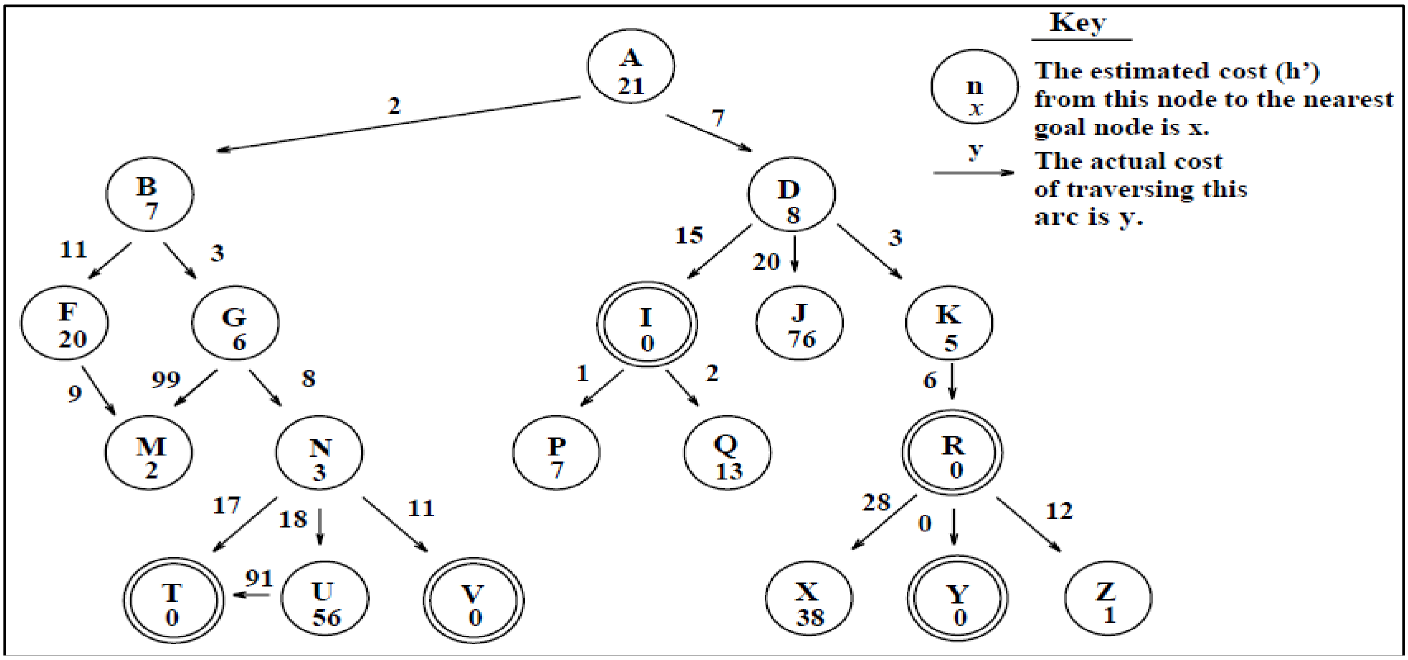
Question 1: Consider the search graph drawn below. The initial state is "A", and goal states are represented by double circles. Values on the arcs represent the actual cost, whereas the values side the states represent the heuristic function's value). **For each of the search strategies listed below, list, in order, the states expanded until a goal state is reached (if any) AND the path found only** (you can use scratch papers for queue traces). Assume that the successor function returns a state's successors in the alphabetic order. **[60 points]**



