Name : Run Zeng

Student ID : 20794087

Email : r24zeng@uwaterloo.ca

Question 1

a) There are 4 execution paths totally.

1. [1, 2, 3, 4, 9, 11, 12, 13, 17]
2. [1, 2, 3, 4, 9, 11, 14, 15, 16, 17]
3. [1, 2, 5, 6, 7, 9, 11, 12, 13, 17]
4. [1, 2, 5, 6, 7, 9, 11, 14, 15, 16, 17]

b)

PATH 1

Edge	Symbolic State (PV)	Path Condition (PC)
$1 \rightarrow 2$	$x \longmapsto X_0$; $y \longmapsto Y_0$	true
$2 \rightarrow 3$	$x \longmapsto X_0+1$; $y \longmapsto Y_0$	$X_0 + Y_0 > 10$
$3 \rightarrow 4$	$x \longmapsto X_0+1$; $y \longmapsto Y_0-2$	$X_0 + Y_0 > 10$
$4 \rightarrow 9$	$x \longmapsto X_0+3$; $y \longmapsto Y_0-2$	$X_0 + Y_0 > 10$
$9 \rightarrow 11$	$x \longmapsto X_0+3$; $y \longmapsto Y_0-2$	$X_0 + Y_0 > 10 \land 2X_0 + 2Y_0 > 25$
$11 \rightarrow 12$	$x \longmapsto 3X_0+9$; $y \longmapsto Y_0-2$	$X_0 + Y_0 > 10 \land 2X_0 + 2Y_0 > 25$
$12 \rightarrow 13$	$x \longmapsto 3X_0+9$; $y \longmapsto 2Y_0-4$	$X_0 + Y_0 > 10 \land 2X_0 + 2Y_0 > 25$
$13 \rightarrow 17$	$x \longmapsto 3X_0+9$; $y \longmapsto 2Y_0-4$	$X_0 + Y_0 > 10 \land 2X_0 + 2Y_0 > 25$

PATH 2

Edge	Symbolic State (PV)	Path Condition (PC)
$1 \rightarrow 2$	$x \longmapsto X_0$; $y \longmapsto Y_0$	true
$2 \rightarrow 3$	$x \longmapsto X_0+1$; $y \longmapsto Y_0$	$X_0 + Y_0 > 10$
$3 \rightarrow 4$	$x \longmapsto X_0+1$; $y \longmapsto Y_0-2$	$X_0 + Y_0 > 10$
$4 \rightarrow 9$	$x \longmapsto X_0+3$; $y \longmapsto Y_0-2$	$X_0 + Y_0 > 10$
$9 \rightarrow 11$	$x \longmapsto X_0+3$; $y \longmapsto Y_0-2$	$X_0 + Y_0 > 10 \land 2X_0 + 2Y_0 \leq 25$
$11 \rightarrow 14$	$x \longmapsto X_0+3$; $y \longmapsto Y_0-2$	$X_0 + Y_0 > 10 \land 2X_0 + 2Y_0 \leq 25$
$14 \rightarrow 15$	$x \longmapsto 4X_0+12$; $y \longmapsto Y_0-2$	$X_0 + Y_0 > 10 \land 2X_0 + 2Y_0 \leq 25$
$15 \rightarrow 16$	$x \longmapsto 4X_0+12$; $y \longmapsto 4X_0+3Y_0+6$	$X_0 + Y_0 > 10 \land 2X_0 + 2Y_0 \leq 25$
$16 \rightarrow 17$	$x \longmapsto 4X_0+12$; $y \longmapsto 4X_0+3Y_0+6$	$X_0 + Y_0 > 10 \land 2X_0 + 2Y_0 \leq 25$

PATH 3

Edge	Symbolic State (PV)	Path Condition (PC)
$1 \rightarrow 2$	$x \longmapsto X_0$; $y \longmapsto Y_0$	true
$2 \rightarrow 5$	$x \longmapsto X_0$; $y \longmapsto Y_0$	$X_0 + Y_0 \leq 10$
$5 \rightarrow 6$	$x \longmapsto X_0$; $y \longmapsto Y_0+7$	$X_0 + Y_0 \leq 10$
$6 \rightarrow 7$	$x \longmapsto X_0-3$; $y \longmapsto Y_0+7$	$X_0 + Y_0 \leq 10$
$7 \rightarrow 9$	$x \longmapsto X_0-1$; $y \longmapsto Y_0+7$	$X_0 + Y_0 \leq 10$
$9 \rightarrow 11$	$x \longmapsto X_0-1$; $y \longmapsto Y_0+7$	$X_0 + Y_0 \leq 10 \land 2X_0 + 2Y_0 > 15$
$11 \rightarrow 12$	$x \longmapsto 3X_0+9$; $y \longmapsto Y_0-2$	$X_0 + Y_0 \leq 10 \land 2X_0 + 2Y_0 > 15$
$12 \rightarrow 13$	$x \longmapsto 3X_0+9$; $y \longmapsto 2Y_0-4$	$X_0 + Y_0 \leq 10 \land 2X_0 + 2Y_0 > 15$
$13 \rightarrow 17$	$x \longmapsto 3X_0+9$; $y \longmapsto 2Y_0-4$	$X_0 + Y_0 \leq 10 \land 2X_0 + 2Y_0 > 15$

