CS60020 : Foundations of Algorithm Design and Machine Learning (Spring 2026)

Class Test 1

28-Jan-2026 (Wednesday) | Maximum Marks: 30 | 06:30pm – 07:45pm

[*Solutions to all questions.*]

Q1. [Algorithm Complexity]

Prove the following: If $f(n) = O(g(n))$ and $f(n) = \Omega(g(n))$, then $f(n) = \Theta(g(n))$. **(3)**

Solution:

$f(n) = O(g(n))$ means $f(n) \leq c_1 g(n)$ (for all $n > n_0$ and for some $c_1 > 0$)

$f(n) = \Omega(g(n))$ means $f(n) \geq c_2 g(n)$ (for all $n > n_1$ and for some $c_2 > 0$)

$\Longrightarrow$ $f(n) = \Theta(g(n))$ by definition, because $c_2 g(n) \leq f(n) \leq c_1 g(n)$ (for all $n > \max\{n_0, n_1\}$)

Q2. [Algorithm Design]

Consider the problem rod cutting, where given a rod of length R, and a set S of n pieces which form the required items that are needed to be cut (each piece less than R), you need to find a subset of S of pieces which can be cut from the rod R such that the remaining unusable portion of the rod is minimized. Please note that each cut also creates a wastage of size p. Therefore, we need to minimize the total loss comprising of the remaining unused portion and the wastage due to cuts.

Answer the following questions.

(a) Present a recursive definition to solve the problem. Clearly define the arguments, the return values, base condition, recursive calls and final solution formation. Explain each of the steps. **(8)**

Solution:

This problem is similar in spirit to the *knapsack* problem and can be solved using a recursive inclusion–exclusion strategy.

Definitions and Parameters

Let:

- $C = \{c_1, c_2, \ldots, c_n\}$ be the initial set of all available pieces.
- S be the subset of pieces selected so far.
- T be the remaining set of pieces yet to be considered.
- L be the remaining usable length of the rod.
- W be the accumulated wastage due to cuts.
- k be the number of remaining pieces in T.
- p be the wastage incurred per cut.

The recursive procedure is initially called as: $\textsc{Pieces}(\emptyset, C, R, 0, n)$

The function returns a pair $\langle P, D \rangle$ where:

- P is the selected subset of pieces.
- D is the total loss (unused rod length plus cut wastage).

Recursive Definition

Algorithm 1 : Recursive Piece Selection with Wastage

```
function PIECES(S, T, L, W, k)
    if k = 0 or L = 0 then
        // Base case: no pieces or no rod length left
        // Total loss = unused rod length + accumulated wastage
        return ⟨S, L + W⟩
    end if
    c ← first element of T
    // Select the next piece to consider
    ⟨P1, D1⟩ ← ⟨∅, ∞⟩
    // Initialize inclusion case with worst possible loss
    if c = L then
        // Case 1: piece fits exactly, no cut required
        ⟨P1, D1⟩ ← PIECES(S ∪ {c}, T \ {c}, L − c, W, k − 1)
    else if c + p ≤ L then
        // Case 2: piece obtained by cutting, wastage incurred
        ⟨P1, D1⟩ ← PIECES(S ∪ {c}, T \ {c}, L − c − p, W + p, k − 1)
    end if
    // Case 3: exclude the current piece
    ⟨P2, D2⟩ ← PIECES(S, T \ {c}, L, W, k − 1)
    if D1 ≤ D2 then
        // Choose the option with smaller total loss
        return ⟨P1, D1⟩
    else
        return ⟨P2, D2⟩
    end if
end function
```

Explanation of Steps

- The algorithm recursively explores inclusion and exclusion of each piece.
- If a piece fits exactly, it is selected without any wastage.
- Otherwise, the piece is selected only if a cut can be made, incurring wastage p.
- The exclusion branch skips the current piece.
- Among all possibilities, the solution with minimum total loss is chosen.

Final Solution

The final solution is obtained from the initial recursive call: $\textsc{Pieces}(\emptyset, C, R, 0, n)$, which returns the subset of pieces that minimizes the total loss.

(b) Show the working of your approach using a non-trivial example on a set S having at least 8 items. **(4)**

Solution:

Working of the Recursive Approach

We illustrate the working of the recursive algorithm using the following non-trivial example.

