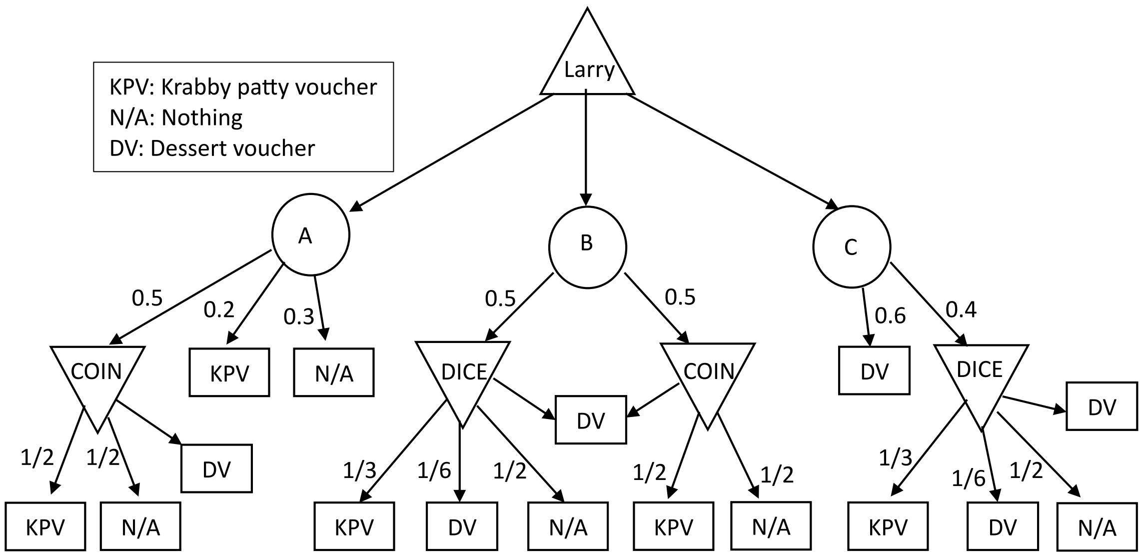
PSET 2

Anusha Murali **CS182** **October 10, 2023**

Problem 1 Krubby Patty

(1) Decision Tree of the problem:



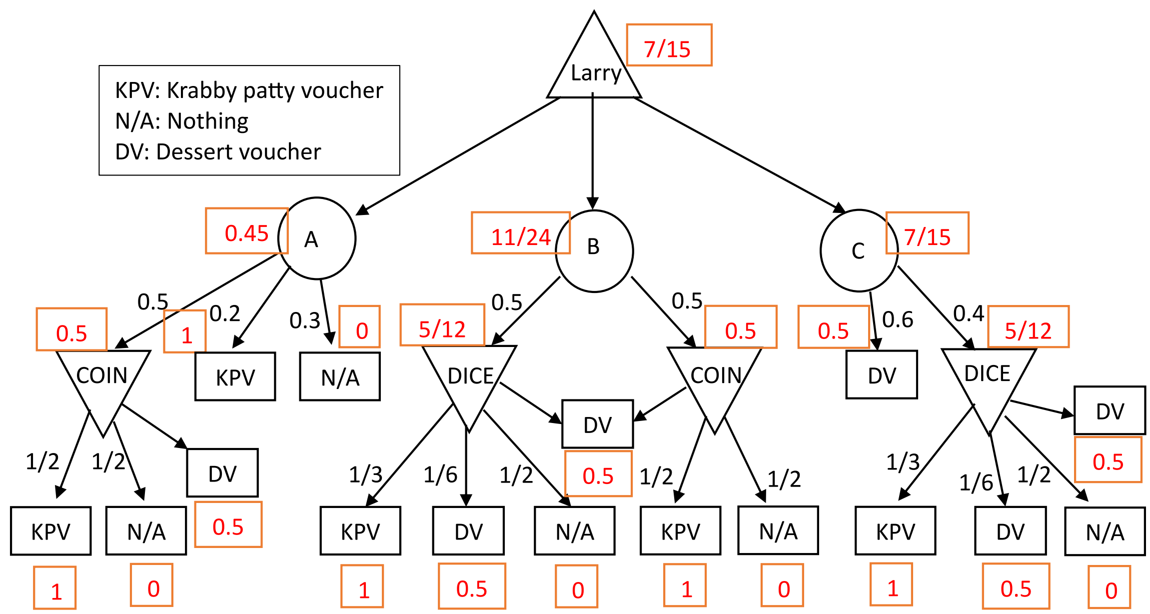
(2) We assume both Larry and Mr. Krabs are playing optimally. Whenever the cup chosen by Larry contains either a dice or a coin, Mr. Krabs can choose to give a dessert voucher, instead of taking his chances of throwing dice or flipping coin. Hence, giving a dessert voucher is the most optimal action for Mr. Krabs as it results in zero probability of Larry winning a Krabby Patty voucher on a dice roll or coin flipping.