PATH 4

Edge	Symbolic State (PV)	Path Condition (PC)
$1 \rightarrow 2$	$x \longmapsto X_0$; $y \longmapsto Y_0$	true
$2 \rightarrow 5$	$x \longmapsto X_0$; $y \longmapsto Y_0$	$X_0 + Y_0 \leq 10$

$5 \to 6$	$x \mapsto X_0$; $y \mapsto Y_0+7$	$X_0 + Y_0 \leq 10$
$6 \to 7$	$x \mapsto X_0-3$; $y \mapsto Y_0+7$	$X_0 + Y_0 \leq 10$
$7 \to 9$	$x \mapsto X_0-1$; $y \mapsto Y_0+7$	$X_0 + Y_0 \leq 10$
$9 \to 11$	$x \mapsto X_0-1$; $y \mapsto Y_0+7$	$X_0 + Y_0 \leq 10 \wedge 2X_0 + 2Y_0 \leq 15$
$11 \to 14$	$x \mapsto X_0-1$; $y \mapsto Y_0+7$	$X_0 + Y_0 \leq 10 \wedge 2X_0 + 2Y_0 \leq 15$
$14 \to 15$	$x \mapsto 4X_0-4$; $y \mapsto Y_0+7$	$X_0 + Y_0 \leq 10 \wedge 2X_0 + 2Y_0 \leq 15$
$15 \to 16$	$x \mapsto 4X_0-4$; $y \mapsto 4X_0+3Y_0+17$	$X_0 + Y_0 \leq 10 \wedge 2X_0 + 2Y_0 \leq 15$
$16 \to 17$	$x \mapsto 4X_0-4$; $y \mapsto 4X_0+3Y_0+17$	$X_0 + Y_0 \leq 10 \wedge 2X_0 + 2Y_0 \leq 15$

c) PATH 1: $X_0 + Y_0 > 10 \wedge 2X_0 + 2Y_0 > 25$, feasible
$X_0 = 10$, $Y_0 = 10$

PATH 2: $X_0 + Y_0 > 10 \wedge 2X_0 + 2Y_0 \leq 25$, feasible
$X_0 = 1$, $Y_0 = 10$

PATH 3: $X_0 + Y_0 \leq 10 \wedge 2X_0 + 2Y_0 > 15$, feasible
$X_0 = 4$, $Y_0 = 4$

PATH 4: $X_0 + Y_0 \leq 10 \wedge 2X_0 + 2Y_0 \leq 15$, feasible
$X_0 = 1$, $Y_0 = 1$

Question 2

a) $\neg a_1 \vee \neg a_2$ $\neg a_1 \vee \neg a_3$ $\neg a_1 \vee \neg a_4$ $\neg a_2 \vee \neg a_3$ $\neg a_2 \vee \neg a_4$ $\neg a_3 \vee \neg a_4$

b) Every vertice can be allocated a propositional variable a_v. According to syntax of propositional logic, $F_1 \to F_2$ is equal to $\neg F_1 \vee F_2$. Therefore, for propositional variable $a_v \to a_u$, " $\neg a_v \vee a_u$" means from vertice v to u is reachable. When the path between v_{init} and v_{end} is reachable, according to CNF, if and only if $a_{init} \wedge a_{end}$=true, the path is satisfied.

So the left of graph Figure 1 can be represented as:
Satisfied path 1: $\neg a_{init} \vee a_1$ $\neg a_1 \vee a_{end}$ $a_{init} \wedge a_{end}$
Satisfied path 2: $\neg a_{init} \vee a_2$ $\neg a_2 \vee a_{end}$ $a_{init} \wedge a_{end}$
Which can be simplified as : $\neg a_{init} \vee a_1 \vee a_2$ $\neg a_{init} \vee a_{end}$

The right of graph Figure 1 can be represented as:
Unsatisfied Path 1: $\neg a_{init} \vee a_1$ $\neg a_{end} \vee a_1$ $a_{init} \wedge a_{end}$
Unsatisfied Path 2: $\neg a_{init} \vee a_2$ $\neg a_{end} \vee a_1$ $a_{init} \wedge a_{end}$

In conclusion, the satisfied path can be showed as $\neg a_{init} \vee a_v$ $\neg a_v \vee a_u$ $a_{init} \wedge a_{end}$
Simplified as: $\neg a_v \vee a_u$ $a_{init} \wedge a_{end}$ $(v, u \in V)$

c) at-most-one(a1, a2, ⋯⋯, ai, ⋯⋯an)
List all clauses:
$\neg a_i \vee \neg a_j$ (i from 1 to n, j from i to n, $i \leq j$)

