B Essay Questions (40 marks)

B.1. A family owns $n \geq 3$ consecutive houses in a row and has three children, Aisha, Bo, and Claire. The parents want to divide the houses among the children in such a way that each child receives a **non-empty** block of **consecutive** houses. Each child has a positive integer value for each house, and a child's value for his/her block is the sum of values for each house in the block. The parents want to divide the houses so that the **sum** of the three children's values for their own block is the highest possible. (10)

[[Example: Suppose $n = 4$, Aisha has value $8, 2, 8, 7$ for the houses in this order, Bo has value $1, 9, 5, 8$, and Claire has value $4, 4, 4, 4$. Then, the highest possible value is 28, achieved by giving the first house to Claire, the second house to Bo, and the third and fourth houses to Aisha.]]

Design and analyze the correctness and running time of an algorithm to output the highest possible value. You should try to optimize the asymptotic running time of the algorithm for the parents, and state this running time in terms of n using the O-notation. (However, you do not need to prove that your running time is optimal.)

Solution: We try all the $3! = 6$ permutations of children separately. Fix a permutation, and suppose that child 1 gets the left block, child 2 the middle block, and child 3 the right block.

For $i \in \{1, \ldots, n\}$ and $j \in \{1, 2, 3\}$, denote child j's value for house i by $A[i, j]$, and let $m[i, j]$ be the maximum value we can get from houses $1, 2, \ldots, i$ if we assign house i to child j. We have

$$m[i,j] = \begin{cases} A[1,1] & \text{if } i = j = 1; \\ A[i,1] + m[i-1,1] & \text{if } i > 1 \text{ and } j = 1; \\ -\infty & \text{if } i = 1 \text{ and } j > 1; \\ A[i,j] + \max\{m[i-1,j], m[i-1,j-1]\} & \text{otherwise.} \end{cases}$$

Here, if $i = j = 1$, we can only assign house 1 to child 1 (due to the non-emptiness condition) and get value $A[1, 1]$. If $i > 1$ and $j = 1$, we must assign house i to child 1, which means that houses $1, 2, \ldots, i-1$ are also assigned to child 1; the value of this is captured by $m[i-1, 1]$. If $i = 1$ and $j > 1$, the non-emptiness condition for child 1 is violated, so there is no valid assignment. Finally, if $i, j > 1$, then assigning house i to child j gives value $A[i, j]$, and we may choose to assign house $i - 1$ to child j (yielding value $m[i - 1, j]$, by a "cut-and-paste" argument) or to child $j - 1$ (yielding value $m[i - 1, j - 1]$, by a "cut-and-paste" argument). The final answer is then $m[n, 3]$.

The running time is $O(n)$. To achieve this, we fill in the table in increasing order of i and j. There are $O(n)$ entries, and computing each entry takes time $O(1)$. Trying different children permutations does not add to the asymptotic running time, since $3! = 6$ is a constant. Computing the final answer also takes time $O(1)$.

Marking scheme:

- (2 marks) Trying all children permutations, and stating that the number of permutations is 6 (or constant).

- (1 mark) Mentioning "dynamic programming" or attempting a dynamic programming solution.
- (1 mark) Indicating a feasible meaning of $m[i, j]$.
- (3 marks) Stating a correct recurrence for $m[i, j]$.
- (1 mark) Describing how to obtain the final answer.
- (2 marks) Analyzing the running time.

Remarks:

Alternative solution 1 (max 4 marks):

- For all i, j, attempt assigning Aisha houses $1, \ldots, i$, Bo $i+1, \ldots, j$, and Claire $j+1, \ldots, n$ (along with the other possible permutations). This solution runs in $O(n^3)$ time as there are $\binom{n-1}{2}$ possible partitions and each assignment takes $O(n)$ time to compute. 3 marks for this solution.
  - It is possible to speedup this to $O(n^2)$ by using prefix sums so that each assignment takes $O(1)$ time to compute. 4 marks if this speedup is used.

