Problem 1 (Card Sharks)

(a) There are 4 ranks of race cards with 4 possible suites to draw from and 52 cards in the deck: \( \frac{4 \times 4}{52} \).

(b) By replacing the card every time, the total pool of cards remains 52: \( \left( \frac{4 \times 4}{52} \right)^3 \).

(c) The total number of possible five card straights is given by choosing a rank that can base a straight (2–10) multiplied by the number of ways to draw the straight (note that the ranks of the cards are determined by the base) and accounting for the 4 possible suites. We finally divide by the total number of possible five-card poker hands to obtain the desired probability:

\[
\frac{\binom{9}{1} \cdot 4^5}{\binom{52}{5}}.
\]

(d) To draw a pair, we choose a rank and then a subset of the 4 possible cards drawn from that rank. We then draw 3 cards (unordered) from the remaining deck, dividing by the total number of possible hands.

\[
\frac{\binom{13}{1} \cdot \binom{4}{2} \cdot \binom{50}{3}}{\binom{52}{5}}.
\]

(e) To draw a full house, we determine the ranks of the pair and the triple and then the subset of suites drawn from those ranks:

\[
\frac{\binom{13}{1} \cdot \binom{4}{2} \cdot \binom{12}{1} \cdot \binom{4}{3}}{\binom{52}{5}}.
\]

Problem 2 (Expectation)

Throughout this problem, define the following functions:

\( A(m) \) = The payoff of match \( m \) with no power play option.

\( A(m, t) \) = The payoff of match \( m \) with power play option \( t \).

\( P(m) \) = The odds of match \( m \).

with the payoffs and odds determined by the table on the Iowa lottery website. Furthermore, define the possible matches \( M \) and power play options \( T \) as:

\[ M = \{ 5 + PB, 5, 4 + PB, 4, 3 + PB, 3, 2 + PB, 1 + PB, 0 + PB \} \]

\[ T = \{ 2, 3, 4, 5, 10 \} \]

(a) \( E[X] = \sum_{m \in M} A(m) \cdot P(m) = 0.4567647276810106 \), so for every play ($2), you are expected to earn $0.46.

(b) We want to solve for \( A(5 + PB) \) with all other numbers determined by the grid and the desired payoff to be $2. This leads to the formula:

\[ A(5 + PB) = (2 - \sum_{m \in M \setminus \{5 + PB\}} A(m) \cdot P(m))/P(5 + PB) = 6.278699405795971 \times 10^8. \]

So the jackpot must be approximately $628,000,000 for us to break even on average.
(c) The expected value of playing with the power play option (where the power play options are all equally likely) is:

\[
E[X] = \sum_{t \in T} E_t[X] \cdot \frac{1}{|T|} = \frac{1}{5}(0.542 + 0.777 + 1.245 + 1.480 + 2.651) = 1.339.
\]

To play with the power play option, you need to spend one extra dollar, a \(3/2 = 1.5\) increase in initial cost. The increase in payoff is greater at \(1.339/0.46 = 2.91\), so it is worthwhile to add the power play option (assuming that the options are equally likely which they aren’t in practice).

**Problem 3 (Faulty)**

(a) \(P(B) = 0.20\), as advertised in the table.

(b) The probability that a part is faulty is the weighted average of the faults of the individual machines:

\[
P(\text{faulty}) = 0.30 \times 0.025 + 0.20 \times 0.015 + 0.40 \times 0.025 + 0.10 \times 0.01 = 0.0215.
\]

The probability that a part is both faulty and comes from \(B\) is the product of these individual probabilities:

\[
P(B \cap \text{faulty}) = 0.20 \times 0.015 = 0.003.
\]

Thus, the final desired probability is given by:

\[
P(B \mid \text{faulty}) = \frac{P(B \cap \text{faulty})}{P(\text{faulty})} = \frac{0.003}{0.0215} = 0.1395.
\]

**Problem 4 (Probabilistic Sampling)** Clearly if \(n \leq k\), then we are able to fit the \(n\) elements of the collection into the \(k\)-sized sample set. Thus, consider the case where \(n > k\), i.e., there are more elements than space in the sample set.

**Claim 1.** After the \(i\)th iteration of the algorithm, the first \(i + k\) elements of the collection have probability \(k/(k+i)\) of being in the sample array.

**Proof.** We prove this claim by induction on the number of iterations of the algorithm, \(i\).

- \(i = 0\). When \(i = 0\), we have loaded the first \(k\) elements of the collection into the sample. These first \(k\) elements have probability \(\frac{k}{k} = \frac{k}{k} = 1\) of being in the sample.

- \(i, i > 0\). By our inductive hypothesis, the first \((i - 1) + k\) elements of the collection have probability \(k/(k + (i - 1))\) of being in the sample array. Consider one of the first \(i + k\) elements in the array: it is is either the \(i\)th element or one of the previous ones. The \(i\)th element has probability \(k/(k + i)\) of entering the array as the random number generated by the algorithm in the range \(k + i\) must hit one of the \(k\) possible indices of the array. An element that is not the \(i\)th element has probability \((k + (i - 1))/(k + i)\) of staying in the area as it is only replaced if its index is randomly chosen. Thus the probability of this element staying in the sample is:

\[
\frac{k}{k + (i - 1)} \cdot \frac{k + (i - 1)}{k + i} = \frac{k}{k + i}.
\]
Claim 2. After completion of the algorithm, the $n$ elements of the collection each have probability $k/n$ of being in the sample array.

Proof. The algorithm goes through $n - k$ iterations, so by the previous claim, after the algorithm is over, each of the $(n - k) + k = n$ elements of the collection have probability:

\[
\frac{k}{k + (n - k)} = \frac{k}{n}
\]

of being in the sample.

(Note: this algorithm, so-called algorithm R, is an example of a probabilistic technique called reservoir sampling.)