**ITCS 6150 Final Exam** (Max 40 points) **Name:**
**SIN:**

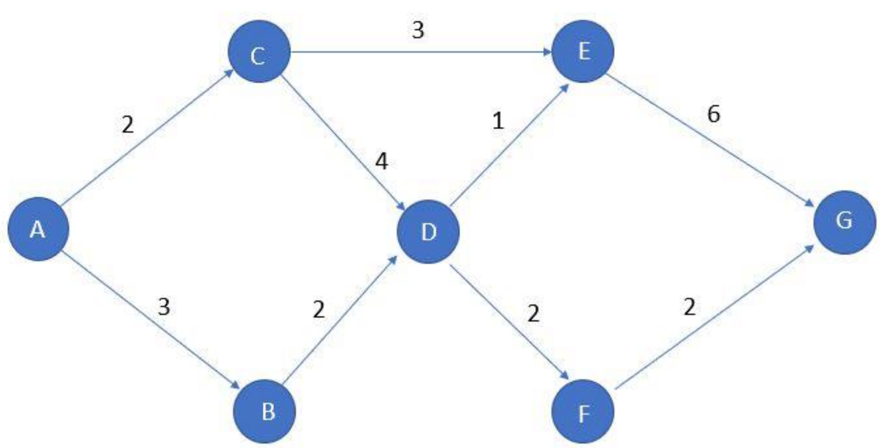
**Problem 1 (max 10).** Suppose we know that:

- If *Paolo is thin*, then *Carlo is* not *blonde* or *Roberta is* not *tall*
- If *Roberta is tall* then *Sandra is lovely*
- If *Sandra is lovely* and *Carlo is blonde* then *Paolo is thin*
- *Carlo is blonde*

Can we deduce that *Roberta is* not *tall*?

Solution:
A – Paolo is thin. C – Carlo is blonde. R – Roberta is tall.
S – Sandra is lovely.
We need to check if
KB = { A -> (~C v ~R), R -> S, (S ^ C) -> A, C } |- ~R
Converting to set of clauses (5 points) & use resolution. (5 points)
(1) ~A v ~C v ~R; (2) ~R v S; (3) ~S v ~C v A; (4) C; (5) R;
Resolution:
(6) = (1)+(4)+(5) ~A; (7) = (2)+(5) S; (8) = (3)+(4)+(6) ~S; (7)+(8) NIL

**Problem 2 (max 10)**. In what order the algorithm using distance function f(n) will visit nodes in the graph G assuming that A is the starting node and G is the final node in the graph. We assume that f(n) = g(n) + h(n), where g(n) is the shortest distance from node A to node n (n ∈ {B, C, D, E, F, G}) and function h is defined by the table below. Function h(x) is a heuristic estimation of the distance from node x to the final node.




Graph G

| h(A) | h(B) | h(C) | h(D) | h(E) | h(F) | h(G) |
|---|---|---|---|---|---|---|
| 7 | 5 | 6 | 4 | 6 | 2 | 0 |

Check if f(n) is admissible (never overestimates the actual cost to get from node n to the goal) [2 points out of 10] and monotonic [2 points out of 10]

Solution:

(A -> C, 2+h(C)=8), (A->C->E, 11), (A->C->D, 10)

(A -> B, 8)

Monotonic: h(A) = 7 =< d(A,C)+h(C)=2+ 6 = 8

Admissible: h(A)= 7 =< d(A,G)=9

---

(A -> C, 2+h(C)=8), (A->C->E, 11), (A->C->D, 10)

(A -> B, 8), (A->B-> D, 9)

Monotonic: h(A) = 7 =< d(A,B)+d(B) =3 + 5 = 8

---

(A -> C, 2+h(C)=8), (A->C->E, 11), (A->C->D, 10)

(A -> B, 8), (A->B-> D, 9), (A->B->D->E, 12), (A->B->D->F, 9)

Monotonic: h(B)=5 =< d(B,D) + h(D) = 2 + 4 = 6

Admissible: h(B) = 5 =< d(B,G)= 6

---

(A -> C, 2+h(C)=8), (A->C->E, 11), (A->C->D, 10)

(A -> B, 8), (A->B-> D, 9), (A->B->D->E, 12), (A->B->D->F, 9), (A->B->D->F->G, 9)