Alternative solution 2 ($O(n^2)$ DP, at most 7 marks):

- Let $m[i, j]$ be defined the same way as in the main solution. Let $A[x \ldots y, j]$ be a shorthand for $A[x, j] + A[x+1, j] + \cdots + A[y, j]$; we can compute this in $O(1)$ time using prefix sums. We then have

$$m[i,j] = \begin{cases} A[1 \ldots i, 1] & j = 1; \\ \max_{k \in \{1,2,\ldots,i-1\}} m[k, j-1] + A[k+1 \ldots i, j] & j > 1, \end{cases}$$

  and we return $m[n, 3]$ as in the main solution.

Alternative solution 3 (Bitmask DP, full credit):

- Let S be a subset of {Aisha, Bo, Claire}. For a house i and a child j, let $m[i, S, j]$ be the maximum value of assigning the houses $1, \ldots, i$ by assigning each child in S a non-empty consecutive block, and assigning house i to child j. Then,

$$m[i,S,j] = \begin{cases} -\infty & j \notin S; \\ A[1,j] & S = \{j\}, i = 1; \\ -\infty & |S| > 1, i = 1; \\ A[i,j] + \max(m[i-1,S,j], & \\ \qquad \max_{k \in S, k \neq j} m[i-1, S \setminus \{j\}, k]) & \text{otherwise.} \end{cases}$$

  Return $\max_{j \in \{1,2,3\}} m[n, \{1,2,3\}, j]$.
- If the current child (i.e., the child who receives house i) is not part of the state, at most 6 marks are given.

- In this solution, it is important to keep track of the set of children who have been assigned a house, not just the number of such children.

Wrong solution:

- Let $m[i]$ be the maximum value obtained by assigning the first i houses. Then,

$$m[i] = \begin{cases} 0 & i = 0; \\ m[i-1] + \max(A[i,1], A[i,2], A[i,3]) & 1 \leq i \leq n. \end{cases}$$

- This is wrong as it does not keep track of who has already been given houses, and therefore effectively a greedy solution that ignores the consecutiveness condition.
- Even if you keep track of the number of children who have been given houses, it is not sufficient.

B.2. There are four points on a circle, labeled $0, 1, 2, 3$ in clockwise order. Sue starts at point 0, which is already counted as visited. At each time step, she jumps to either the adjacent point clockwise or the adjacent point counterclockwise with probability $1/2$ each. She continues doing this until she has visited all four points. [[Example: A possible sequence is $0 \to 1 \to 2 \to 1 \to 0 \to 3$.]] (10)

What is the probability that point 2 is the **last** point that Sue visits among the four points?

Solution: Let x be the probability that Sue visits point 2 as the last point, given that she has visited only points 0 and 1 and is currently at point 0. Similarly, let y be the probability that she visits point 2 as the last point, given that she has visited only points 0 and 1 and is currently at point 1.

Consider the state where she is at point 0 and has visited only points 0 and 1. With probability $1/2$, she jumps to point 3, and point 2 is the last point that she visits. With probability $1/2$, she jumps to point 1, and she arrives at the state where she is at point 1 and has visited only points 0 and 1. Hence, $x = 1/2 + y/2$.

Next, consider the state where she is at point 1 and has visited only points 0 and 1. With probability $1/2$, she jumps to point 2, and point 3 is the last point that she visits; in particular, point 2 is *not* the last such point. With probability $1/2$, she jumps to point 0, and she arrives at the state where she is at point 0 and has visited only points 0 and 1. Hence, $y = x/2$. Plugging this into $x = 1/2 + y/2$ and solving, we get $x = 2/3$ and $y = 1/3$.