A. Uniform Cost Search

List of expanded nodes: A, B, G, D, K, F, N, R

Path: ADKR

B. Greedy Search

List of expanded nodes: A, B, G, M, N, T

Path: ABGNT

C. A*

List of expanded nodes: A, B, G, D, K, R

Path: ADKR

D. Iterative Deepening Search (IDS)

A ABD ABFGDI

Path: ADI

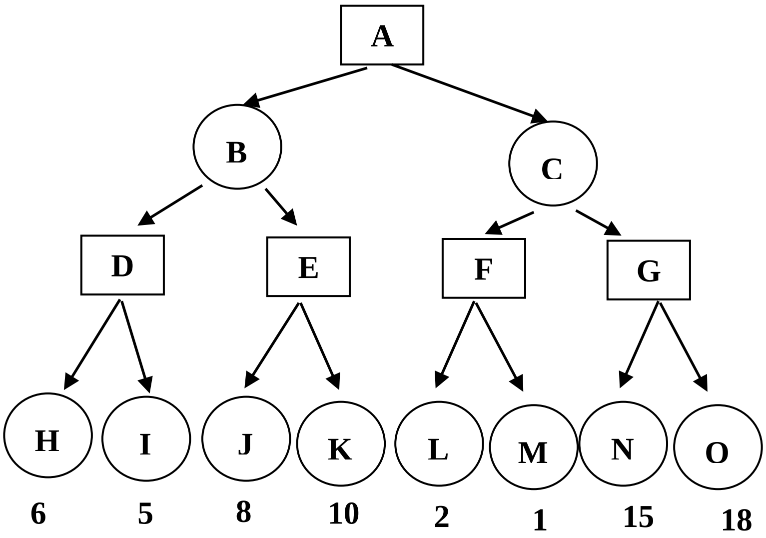
E. BFS

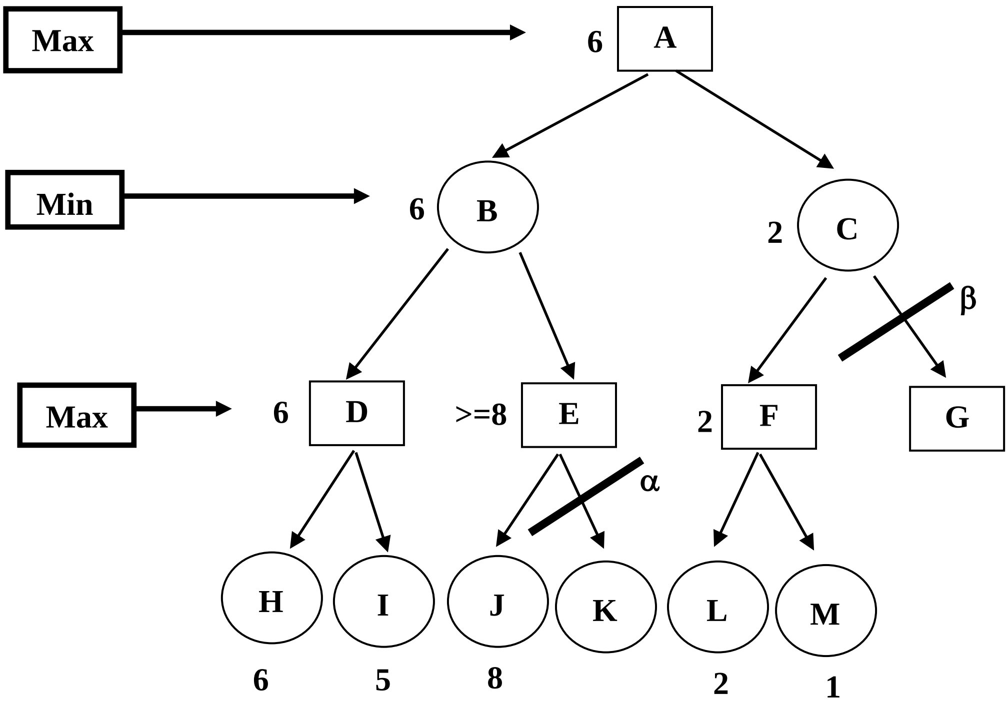
ABDFGI

Path: ADI

Question 2:

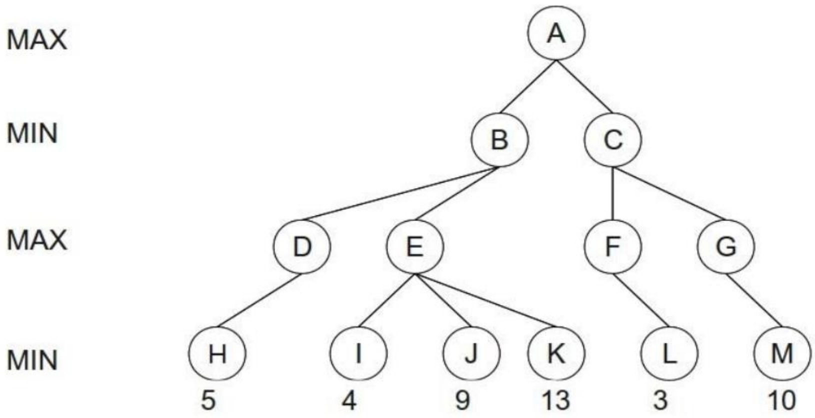
Given the following search tree (Max is represented by rectangle and min by a circle), apply the alpha-beta pruning algorithm to it and show the search tree that would be built by this algorithm. Make sure that you show where the alpha and beta cuts are applied and which parts of the search tree are pruned as a result.





Question 3: Minimax Algorithm (15 points)

Perform Minimax on this tree. Write the Minimax value associated with each node in the box below, next to its corresponding node letter.(1.5 points each)



A= 5	**B=** 5	**C=** 3	**D=** 5	**E=** 13	**F=** 3	**G=** 10

What path will be selected by MAX? Assume that MIN and MAX are both rational agents.

A – B – D – H (4.5 points)

Question 4:

Convert the following sentences to FOL

You should use the following predicates:
Student (x) means x is a student. Love (x, y) means student x loves student y. Takes (x, y) means that student x takes course y.

a. There is no student who is loved by all other students.

¬[∃x. Student (x) ∧ (∀y (Student (y) ∧ (x≠y) => Love(y,x)))]

b. Every student loves some student

∀x Student (x) => ∃y. Student (y) ∧ Love (x,y)

c. Each student must take at least two courses.

∀x Student (x) =>(∃y ∃z Takes(x,y) ∧ Takes(x,y) ∧ ¬ (y=z))

d. Each student must take at most one course.

∀x∀y∀z Student (x) ∧Takes(x,y) ∧Takes (x,z) => x=y

e. Suzan takes every course that Bill takes.

∀x Takes (Bill, x) => Takes (Suzan,x)

f. There is a student who does not take any courses.

∃x. (∀y Student (x) => ¬Takes(x,y))

Question 4:

Given the following English sentences:

Since handwritten solutions are not accepted, here are the symbols you may need: ∀ ∃ ∧ ∨ ¬ ⇒ ⇔

A. Every kid loves Spiderman.
B. Everyone who loves Spiderman loves any spider.
C. Spindra is a spider, and Spindra has a red back.
D. Anything which has a red back is creepy or is a scary.
E. No spider is scary.
F. John does not love anything which is creepy.

1. Convert each sentence to FOL using the following predicates:
   - Unary (one parameter): child(x), spider(x), redback(x), creepy(x), scary(x)
   - Binary (2 parameters): loves(x,y)

   Note: Sentence E should be converted into FOL using two forms.

Part 1)

A. Every kid loves Spiderman.

   $\forall x\ (child\ (x) \rightarrow loves(x,\ \text{Spiderman}))$

B. Everyone who loves Spiderman loves any spider.

   $\forall x\ (loves\ (x,\ \text{Spiderman}) \rightarrow \forall y\ (\text{spider}\ (y) \rightarrow loves\ (x,\ y)))$

C. Spindra is a spider, and Spindra has a red back.

   spider(Spindra) ∧ redback(Spindra)

D. Anything which has a red back is creepy or is a scary.

   $\forall x\ (\text{redback}\ (x) \rightarrow \text{creepy}\ (x) \lor \text{scary}(x))$

E. No spider is scary.

   $\neg\ \exists x\ (\text{spider}\ (x) \land\ \text{scary}(x))$

   $\forall x\ \neg\text{spider}\ (x) \lor \neg\text{scary}(x)$ *or* $\forall x\ spider(x) \Rightarrow \neg scary(x)$

F. John does not love anything which is creepy.

   $\forall x\ (\text{creepy}\ (x) \rightarrow \neg\text{loves}\ (\text{John},\ x))$