Therefore, assuming Mr. Krabs is playing optimally, using the above argument, we find that Mr. Krabs will always give a dessert voucher if Larry's choice of cup contains either a coin or dice.. We note that cup A can contain a krabby patty voucher with probability 0.2, while cup B and C contain no krabby patty vouchers. Moreover, cup B and C can have a coin or dice, and if Larry selects either of them, there is a chance that Mr. Krabs may give a dessert voucher. Hence using Expectminimax algorithm, since Larry wants to maximize his chances of winning a krabby patty, he will choose cup A as $0.2 > 0$.

In the event a dessert voucher is more expensive than a krabby patty voucher, Mr. Krabs will decide to flip the coin or roll the dice. In this case, the utility value of winning krabby patty via cup A is $0.2 + 0.5 \times 0.5 = 0.45$, the utility value of winning krabby patty via cup B is $0.5 \times \frac{1}{3} + 0.5 \times \frac{1}{2} = \frac{5}{12}$, and the utility value of winning krabby patty via cup C is $0.4 \times \frac{1}{3} = \frac{2}{15}$. Since the utility value of winning krabby patty via cup A is larger than winning via cup B or cup C, and since Larry is playing optimally, he will select cup A, maximizing his chances of winning.

Hence, using the given probabilities in the problem, using the Expectiminimax algorithm guarantees an optimal strategy for Larry in the game, assuming Mr. Krabs plays optimally himself.

(3) If the dollar to play is worth 0.4, a krabby patty voucher is worth 1, a dessert voucher is worth 0.5, and winning nothing is worth 0, using the following decision tree of the expectiminimax game, we can show that it is worth it to play the game.



The chance nodes are shown in circles, the minimizer nodes are shown in upside-down triangles, and the maximizer node is shown in upright triangle. The minimizer nodes correspond to Mr. Krabs' actions, and the maximizer node corresponds to Larry's action.

The payoffs at the leaf nodes and the expected payoffs at the minimizer nodes are shown in red boxes. For the minimizer nodes (which correspond to Mr. Krab's actions), from left to right, the expected payoffs are computed as follows:

(a) Left most coin: Mr. Krabs either gives a dessert voucher of value 0.5 or flips the coin. If he flips the coin, we get a krabby patty voucher (value 1) with probability 1/2, which has an expected payoff of $1 \times 1/2 = 0.5$. Hence, Mr. Krabs can either decide to give a dessert voucher of value 0.5 or flip the coin with the expected payoff 0.5. Therefore, the expected payoff of the left most coin is 0.5.

(b) Left dice: Using the argument as above, we find that the expected payoff of the left dice is 5/12.

(c) Right coin: Using similar arguments, we find that the expected payoff of the right coin is 0.5.

(d) Right dice: Using the argument as above, we find that the expected payoff of the left dice is 5/12.

Using the expected payoffs computed for the two coins and two die, we can now compute the expected payoffs for cups A, B, and C as follows:

(a) Expected payoff for cup A: $0.5 \times 0.5 + 1 \times 0.2 = 0.45$.

(b) Expected payoff for cup B: $\frac{5}{12} \times 0.5 + 0.5 \times 0.5 = \frac{11}{24}$.