Finally, consider the initial state where she is at point 0 and has not visited any other point. With probability $1/2$, she jumps to point 1, and she arrives at the state where she is at point 1 and has visited only points 0 and 1—we know that from this state, the probability that she visits point 2 as the last point is $y = 1/3$. With probability $1/2$, she jumps to point 3; by symmetry, from this state, the probability that she visits point 2 as the last point is also $1/3$. Hence, the overall probability that she visits point 2 as the last point is $(1/2)(1/3) + (1/2)(1/3) = 1/3$.

Alternative Solution:

We prove the more general statement that if the points are $0, 1, \ldots, n$, then the probability that each point other than 0 is the last visited point is $1/n$. Take any point $i \neq 0$, and consider the first moment where a point j adjacent to i is visited. Note that i will be the last visited point if and only if Sue visits every other point *at or after this moment* before visiting i. This probability is the same no matter which point i is. Since the probabilities must sum to 1, the corresponding probability for each point $i \in \{1, \ldots, n\}$ is $1/n$. In particular, for $n = 3$ it is $1/3$.

Remarks: Besides the possible solutions above, many students also gave solutions along the following lines:

1. Calculate the probability that 2 is not the last point visited. Suppose without loss of generality that 1 is the first point visited from 0. Then, for 2 not to be the last point visited, the sequence of visits should be $0, 1, (0, 1)^*, 2 \ldots$, where * denotes any (possibly zero) number of repetition. This has probability $\frac{1}{2} \cdot (1 + \frac{1}{4} + \frac{1}{16} + \cdots) \cdot \frac{1}{2} = \frac{1}{4} \cdot \frac{1}{1-\frac{1}{4}} = \frac{1}{3}$. Similarly, if 3 is the first point visited from 0, then the probability of 2 not being the last point visited is $1/3$. Taken together, the probability of 2 not being the last point visited is $2/3$, and thus the probability of 2 being the last point visited is $1/3$.

2. Calculate the probability that 2 is the last point visited as follows. Without loss of generality, assume the first point visited from 0 is 1. Then, for 2 to be the last point visited, 3 must be visited before 2 starting from 1. For this to happen, we must visit 0, and then a finite number of times $(1, 0)$, followed by 3. This has probability $\frac{1}{4} \cdot (1 + \frac{1}{4} + \frac{1}{4^2} + \cdots) = \frac{1}{3}$.

These solutions are acceptable.

Marking scheme: For the main solution:

- Any correct proof and answer receives 10 marks. If students only compute the probability of 2 *not* being the last point visited correctly (without computing the probability of 2 being the last point visited), or do not assume without loss of generality the first visit to 1 or 3, then they lose 1 mark.
  Minor errors in the above proof or computation result in a deduction of 1 or 2 marks.

- Obtaining correct recurrence relations either in the main solution, or explaining the setting as in the solutions in the remarks, and setting the correct equation and explanation, but some minor errors in the calculations or minor incomplete proof details: 7 marks

- Has the idea of repeating visits to point 1 or 3 as explained above, and sets up almost correct recurrence relations but makes errors in the recurrence relations, or proofs lacking details: 5 marks

- Some reasonable attempt to solve, where the student realizes symmetry, some recurrence, the fact that point 2 may come in the future (rather than almost immediately), etc: 3 marks.
  Some attempts (without much progress) may receive 1 or 2 marks.

- Some students tried to enumerate the sequences which have 2 as the last visited point, but didn't take care of all such possibilities properly. Depending on what they did, they received 3–5 marks.
- Does only the first few steps of Sue and calculates probability based on it: 2 marks
- Some students just claim that the probability of each of $1, 2, 3$ being last point visited is equal, and thus the answer is $1/3$. Without proof of the claim, this doesn't receive much credit—they receive 3 marks. When some reasonable argument is given (but not correct), they receive 4 marks.

Note that in some cases, further marks may be deducted if the notations are not clear and/or the solution provided lacks explanation.

Other answers which are different from above would be graded based on similar general rules.