Question 3

a) Assume : constint/$_1$—term is a constant integer
int/$_1$—term is an integer
$\le$/$_2$—one term is equal or less than the other term
value/$_2$—the value of x th row and y th column in magic square
sum/$_n$—the sum of all n values
r/$_0$—row
c/$_0$—column
n/$_0$—size of columns or rows of the square grid

Constrains of magic square:

1) $int(n) \wedge 1 \le n$
2) $int(x) \wedge 1 \le x \le n$
3) $int(y) \wedge 1 \le y \le n$
4) $int(q) \wedge 1 \le y \le n$
5) $int(p) \wedge 1 \le y \le n$
6) $constint(r) \wedge 1 \le r \le n$
7) $constint(c) \wedge 1 \le c \le n$
8) $value(x, y) \neq value(p, q)$
9) $1 \le value(x,y) \le n^2$
10) $sum(value(r, 1), value(r, 2), \ldots, value(r, n)) = n(n^2+1)/2$
11) $sum(value(1, c), value(2, c), \ldots, value(n, c)) = n(n^2+1)/2$
12) $x = y \implies sum(value(x, y)) = n(n^2+1)/2$
13) $x+y = n \implies sum(value(x, y)) = n(n^2+1)/2$

Question 3

e) My code is in test_sym.py, which is the last test unit.

Question 5

a) According to the algebraic rules of FOL:

$$
\begin{aligned}
& \forall x \cdot \exists y \cdot P(x) \vee Q(y) \\
\equiv\ & \forall x \cdot \exists y \cdot (P(x) \vee Q(y)) \\
\equiv\ & \forall x \cdot (\exists y \cdot (P(x) \vee \exists y \cdot Q(y)) \\
\equiv\ & (\forall x \cdot (P(x)) \vee (\exists y \cdot Q(y))
\end{aligned}
$$

Therefore, the FOL must be valid.

b) The FOL is invalid. For example:
Let domain $S=\{a, b\}$. Assume model of P is $P^M=\{(a, b)\}$, model of Q is $Q^M=\{(b, b)\}$. Because when x=a, P(a,y) is true, Q(a, y) is false. So this formula is true. When x=b, P(b,y) is false, Q(b,b)is true. So this formula is also true. Therefore $M=\{S, P^M, Q^M\}$, $M \vDash (\forall x \cdot \exists y \cdot P(x,y) \vee Q(x,y))$.

However, for right formula, when $x=b$, $\exists y \cdot P(b, y)$ is false. When $x=a$, $\exists y \cdot Q(a, y)$ is false. Therefore, M of left formula is not M of right formula. Then the sentence is false.

c) 1, According to FOL formula and model (P_1):

$x<y$ and $z<y$ and $x<z$ and $z \geq x$

Simplify to say, $x<z<y$ is the model. So satisfy.

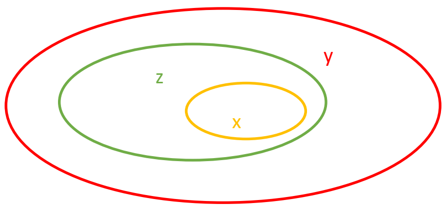
2, According to FOL formula and model (P_2):

$y=x+1$ and $y=z+1$ and $z=x+1$ and $x \neq z+1$

Simply to say, $x=z$ and $z= x+1$ conflict. So not satisfy.

3, According to FOL formula and model (P_3):

$x \subseteq y$ and $z \subseteq y$ and $x \subseteq z$ and $z \subsetneq x$. The model can be represented by this figure.



d) $isArray(A) \land 0<i<len(A)$

$\forall x, \forall y \cdot 0 \leq x<i<y<len(A) \Longrightarrow read(A,x)<read(A,y)$

e) $isArray(A) \land isArray(B) \land (len(A)=len(B))$

$(\forall x \cdot 0 \leq x<len(A)) \Rightarrow (\exists y \cdot 0 \leq y<len(B)) \land (read(A,x)=read(B,y))$

$(\forall y \cdot 0 \leq y<len(A)) \Rightarrow (\exists x \cdot 0 \leq x<len(B)) \land (read(B,y)=read(A,x))$

f) Define isOnStack(x, y) means to check y is on stack x so return true.

Define isNotOnStack(x, y) means to check y is not on stack x so return true.

$x=nil \Longrightarrow empty(x)=true$

$\forall x \cdot empty(x) \Longrightarrow \neg empty(push(x, y))$

$\forall x, y \cdot top(push(x, y))=y$

$\forall x, y \cdot pop(push(x, y))=x$

$isOnStack(push(x, y),y)=true$

$isNotOnStack(pop(x), top(x))=true$