(c) Expected payoff for cup C: $0.5 \times 0.6 + \frac{5}{12} \times 0.4 = \frac{7}{15}$.

Now, applying the expectiminimax algorithm, starting from the root node, we travel to the left most minimizer node, which has an expected payoff of 0.6. Its successors are the krabby patty voucher with an expected payoff of 1 and "nothing" with an expected payoff of 0. Therefore, we assign the cumulative expected pay off of $0.5 \times 0.5 + 0.2 + 0 = 0.45$ to their parent node A. Then we backtrack and reach the root node and then travel to node B and its two children minimizer nodes, namely the dice and the coin. As before we compute the cumulative payoff of their parent node B, which is $\frac{5}{12} \times 0.5 + 0.5 \times 0.5 = \frac{11}{24}$. We again backtrack, reach the root node and then travel to node C and its children. We obtain a cumulative payoff of $0.5 \times 0.6 + \frac{5}{12} \times 0.4 = \frac{7}{15}$ for node C. Now, we backtrack and reach the root node, which is a maximizer node (Larry). Since $\max(A, B, C) = C$, the maximizer chooses node C. The payoff value of node C $= \frac{7}{15} \approx 0.467$. Since this is greater than the dollar to play value of 0.4, we find that it is worth it to play the game. $\square$

PSET 2

Anusha Murali **CS182** **October 10, 2023**

Problem 2 Game Theory

(1) **Pure Nash Equilibrium** In the following diagram, Fiona's optimal choices for each of Paul's choices are indicated in red and similarly Paul's optimal choices for each of Fiona's choices are indicated in blue.

The expected payoff values are shown in red.

We apply the expectiminimax algorithm to the root node, which corresponds to Larry. We recursively traverse down to the left most minimizer node and compute its expected payoff value The expected payoff value is computed by multiplying the utility value of each leaf node and its probability and then summing up the values for all leafs. We then compute the expected payoffs of its sibling nodes in a similar manner. Once the expected payoffs of all the three minimizer nodes attached to the chance node A are computed, the minimum value from these three nodes are propagated to the chance node A above.

	Paul: red	Paul: green	Paul: blue
Fiona: red	F = 8, P = 2	F = 10, P = 3	F = 7, P = 4
Fiona: green	F = 9, P = 11	F = 8, P = 6	F = 8, P = 2
Fiona: blue	F = 8, P = 3	F = 9, P = 4	F = 9, P = -4

For example, from Fiona's perspective, when Paul's choice is red, given the three choices for Fiona in the first column (i.e: F = 8, F = 9 and F = 8), her best choice would be F = 9. This is circled in red in the first column. In the same manner, her choices in the second and third columnswould be F = 10 and F = 9, both of which are circled in red.

Similarly, from Paul's perspective, when Fiona's choice is red, given the three choices for Paul in the first row (i.e: P = 2, P = 3 and P = 4), his best choice would be P = 4. This is circled in blue in the first row. In the same manner, Paul's choices in the second and third rows would be P = 11 and P = 4, both of which are circled in blue.

We notice that F = 9 and P = 11 (Fiona selecting green and Paul selecting red) are both circled in the above payoff matrix. Therefore F = 9 and P = 11 (Fiona selecting green and Paul selecting red) is in Nash Equilibrium. □

Mixed-strategy Nash Equilibrium Let p_1, p_2 be the probabilities of Fiona selecting red and green respectively. Therefore, the probability that she selects blue is $1-p_1-p_2$. Similarly, let q_1, q_2 be the probabilities of Paul selecting red and green respectively. Therefore, the probability that he selects blue is $1 - q_1 - q_2$.

In mixed-strategy Nash Equilibrium, Fiona selects the probabilities p_1 and p_2 such that Paul's choices of the colors will not make any difference in the payoff for him. In the same manner, Paul selects the probabilities q_1 and q_2 such that Fiona's choices of the colors will not make any difference in the payoff for her.

