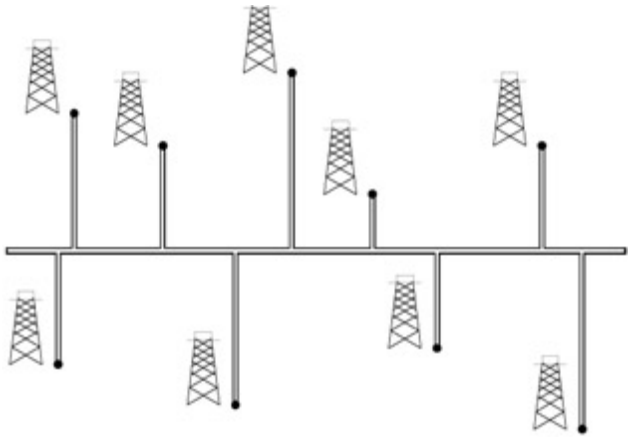
Part II: Analysis and Design of Algorithms

*Answer as much as you can. You can attempt parts of a question. Use pseudo-code or code to describe required algorithms. Maximum grade = **50 pts.***

Question 1 [**14 Marks**] Professor Olay is consulting for an oil company, which is planning a large pipeline running east to west through an oil field of *n* wells. From each well, a spur pipeline is to be connected directly to the main pipeline along a shortest path (either north or south), as shown in the Figure below. In other words, Professor Olay needs to determine the position of the east-west oil pipeline that minimizes the total length of the north-south spurs.



a. **[3 marks] Define** the selection (order statistic) problem.
b. **[4 marks] Give** an O(n) deterministic algorithm to find the median of *n* integers.
c. **[4 marks]** Given x- and y-coordinates of the wells, give an O(n) algorithm to pick the optimal location of the main pipeline (the one that minimizes the total length of the spurs)?
d. **[3 marks]** Using the deterministic algorithm in part (b), show how to make the quicksort algorithm run in O(n lg n) time in the worst case.

Question 2 [**15 Marks**] In this problem, you are given pseudo-code for an algorithm that takes a graph *G* with edge weights *w* as input and returns a set of edges *T*.

```
MAYBE-MST(G, w)
 1 sort the edges into non-increasing order of edge weights w
 2 T ← E
 3 for each edge e, taken in non-increasing order by weight
 4    do if T - {e} is a connected graph
 5          then T ← T - e
 6 return T
```

a. **[3 marks] Define** the minimum spanning tree problem.
b. **[6 marks]Would** the algorithm MAYBE-MST produce a minimum spanning tree? **Prove** your answer.
c. **[6 marks] Show** how you would implement each step of the algorithm in the most efficient way. **What** is the running time of the whole algorithm in terms of the number of graph edges, *m,* and the number of graph vertices, *n.* **What** is the running time of the whole algorithm in terms of *n* **only**?

Question 3 [**36 Marks**] An independent set of a graph $G = (V, E)$ is a subset $V' \subseteq V$ of vertices such that each edge in E is incident on **at most one** vertex in V'. The independent-set problem is to find a **maximum** size independent set in G.

a. **[3 marks] Give** an example of the input and output of the independent-set problem.
b. **[2 mark] Choose:** The independent-set problem is a(n) _____ problem (optimization - decision)
c. **[3 marks] Formulate** a related decision problem for the independent-set problem.
d. **[5 marks] Sketch** how to reduce to the decision problem you defined in part (c) from the clique problem. **Why** would you do that reduction?
e. **[5 marks]** Suppose that you are given a "black-box" subroutine to solve the decision problem you defined in part (c). **Give** an algorithm to find an independent set of maximum size. **What** is the running time of your algorithm, where each queries to the black box is counted as a single step.
f. **[3 marks]** Draw an example of a graph in which each vertex has degree 2.
g. **[6 marks] Give** an efficient (i.e., polynomial time) algorithm to solve the independent-set problem when each vertex in G has degree 2. **Analyze** the running time, and **prove** that your algorithm works correctly.
h. **[3 marks]** Draw an example of a bipartite graph.
i. **[6 marks] Give** an efficient algorithm to solve the independent-set problem when G is bipartite. **Analyze** the running time, and **prove** that your algorithm works correctly.