

By using ordered crossover let K_1= 3 and K_2 = 6

[2 3 1 0 4 6 5 7 8] …..2

[4 3 2 1 0 5 6 7 8] …..4

[0 8 7 6 4 3 2 1 5] …..0

[8 7 6 5 3 4 1 0 2] …..0

Now we apply crossover operator between chromosomes 1 &2 and chromosomes 3&4
2 3 1 0 4 6 5 7 8
4 3 2 1 0 5 6 7 8
Forms new two chromosomes
3 4 61 0 5 7 8 2
2 1 5 0 4 6 7 8 3

0 8 7 6 4 3 2 1 5
8 7 6 5 3 4 1 0 2
Forms new two chromosomes
8 7 6 5 3 4 2 1 0
8 7 5 6 4 3 1 0 2

The new population is
[3 4 61 0 5 7 8 2] …..2
[2 1 5 0 4 6 7 8 3] …..0
[8 7 6 5 3 4 2 1 0] …..0
[8 7 5 6 4 3 1 0 2] …..0

Now we do mutation in te 4th chromosome by swapping the 1st gen with 7th gen

[3 4 61 0 5 7 8 2] …..2
[2 1 5 0 4 6 7 8 3] …..0
[8 7 6 5 3 4 2 1 0] …..0
[1 7 5 6 4 3 8 0 2] …..1

Now we apply the reproduction operator between initial populstion and new population to choose the best chromosomes.

After reproduction we obtain the following chromosomes

[4 3 2 1 0 5 6 7 8] …..4

[2 3 1 0 4 6 5 7 8] …..2

[3 4 61 0 5 7 8 2] …..2

[1 7 5 6 4 3 8 0 2] …..1

Now we apply crossover operator again between chromosomes 1 &2 and chromosomes 3&4
4 3 2 1 0 5 6 7 8
2 3 1 0 4 6 5 7 8
Forms new two chromosomes

2 1 5 0 4 6 7 8 3

3 4 6 1 0 5 7 8 2

3 4 6 1 0 5 7 8 2

1 7 5 6 4 3 8 0 2

Forms new two chromosomes

1 0 5 6 4 3 7 8 2

6 4 3 1 0 5 8 2 7

The new population is

[2 1 5 0 4 6 7 8 3] …..0

[3 4 6 1 0 5 7 8 2] …..2

[1 0 5 6 4 3 7 8 2] …..1

[6 4 3 1 0 5 8 2 7] …..3

We obtain new chromosome with better fitness value (3) which is not found before, and the process is continue until we obtain a chromosome with (9) fitness value.