		Paul: red	Paul: green	Paul: blue
	Probabilities	(q_1)	(q_2)	$(1 - q_1 - q_2)$
Fiona: red	(p_1)	F = 8, P = 2	F = 10, P = 3	F = 7, P = 4
Fiona: green	(p_2)	F = 9, P = 11	F = 8, P = 6	F = 8, P = 2
Fiona: blue	$(1 - p_1 - p_2)$	F = 8, P = 3	F = 9, P = 4	F = 9, P = -4

From Fiona's perspective, Paul's payoff for selecting red is,

$$\Pi_P(\text{red}) = p_1(2) + p_2(11) + (1 - p_1 - p_2)(3) = 3 - p_1 + 8p_2$$

Similarly, from Fiona's perspective, Paul's payoff for selecting green is,

$$\Pi_P(\text{green}) = p_1(3) + p_2(6) + (1 - p_1 - p_2)(4) = 4 - p_1 + 2p_2$$

Similarly, from Fiona's perspective, Paul's payoff for selecting blue is,

$$\Pi_P(\text{blue}) = p_1(4) + p_2(2) + (1 - p_1 - p_2)(-4) = -4 + 8p_1 + 6p_2$$

We want Paul's choices of red, green or blue are indifferent to each other. In other words, the payoffs, $\Pi_P(\text{red}), \Pi_P(\text{green})$ and $\Pi_P(\text{blue})$ must be equal to one another or,

$$\Pi_P(\text{red}) = \Pi_P(\text{green}) = \Pi_P(\text{blue})$$

Therefore, equating the above payoffs, we obtain the following two equations:

$$\begin{aligned} 3 - p_1 + 8p_2 &= 4 - p_1 + 2p_2 \\ 4 - p_1 + 2p_2 &= -4 + 8p_1 + 6p_2 \end{aligned}$$

Solving, we obtain, $p_1 = \dfrac{22}{27}$ and $p_2 = \dfrac{1}{6}$.

Similarly, from Paul's perspective, Fiona's payoff for selecting red is,

$$\Pi_F(\text{red}) = q_1(8) + q_2(10) + (1 - q_1 - q_2)(7) = 7 + q_1 + 3q_2$$

Similarly, from Paul's perspective, Fiona's payoff for selecting green is,

$$\Pi_F(\text{green}) = q_1(9) + q_2(8) + (1 - q_1 - q_2)(8) = 8 + q_1$$

Similarly, from Paul's perspective, Fiona's payoff for selecting blue is,

$$\Pi_F(\text{blue}) = q_1(8) + q_2(9) + (1 - q_1 - q_2)(9) = 9 - q_1$$

We want Fiona's choices of red, green or blue are indifferent to each other. In other words, the payoffs, $\Pi_F(\text{red}), \Pi_F(\text{green})$ and $\Pi_F(\text{blue})$ must be equal to one another or,

$$\Pi_F(\text{red}) = \Pi_F(\text{green}) = \Pi_F(\text{blue})$$

Therefore, equating the above payoffs, we obtain the following two equations:

$$
\begin{aligned}
7 + q_1 + 3q_2 &= 8 + q_1 \\
8 + q_1 &= 9 - q_1
\end{aligned}
$$

Solving, we obtain, $q_1 = \dfrac{1}{2}$ and $q_2 = \dfrac{1}{3}$.

Hence, in summary, Fiona should select the probabilities $p_1 = \dfrac{22}{27}$ and $p_2 = \dfrac{1}{6}$ so that Paul's choices of the colors will not make any difference in the payoff for him. In the same manner, Paul selects the probabilities $q_1 = \dfrac{1}{2}$ and $q_2 = \dfrac{1}{3}$ so that Fiona's choices of the colors will not make any difference in the payoff for her. $\square$

(2) Fiona wants to minimize the maximum payoff for Paul. Let p_1, p_2 and p_3 be the probabilities that she uses for selecting red, blue and green respectively. We define the following key terms, which will be required for our program:

(a) **Decision Variables** The decision variables are p_1, p_2 and p_3 that would help Fiona minimize the maximum payoff for Paul.