Monotonic: h(D) = 4 =< d(D,F) + h(F) = 2 + 2 = 4

Admissible: h(D) = 4 =< d(D,G) = 4

---

(A -> C, 2+h(C)=8), (A->C->E, 11), (A->C->D, 10)

(A -> B, 8), (A->B-> D, 9), (A->B->D->E, 12), (A->B->D->F, 9), (A->B->D->F->G, 9)

Monotonic: h(F) = 2 =< d(F,G) + h(G) = 2 + 0

---

**Problem 3 (max 12).**

**Let's assume that:**

1) Anyone whom Mary loves is a football star.
2) Any student who does not pass does not play.
3) John is a student.
4) Any student who does not study does not pass.
5) Anyone who does not play is not a football star.

Use resolution to prove that: If John does not study, then Mary does not love John.

To what class (Breadth-First, Set-of-Support, Unit-Preference, Linear-Input Form, Ancestry-Filtered Form) does your strategy belong? List all.

**Use the following alphabet:**

Constants: M – Mary, J- John,

Predicates: S(x) – x is a student, ML(x) – Mary loves x

F(x) – x is a football star, St(x) – x does not study,

Play(x) – x does not play, Pass(x) – x does not pass.

**KB: Knowledge Base**

ML(x) -> F(x). (S(x) ^ Pass(x)) -> Play(x).

S(John). (S(x) ^St(x)) -> Pass(x). Play(x) -> ~F(x).

**Prove that**:

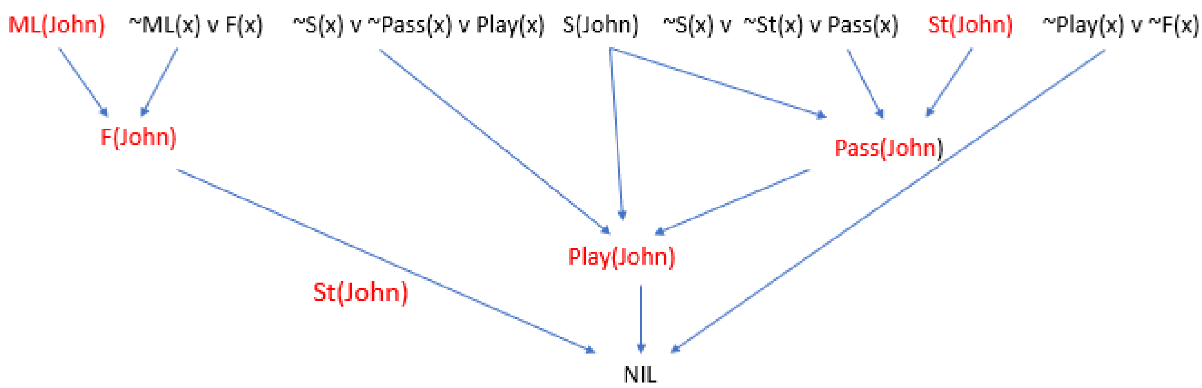
St(John) -> ~ML(John).

**Converting KB to the Set of Clauses + Negation of the Goal** (max 4 points)

~ML(x) v F(x). ~S(x) v ~Pass(x) v Play(x).

S(John). ~S(x) V ~St(x) v Pass(x). ~Play(x) v ~F(x). St(John). ML(John)