B.3. In the country Wonderland, there are 1-cent, 5-cent, 10-cent, and 25-cent coins (and no other types of coins). Your friend Alice from Wonderland claims that for every positive integer n, she can make n cents using the **smallest** number of coins via the following algorithm: (10)

"Start with an empty set of coins. Take the largest number $r \in \{1, 5, 10, 25\}$ that does not exceed n, add one r-cent coin to the set, and replace n by $n - r$. Repeat this procedure until n becomes 0, and output the final set of coins." [[Example: If $n = 38$, Alice outputs $25 + 10 + 1 + 1 + 1$.]]

Is Alice's claim correct? Justify your answer.

Solution: Alice's claim is correct – By the greedy property (proof below) combined with the optimal substructure property (not required, as this was already proven in lecture), we conclude that the greedy algorithm works.

We first observe that any optimal set of coins contains

- at most $4 \times$ 1-cent coins ($5 \times$ 1-cent $\rightarrow 1 \times$ 5-cent);
- at most $1 \times$ 5-cent coin ($2 \times$ 5-cent $\rightarrow 1 \times$ 10-cent);
- at most $2 \times$ 10-cent coins ($3 \times$ 10-cent $\rightarrow 1 \times$ 25-cent $+ 1 \times$ 5-cent);
- never $2 \times$ 10-cent and $1 \times$ 5-cent together ($2 \times$ 10-cent $+1 \times$ 5-cent $\rightarrow 1 \times$ 25-cent).

With these observations, we proove the greedy property by induction on n.

Base case. If $0 \leq n < 5$, the only coin available is the 1-cent coin, so the greedy solution uses n coins of 1-cent, which is clearly optimal.

Inductive Step: Assume that for all amounts $m < k$, the greedy algorithm produces an optimal solution.

Let c be the largest coin in $\{1, 5, 10, 25\}$ with $c \leq k$. The greedy algorithm picks one coin of value c and then makes change for $m = k - c$ by the inductive hypothesis, using

$$1 + \text{CHANGE}(k - c)$$

coins, where CHANGE(m) is the minimum number of coins for amount m.

We prove that any optimal solution for k *must* include at least one coin of value c. If not, then an optimal solution makes k entirely with smaller denominations, which requires strictly more coins than a single c-coin. This contradicts optimality.

1. **Case** $5 \leq k \leq 9$ **(so** $c = 5$**).** Omitting a 5-cent coin forces the use of at least five 1-cent coins; replacing those five by one 5-cent coin saves four coins, contradicting optimality.
2. **Case** $10 \leq k \leq 24$ **(so** $c = 10$**).** Without a 10-cent coin, one needs at least two smaller coins to reach 10-cent; replacing two 5-cent coins by a single 10-cent coin saves one coin, contradicting optimality.
3. **Case** $k \geq 25$ **(so** $c = 25$**).** Without a 25-cent coin, the minimum coins to reach *at least* 25 cents is *at least* 3 coins. Consider 3 maximum-value coins, yielding two cases:
   (a) Consider three 10-cents; replacing with one 25-cent and one 5-cent saves one coin.
   (b) Consider two 10- and one 5-cent coins; replacing with one 25-cent saves two coins.

   Hence any such solution uses ≥ 3 coins to cover 25-cents, and swapping coins out as above for a composition with 25-cent coin yields a shorter decomposition, contradicting optimality.

In every case, omitting the c-cent coin forces the use of *more than one* coin for at least the value of c cents, so an optimal solution must contain at least one c-cent coin. Removing it yields an optimal solution for $k - c$. By induction the equality holds for all $n \geq 1$, proving the greedy property.