(b) **Objective Function** Using Fiona's choice of p_1, p_2, p_3, Paul's expected utility for selecting red, green, and blue are as follows:

$$
\begin{aligned}
P_{\text{red}} &= C_{00}p_1 + C_{10}p_2 + C_{20}p_3 = 2p_1 + 11p_2 + 3p_3 \\
P_{\text{green}} &= C_{01}p_1 + C_{11}p_2 + C_{21}p_3 = 3p_1 + 6p_2 + 4p_3 \\
P_{\text{blue}} &= C_{02}p_1 + C_{12}p_2 + C_{22}p_3 = 4p_1 + 2p_2 - 4p_3
\end{aligned}
$$

Fiano wants to minimize the maximum of $\{P_{\text{red}}, P_{\text{green}}, P_{\text{blue}}\}$. In other words, she wants to find,

$$
\text{minimize } \max \begin{cases} P_{\text{red}} &= 2p_1 + 11p_2 + 3p_3 \\ P_{\text{green}} &= 3p_1 + 6p_2 + 4p_3 \\ P_{\text{blue}} &= 4p_1 + 2p_2 - 4p_3 \end{cases}
$$

Therefore, introducing an auxiliary variable z, we obtain the following three constraints:

$$
\begin{aligned}
z &\le 2p_1 + 11p_2 + 3p_3 \\
z &\le p_1 + 6p_2 + 4p_3 \\
z &\le 4p_1 + 2p_2 - 4p_3
\end{aligned}
$$

(c) **Constraints** Hence, including the properties of p_1, p_2, p_3, the constraints of the problem are as follows:

$$\begin{aligned}
z &\leq 2p_1 + 11p_2 + 3p_3 \\
z &\leq p_1 + 6p_2 + 4p_3 \\
z &\leq 4p_1 + 2p_2 - 4p_3 \\
1 &= p_1 + p_2 + p_3 \\
0 &\leq p_1, p_2, p_3
\end{aligned}$$

Hence our linear program will use the above constraints find the minimum value of z. The above design was implemented using `scipy.linprog` and we found $p_1 = 8/9, p_2 = 0$, and $p_3 = 1/9$.

In a similar manner, we can find the optimal probabilities q_1, q_2, and q_3 that Paul will select to maximize his minimum payoff. For this scenario, we have the following:

(a) **Decision Variables** The decision variables are q_1, q_2 and q_3 that would help Paul maximize his minimum payoff.

(b) **Objective Function** Using Paul's choice of q_1, q_2, q_3, Paul's expected utility for selecting red, green, and blue are as follows:

$$\begin{aligned}
P_{\text{red}} &= C_{00}q_1 + C_{01}q_2 + C_{02}q_3 = 2q_1 + 3q_2 + 4q_3 \\
P_{\text{green}} &= C_{10}q_1 + C_{11}q_2 + C_{12}q_3 = 11q_1 + 6q_2 + 2q_3 \\
P_{\text{blue}} &= C_{20}q_1 + C_{21}q_2 + C_{22}q_3 = 3q_1 + 4q_2 - 4q_3
\end{aligned}$$

Paul wants to maximize the minimum of $\{P_{\text{red}}, P_{\text{green}}, P_{\text{blue}}\}$. In other words, he wants to find,

$$\text{maximize } \min \begin{cases} P_{\text{red}} & = 2q_1 + 3q_2 + 4q_3 \\ P_{\text{green}} & = 11q_1 + 6q_2 + 2q_3 \\ P_{\text{blue}} & = 3q_1 + 4q_2 - 4q_3 \end{cases}$$

Therefore, introducing an auxiliary variable z, we obtain the following three constraints:

$$\begin{aligned}
z &\geq 2q_1 + 3q_2 + 4q_3 \\
z &\geq 11q_1 + 6q_2 + 2q_3 \\
z &\geq 3q_1 + 4q_2 - 4q_3
\end{aligned}$$

(c) **Constraints** Hence, including the properties of q_1, q_2, q_3, the constraints of the problem are as follows:

$$\begin{aligned}
&z \geq 2q_1 + 3q_2 + 4q_3 \\
&z \geq 11q_1 + 6q_2 + 2q_3 \\
&z \geq 3q_1 + 4q_2 - 4q_3 \\
&1 = q_1 + q_2 + q_3 \\
&0 \leq q_1, q_2, q_3
\end{aligned}$$

Hence our linear program will use the above constraints find the maximum value of z. The above design was implemented using `scipy.linprog` and we found $q_1 = 0, q_2 = 8/9$, and $q_3 = 1/9$.

