Problem 5

Collaborators: List students that you have discussed problem 5 with: Vishnu Teja Narapareddy, Tulika Sureka

1. P(Yes|Senioritis) = 4/5
   P(No|Senioritis) = 1/5
   P(Yes|No Senioritis) = 0
   P(No|No Senioritis) = 1

   Given p = 1-$\frac{1}{poly(n)}$
   So, we see that algorithm A always provides correct answer if the person does not have senioritis. However, if the person has senioritis, then the algorithm detects it with probability 4/5.
   In this part, we need to increase this probability to p.

   For this problem, we output YES if the algorithm gives yes in any of the T trials, otherwise, we output NO.
   The probability of not detecting the disease, given that the person has disease = 1/5
   If we run this algorithm multiple times, let say for T = $\log_5(poly(n))$:

   $$\begin{aligned}\text{probability of failure for all the T iterations} &= (\frac{1}{5})^T \\ &= (\frac{1}{5})^{\log(poly(n))} \\ &= (\frac{1}{poly(n)})^{\log_5 5} \\ &\leq \frac{1}{poly(n)}\end{aligned}$$

   Probability that algorithm predicts correctly when the person has disease $\geq 1 - \frac{1}{poly(n)}$
   Probability that algorithm predicts correctly when the person does not has disease $= 1 \geq 1 - \frac{1}{poly(n)}$ as the algorithm A always gives the correct results for NO instances.

   Analysing TC:
   We know that algorithm A runs in O(n). As in the algorithm discussed above, we run the algorithm A for T = ($\log(poly(n))$) trials, so we can say that the overall complexity of the proposed algorithm is O(n $\log(poly(n))$)

2. P(Yes|Senioritis) = 4/5
   P(No|Senioritis) = 1/5

P(Yes|No Senioritis) = 1/5
P(No|No Senioritis) = 4/5

Given p = 1-$\frac{1}{poly(n)}$
So, we see that algorithm A provides correct answer if the person does not have senioritis with a probability of 4/5. Also, if the person has senioritis, then the algorithm detects it with probability 4/5.

In this part, we need to increase both these probabilities to p.

For this problem, we output the majority answer which we get after T trials.

The probability of not detecting the disease, given that the person has disease = 1/5. The probability of detecting the disease, given that the person does not has disease = 1/5
If we run this algorithm multiple times, let say for T = $\frac{50}{9}\log(poly(n))$:

$$
\begin{aligned}
\text{probability of failure in majority of the T iterations} &= \sum_{i=0}^{T/2} \binom{T}{i} (\frac{4}{5})^i (\frac{1}{5})^{T-i} \\
&<= \sum_{i=0}^{T/2} \binom{T}{i} (\frac{4}{5})^{T/2} (\frac{1}{5})^{T/2} \\
&= (\frac{4}{25})^{T/2} \sum_{i=0}^{T/2} \binom{T}{i} \\
&<= (\frac{4}{25})^{T/2} 2^T \\
&= (1-\frac{9}{25})^{T/2} \quad \text{by using the inequality } 1-x \leq e^{-x} \\
&<= e^{-\frac{9}{50}T} \\
&<= \frac{1}{poly(n)}
\end{aligned}
$$

Probability that algorithm predicts correctly $\geq 1-\frac{1}{poly(n)}$
Hence, the algorithm would now give the correct result whether the person has disease or not with a probability atleast p.

Analysing TC:
We know that algorithm A runs in O(n). As in the algorithm discussed above, we run the algorithm A for T = $(\log(poly(n)))$ trials, so we can say that the overall complexity of the proposed algorithm is O(n $\log(poly(n))$)

3. For this part, consider q = 1/2 + ε
Now, we need to show that there exists an algorithm which runs in polynomial time which gives correct answers with probability atleast 3/4.

Using the same algorithm as used in part (b).
We run this algorithm multiple times, let say for T = $\frac{1}{2\varepsilon^2}\log(4)$:

$$
\begin{aligned}
\text{probability of failure in majority of the T iterations} &= \sum_{i=0}^{T/2} \binom{T}{i} (\frac{1}{2}+\varepsilon)^i (\frac{1}{2}-\varepsilon)^{T-i} \\
&<= \sum_{i=0}^{T/2} \binom{T}{i} (\frac{1}{2}+\varepsilon)^{T/2} (\frac{1}{2}-\varepsilon)^{T/2} \\
&= (\frac{1-4\varepsilon^2}{4})^{T/2} \sum_{i=0}^{T/2} \binom{T}{i} \\
&<= (\frac{1-4\varepsilon^2}{4})^{T/2} 2^T \\
&= (1-4\varepsilon^2)^{T/2} \quad \text{by using the inequality } 1-x \leq e^{-x} \\
&<= e^{-2\varepsilon^2 T} \\
&<= \frac{1}{4}
\end{aligned}
$$

Probability that algorithm predicts correctly $\geq 1-\frac{1}{4}$
Hence, the algorithm would now give the correct result whether the person has disease or not with a probability atleast 3/4.