Marking scheme:

For the full-credit rubric:

- (2 marks) Proving that if $5 \leq k \leq 9$, any optimal solution must contain a 5-cent coin, using observation ($5 \times$ 1-cent $\rightarrow 1 \times$ 5-cent).
- (2 marks) Proving that if $10 \leq k \leq 24$, any optimal solution must contain a 10-cent coin, using observation ($2 \times$ 5-cent $\rightarrow 1 \times$ 10-cent).
- (5 marks) Proving that if $k \geq 25$, any optimal solution must contain a 25-cent coin.
  - (2 marks) Prove using observation ($3 \times$ 10-cent $\rightarrow 1 \times$ 25-cent $+ 1 \times$ 5-cent).
  - (2 marks) Prove using observation ($2 \times$ 10-cent $+ 1 \times$ 5-cent $\rightarrow 1 \times$ 25-cent).
  - (1 mark) Using 25-cent coin in either case produces a shorter decomposition.
- (1 mark) Concluding with the greedy property and the optimal substructure property.

Otherwise, where there is no proof (or severely flawed), we apply the reduced-credit rubric:

- (4 mark) For the 4 observations, 1 mark for each observation.
- (1 mark) Proof for optimal substructure.

- (1 mark) Concluding with the greedy property and the optimal substructure property.

Remarks. Almost every student attempted this question, and the submitted proofs covered a wide spectrum of ideas. When marking, we supplied counter-examples whenever possible in order to test each argument. The most frequent weaknesses (without considering the specific dynamics) were:

- *Focusing only on a common factor.* Pointing out that all coins are multiples of 5 does not, by itself, guarantee optimality. For instance, the set $\{1, 2, 6, 8\}$ consists of multiples of 2, yet greedy fails for the value 12: $6 + 6$ (2 coins, optimal) versus the greedy choice $8 + 2 + 2$ (3 coins).
- *Appealing only to coin decomposability.* Showing that the largest coin can be expressed as a combination of smaller coins is insufficient. The set $\{1, 2, 6, 8\}$ adheres to coin decomposability, but greedy still fails for 12: $6 + 6$ versus $8 + 2 + 2$.
- *Arguing at too high a level of abstraction.* Several proofs stated general principles without examining the concrete ranges of n relevant to this specific coin set. A correct proof must analyse the actual cases (e.g. $5 \le n \le 9$, $10 \le n \le 24$, etc.) and argue for each case.

We assessed student solutions using one of two mutually exclusive rubrics. If the key observations were integrated into a coherent proof, we applied the standard, full-credit rubric. If the observations appeared only as standalone remarks or if the proof is severely flawed, we used a reduced-credit rubric that rewards insight but not its integration. Each script was marked under exactly one rubric; scores from the two rubrics were never combined.

Demonstrating the optimal-substructure property was optional – it carried no credit under the full-credit rubric and was worth only one mark in the reduced-credit rubric, yet most students proved it correctly. A small number of students proved optimal-substructure property without proving greedy property.

B.4. Don is organizing a sports camp for d days, with s students taking part in the camp. On each day, he needs to decide whether to let the students play tennis or volleyball. Because he has limited budget, all students must play the **same** sport on any given day. Each student has a preference on each day whether he/she is happy to play tennis, volleyball, both, or neither (a student is allowed to have different preferences on different days). The SportScheduling problem asks whether it is possible for Don to schedule the sports so that each student is happy on **at least one** day (different students are allowed to be happy on different days). (10)

[[Example: Suppose that $s = 3$, $d = 2$, the first student prefers (tennis, tennis), the second student prefers (neither, volleyball), and the third student prefers (tennis, both). This instance is a YES-instance, since Don can schedule tennis on the first day, which makes the first and third students happy, and volleyball on the second day, which makes the second student happy.]]

Prove that SportScheduling is NP-complete.

(You may assume, without proof, that the 3-SAT problem is NP-complete: Given c clauses and v variables $x_1, x_2, \ldots, x_v$, where each clause is a disjunction of three literals, such as $x_1 \vee \overline{x_3} \vee x_6$, decide whether there exists a truth assignment to the variables such that every clause is satisfied. You are also allowed to assume that no clause contains both a variable x_i and its negation $\overline{x_i}$.)

Solution: First, observe that SportScheduling belongs to NP: Given a sports schedule, it is possible to decide in time $O(sd)$ whether the schedule makes each student happy on at least one day.