(3) The above design has been implemented using `scipy.linprog` in `pset2a.py`.

Fiona's final strategies are to play red, green, and blue with probabilities $p_1 = 8/9, p_2 = 0$, and $p_3 = 1/9$ respectively. With these probabilities, Paul's expected utility is 3.11.

Paul's final strategies are to play red, green, and blue with probabilities $q_1 = 0, q_2 = 8/9$, and $q_3 = 1/9$ respectively. With these probabilities, Paul's expected utility is 3.11.

Problem 3 A real world example of the Stackelberg competition is the competition between two companies, for example Ford Motor Company and Toyota, in a country where both of them are trying to dominate the market. Following are the parameters of interest:

1. 1 defender and 1 attacker (Ford and Toyota)
2. The production quantities ($s_n > 0,\ n = 1, 2$) are selected by each of them
3. Total cost of production $= c_n \cdot s_n$. (c_n is the cost of company n, where $n = 1, 2$)
4. The demand curve is dictated by the price, $P(s_1 + s_2)$
5. Profit is the difference between the price and the cost, $P(s_1 + s_2)s_n - c_n s_n$

Assume that Ford Motor Co. is the leader, who selects its production quantity in advance. Toyota chooses its quantity in response to Ford. Therefore, Toyota is the attacker and Ford is the defender.

The utilities are computed from the prices established by both automakers, their production quantities, and the cost of production.

The targets are the production quantities of the respective automakers. Since Ford is the leader, it will have higher profits due to the first mover advantage. The follower, Toyota, will have lower profits.

The resources are any available information for both companies. In the case of the above automakers, the information can be obtained from any publicly available source, such as quarterly release of the company.

The schedules correspond to the time when both companies decide on their production quantities. The leader (Ford) decides first. When this information becomes publicly available, the follower (Toyota) determines its production quantities and schedule.

The Stackelberg competition is a logical and effective application to solve the above example competition between Ford and Toyota. Both companies are trying to maximize their payoffs in a single market dictated by a fixed demand curve. The leader wants to choose its production quantity to maximize its payoff, while the follower wants to choose its production quantity to maximize its payoff. The leader (Ford) knows that the follower (Toyota) will choose its production quantity optimally in equilibrium, as the follower also wants to maximize its payoff. Therefore, both automakers in his example reach Stackelberg equilibrium. Assuming their respective strategies to be c and g, we can summarize:

- The leader (Ford) plays a best response:

$$U_{\text{Ford}}(c, g(c)) \geq U_{\text{Ford}}(c', g(c')),\ \text{for all leader mixed strategies } c'$$

- The follower (Toyota) plays a best response:

$$U_{\text{Toyota}}(c, g(c)) \geq U_{\text{Toyota}}(c', g(c')),\ \text{for all } c, g'.$$

PSET 2

Anusha Murali **CS182** **October 10, 2023**

Problem 4

(1) Angela's risk-attitude is risk averse. This is because her utility function $u_1(w) = \sqrt{w}$ is a concave function, which has a decreasing slope. (The second derivative of the utility function $u'' = -\frac{1}{4}w^{-3/2}$ is negative).