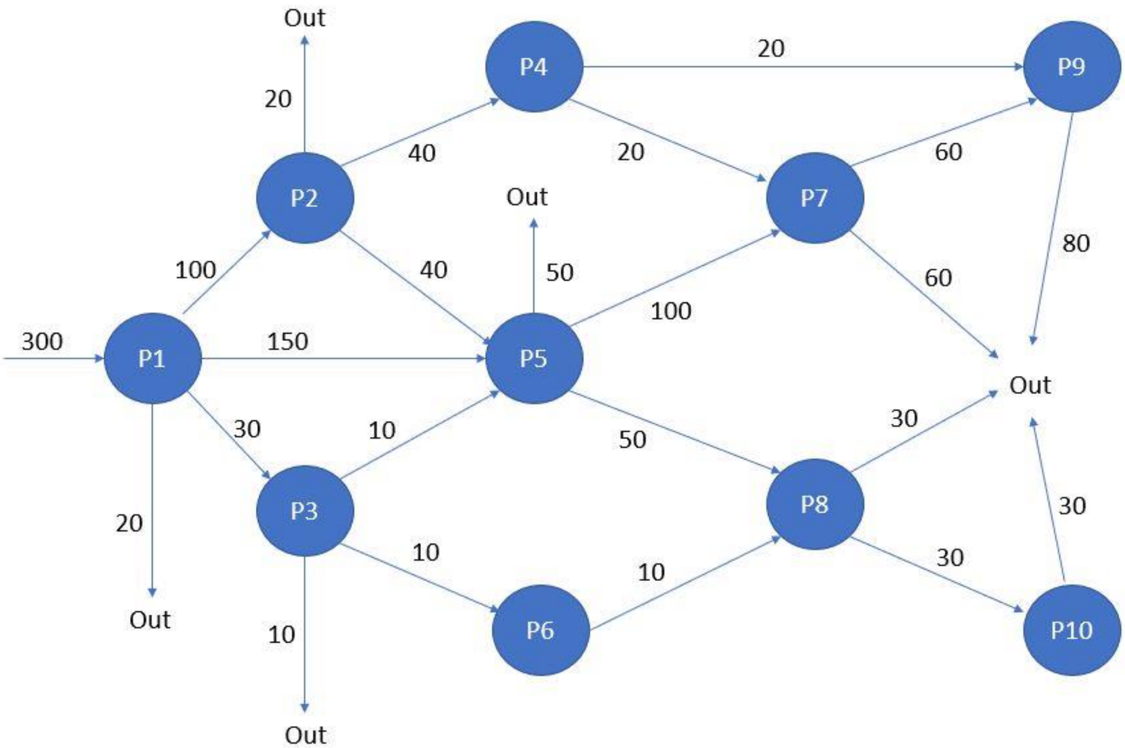
**Resolution**.(max 5 points)




Resolution belongs to the following classes: linear-input, unit-preference, & set-of-support.(max 3 points)

**Problem 4 (max 8).** Assuming that the procedure graph for surgery P1 is represented by the figure below, find the score associated with a surgery P3 (the average number of readmissions to hospital after surgery P3).

**Remark**: Out of 300 patients having surgery P1, 20 are fine, 100 returned to the hospital for surgery P2, 150 for surgery P5, and 30 for surgery P3.




**Solution**:

Score(P9) = 8/8*0 = 0
Score(P7) = 6/12[1 + Score(P9)] + 6/12*0 = ½
Score(P(10)) = 3/3*0= 0
Score(P8) = 3/6*0 + 3/6[1 + Score(P10)] = 1/2

Score(P5) = 5/20*0 + 10/20[1 + Score(P7)] + 5/20[1 + Score(P8)] =
½[1 + ½] + ¼[1 + ½] = ½*3/2 + ¼*3/2 = ¾*3/2 = 9/8

Score(P6) = 1*[1+ Score(P8)] = 3/2

Score(P3) = 1/3*0 + 1/3*[1 + Score(P6)] + 1/3*[1 + Score(P5)] =
1/3*[1 + 3/2] + 1/3*[1 + 9/8] = 1/3*5/2 + 1/3*17/8 = 1/3[20/8 + 17/8] =(1/3)*(37/8)
= 37/24.

Score(P4) = 2/4*[1 + Score(P9)] + 2/4*[1 + Score(P7)] = ½*[1+0] + ½*[1+ ½] =
½ + ½*3/2 = 2/4 + ¾ = 5/4

Score(P2) = 2/10*0 + 4/10*[1 + Score(P4)] + 4/10*[1 + Score(P5)] =
4/10*[1 + 5/4] + 4/10*[1 + ½] = 2/5*9/4 + 2/5*[1 + 9/8] = 18/20 + 2/5*17/8 =
18/20 + 17/20 = 35/20 = 7/4

Score(P1) =
2/30*0 + 1/3*[1 + Score(P2)] + ½[1 + Score P(5)] +1/10*[1 + Score (P3)] =
1/3*[1 + 7/4] + ½*[1 + 9/8] + 1/10*[1 + 37/24] = 1/3*11/4 + ½*17/8 + 1/10*61/24 =
11/12 + 17/16 + 61/240

(If calculation mistakes, drop 1 point. If the strategy is correct but the solution is not finished, drop 2 points. If strategy is wrong, drop 5 points)