Next, we show that SportScheduling is NP-hard by reducing from 3-SAT. Consider an instance I of 3-SAT with c clauses and v variables $x_1, x_2, \ldots, x_v$, such that no clause contains both a variable x_i and its negation $\overline{x_i}$. We create an instance I' of SportScheduling with c students and v days. For each $i \in \{1, 2, \ldots, c\}$ and $j \in \{1, 2, \ldots, v\}$, on day j, we let student i prefer only tennis if x_j appears in clause i, prefer only volleyball if the negation $\overline{x_j}$ appears in clause i, and prefer neither sport if neither x_j nor $\overline{x_j}$ appears in clause i. Clearly, this reduction can be done in polynomial time.

($\Rightarrow$) If I is a YES-instance of 3-SAT, there exists a truth assignment that satisfies every clause. For each $j \in \{1, 2, \ldots, v\}$, on day j, we let the students play tennis if x_j is set to True, and volleyball if x_j is set to False. Consider any student $i \in \{1, 2, \ldots, c\}$. Since clause i is satisfied, either it contains some x_j which is set to True, or it contains some $\overline{x_j}$ and x_j is set to False. In the first case, student i prefers tennis on day j and we choose tennis on that day, so the student is happy on day j. In the second case, student i prefers volleyball on day j and we choose volleyball on that day, so the student is again happy on day j. Hence, I' is a YES-instance of SportScheduling.

($\Leftarrow$) If I' is a YES-instance of SportScheduling, there exists a sports schedule that makes each student happy on at least one day. For each $j \in \{1, 2, \ldots, v\}$, we set x_j to True if the schedule chooses tennis on day j, and False if it chooses volleyball on that day. Consider any clause $i \in \{1, 2, \ldots, c\}$. Since student i is happy on at least one day, either she prefers tennis on some day j and the schedule chooses tennis on that day, or she prefers volleyball on some day j and the schedule chooses volleyball on that day. In the first case, x_j appears in clause i and we set x_j to True, so the clause is satisfied. In the second case, $\overline{x_j}$ appears in clause i and we set x_j to False, so the clause is again satisfied. Hence, I is a YES-instance of 3-SAT.

Marking scheme:

- (2 marks) Showing membership in NP.
- (3 marks) Describing a feasible reduction.
- (1 mark) Proving that the reduction can be done in polynomial time.
- (2 marks) Proving that if I is a YES-instance of 3-SAT, then I' is a YES-instance of SportScheduling.
- (2 marks) Proving that if I' is a YES-instance of SportScheduling, then I is a YES-instance of 3-SAT.

Remarks: Some common errors are listed below. Solutions with any of these errors are graded based on how close they are to the correct solution.

- Some students did the reduction in the wrong direction (i.e., from SportScheduling to 3-SAT), or stated the correct direction but effectively tried to reduce in the opposite direction.
- Several students did not correctly map elements of a 3-SAT instance to elements of a SportScheduling instance (i.e., a clause to a student, and a variable to a day). The correct mapping can be deduced by observing that the condition "each student is happy on at least one day" is analogous to "each clause contains at least one satisfied variable". Some students mapped a variable to a sport, which does not make sense as there are only two sports. Some students set the number of days to 3, with the reason being that a 3-SAT instance contains 3 literals in each clause; this does not work.
- Several students with the correct mapping (i.e., a clause to a student, and a variable to a day) did not properly set the preferences of the students. Many of these students stated correctly that a student i prefers only tennis on day j if x_j appears in clause i and prefers only volleyball if $\overline{x_j}$ appears in clause i, but either did not state the preference when neither x_j nor $\overline{x_j}$ appears in clause i, or did so incorrectly (e.g., by saying that the preference is "both", or that it can be either "both" or "neither").
- Some students with a correct reduction either did not make the claim that the reduction can be done in polynomial time, or did not prove membership in NP.