(2) The expected utility function of purchasing insurance for this month is:

$$U_{\text{insurance}} = 0.996\sqrt{w-1} + 0.004\sqrt{w-1} = \boxed{\sqrt{w-1}}\,.$$

(3) No, it does not make sense for Angela to buy this insurance for this month. The reason is as follows:

When not taking the insurance, Angela's expected utility is,

$$U_{\text{no insurance}} = 0.996\sqrt{w} + 0.004\sqrt{w-65}$$

When $w = 100$, $U_{\text{insurance}} = \sqrt{w-1} = \sqrt{99} \approx 9.94987$. When $w = 500$, $U_{\text{insurance}} = \sqrt{w-1} = \sqrt{499} \approx 22.33831$. Therefore, when taking insurance, Angela's utility is found to be in the following range:

$$9.94987 \leq U_{\text{insurance}} \leq 22.33831.$$

On the other hand, when $w = 100$, $U_{\text{no insurance}} = 0.996\sqrt{w}+0.004\sqrt{w-65} =\approx 9.98366$. When $w = 500$, $U_{\text{no insurance}} = 0.996\sqrt{w} + 0.004\sqrt{w-65} =\approx= 22.35466$. Therefore, when taking mp insurance, Angela's utility is found to be in the following range:

$$9.98366 \leq U_{\text{no insurance}} \leq 22.35466.$$

Since Angela's utility when not taking insurance is larger than that when taking insurance (for $100 < w < 500$), Angela should not buy this insurance for this month. $\square$

(4) Assume that Angela buys the insurance. In this case, if her pencil breaks, her gain will be $65 - 1 = 64$, which corresponds to a prospect utility of $0.004 \times (65-1)^{1/3}$. If the pencil does not break, her loss will be -1, which corresponds to a prospect utility of $\lambda \cdot 0.996 \times (-1)^{1/3}$. Therefore, the net prospect utility when Angela purchases the insurance is,

$$U_1 = 0.004 \times (65-1)^{1/3} + \lambda\left(0.996 \times (-1)^{1/3}\right) = 0.016 - 0.996\lambda.$$

Now, assume that Angela does not buy the insurance. In this case, if her pencil breaks, her net loss (in comparison to having purchased the insurance) will be $-65+1 = -64$. Hence the prospect utility when Angela does not purchase the insurance is,

$$U_2 = \lambda\left(0.004 \times (-65+1)^{1/3}\right) = -0.016\lambda$$

For Angela to benefit, we want $U_1 \geq U_2$. Therefore,

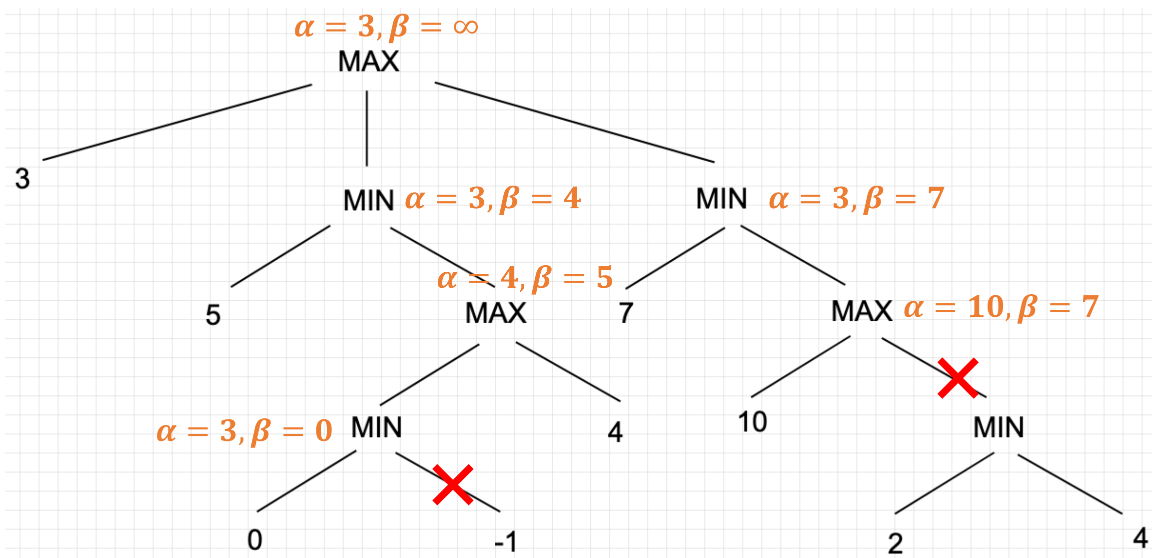
$$0.016 - 0.996\lambda \geq -0.016\lambda$$
$$\lambda \leq \frac{0.016}{0.980} = \boxed{\frac{4}{245}}.$$

Since $\lambda = \frac{4}{245} < 1$, the convexity of the curve in the third quadrant will be smaller than the convexity for $\lambda = 1$, which means Angela is justified in paying \$1/month for the insurance premium. $\square$

Problem 5

(1) I have implemented `MinimaxAgent` in `pset2b.py`

(2) (a) The (α, β) values at each node are indicated below. The subtree is pruned at the two places indicated by the red X in the diagram. We will not visit those nodes below the points marked by the red-X's.



The actions taken by Paul and Fiona are as follows:

i. Paul would want to select the maximum of the three values at the level just below the root. The left most node is 3, but the middle and right nodes have to be determined based on Fiona's and Paul's actions on the respective subtrees.

ii. We now look at the middle subtree at the level just below root. Fiona selects 0 a the bottom of the middle tree (because the node $= -1$ has been already pruned using alpha-beta pruning. Paul, being the maximizer, now selects 4 (as $4 > 0$. Fiona now selects 4 (as $4 < 5$).

iii. We now look at the right subtree below the root. The entire right-most subtree has been pruned by alpha-beta pruning. Therefore, Paul selects 10. Now Fiona, as the minimizer, has to choose between 7 and 10. She selects 7.

iv. Hence the 3 nodes just below the root node contain the values 3, 4, and 7. Since Paul (who is the maximizer) is at the root, he selects the maximum of 3, 4, and 7, which is 7. □

(b) I have implemented `AlphaBetaAgent` in `pset2b.py`

(3) When the opponent is not playing optimally, we can model her actions as a probability distribution. Specifically, we can use the Expectimax search, where the MAX nodes behave similar to that as in the MiniMax search, but the opponent's nodes are modeled by CHANCE nodes. The CHANCE nodes use the expected utilities computed for the nodes below. This idea is based on the principle that an agent should select the action that maximizes its expected utility. Therefore, the MAX player in this scenario should choose a CHANCE node with the maximum utility. Therefore, we can summarize our strategy as follows:

   (a) if a node is a MAX node, then it will return the max value of its successors.

   (b) if a node is a CHANCE node, then it will return the expected utility (or average for a uniform random agent) of its successors. If CHANCE nodes have different probabilities, then we can compute the expected utility using $\sum x_i p_i$, where x_i and p_i are the the value of the ith successor and probability of that value.

   A strategy which requires minimizing the value is analogous to the above, where the maximizer nodes are replaced by minimizer nodes.

(4) I have implemented `OptimizedAgainstRandomAgent` in `pset2b.py`

(5) The simulation results are shown in the table below. The last column shows the percentage of improvement of the first agent against the random agent.

Prefix	Optimized Agent vs Random Agent	Minimax Agent vs Random Agent	Percentage Improvement
`beh`	0.007852	-0.021973	$\approx 280\%$
`feb`	-0.005510	-0.008797	$\approx 160\%$
`gw`	0.101368	0.0554964	$\approx 183\%$

The `MinimaxAgent` assumes that it is playing against an adversary who always makes optimal decisions. This is not the case when the opponent is making her decisions randomly. On the other hand, the `OptimizedAgainstRandomAgent` chooses its actions based on the maximum expected utility of its successors. Therefore, for example, the MAX player in the `OptimizedAgainstRandomAgent` selects a CHANCE node, which has the maximum expected utility (which is based on the probability distribution of the random agent, which could be a uniform random distribution). This is why the `OptimizedAgainstRandomAgent` consistently beats `MinimaxAgent` while competing against a random agent.

PSET 2

Anusha Murali **CS182** **October 10, 2023**

Problem 6

(1) I worked on this by myself. I did not use any other resources besides the lecture slides and the textbook.

(2) I spent 35 hours (20 on the coding part, and 15 on the theory part) on this assignment.