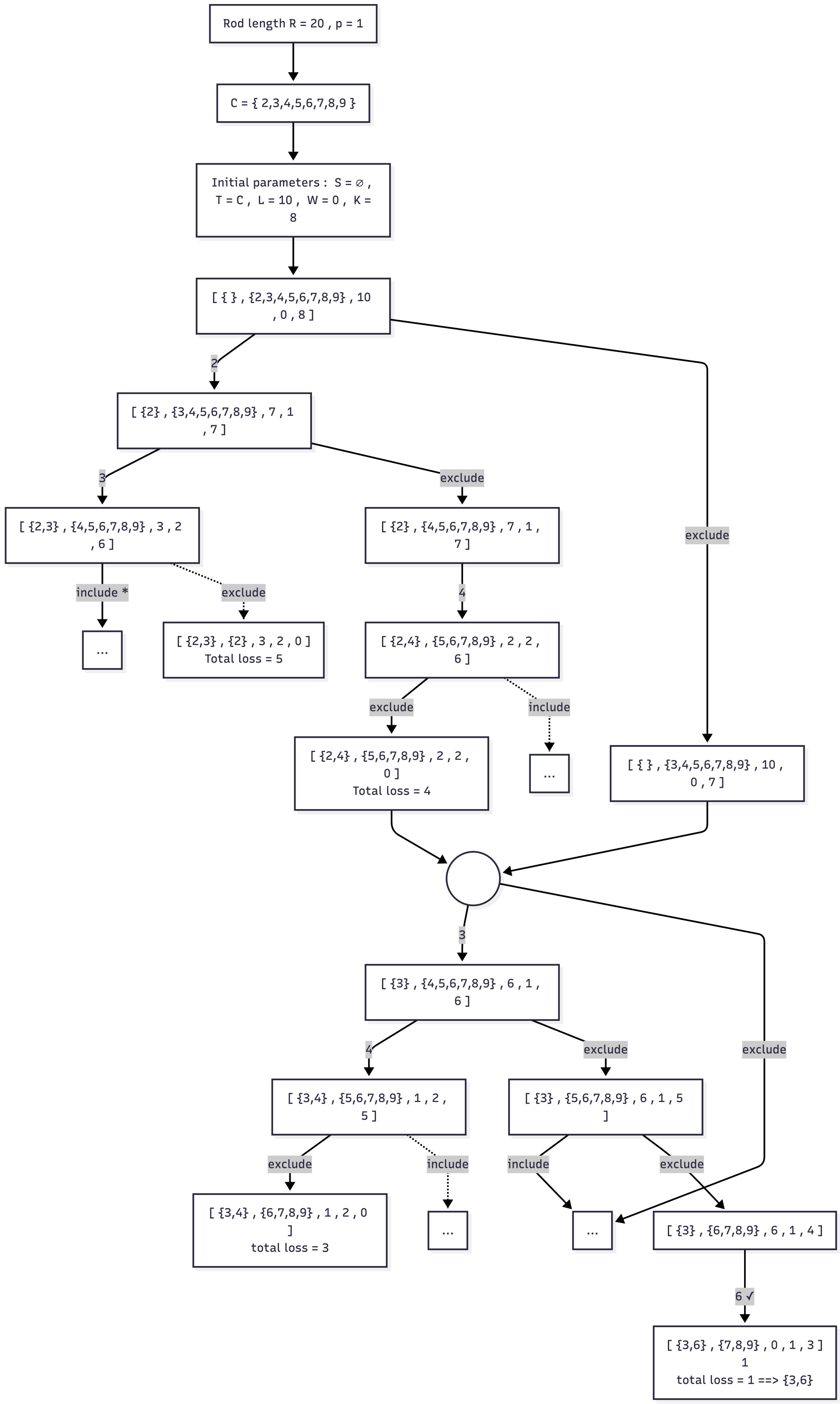
Let: $R = 20, \quad p = 1, \quad C = \{2,3,4,5,6,7,8,9\}$

The initial recursive call is: $\textsc{Pieces}(\emptyset, C, L = 10, W = 0, k = 8)$

The recursion proceeds by considering each piece using an inclusion–exclusion strategy. At every step, the algorithm decides whether to include the current piece (with or without cut wastage) or exclude it. The total loss is computed only at the base case.

Recursive Execution Tree

The complete working of the algorithm for this example is shown below as a recursion tree. One branch is expanded fully until the base case, while other branches are abbreviated for clarity.



Observations

The recursion tree demonstrates how the algorithm explores different subsets of pieces using inclusion and exclusion. Each leaf node corresponds to a base case where the total loss (unused rod length plus cut wastage) is computed. The algorithm finally selects the subset of pieces that yields the minimum total loss.

Q3. [Decision-Tree Learning]

Consider the following data set containing the information for 10 employees of IIT Kharagpur for a binary classification problem about their mode of travel to office. The Travel-Mode (by cycle or by car) has been collected along with their Designation (faculty or staff) and Residence (in campus or outside). For example, the first row of the following table is interpreted as – two *Faculty* members staying in *Campus* travel to their office via *Cycle*. Answer the following questions.

	(Attributes)		(Outcome)
# Instances	**Designation**	**Residence**	**Travel-Mode**
2	Faculty	Campus	Cycle
1	Faculty	Campus	Car
1	Faculty	Outside	Cycle
3	Faculty	Outside	Car
3	Staff	Campus	Cycle

(a) Calculate the overall entropy and Gini-index for this dataset (before any splitting). **(2)**

Solution:

The overall entropy before splitting is:

$$E_{original} = -\frac{6}{10}\log\left(\frac{6}{10}\right) - \frac{4}{10}\log\left(\frac{4}{10}\right) = 0.9710$$

The overall Gini-index before splitting is:

$$G_{original} = 1 - \left(\frac{6}{10}\right)^2 - \left(\frac{4}{10}\right)^2 = 0.4800$$

(b) Calculate the information gain when splitting on attributes, Designation and Residence. Which attribute would the decision tree algorithm choose to split? Show the calculations. **(4)**

Solution:

The information gain after splitting on the attribute Designation is:

$$E_{\text{Designation}=Faculty} = -\frac{3}{7}\log\left(\frac{3}{7}\right) - \frac{4}{7}\log\left(\frac{4}{7}\right) = 0.9852$$

$$E_{\text{Designation}=Staff} = -\frac{3}{3}\log\left(\frac{3}{3}\right) - \frac{0}{3}\log\left(\frac{0}{3}\right) = 0$$

$$\therefore \quad \text{Information Gain}|_{\text{Designation}} = E_{original} - \frac{7}{10}E_{\text{Designation}=Faculty} - \frac{3}{10}E_{\text{Designation}=Staff} = 0.2814$$

The information gain after splitting on the attribute Residence is:

$$E_{\text{Residence}=Campus} = -\frac{5}{6}\log\left(\frac{5}{6}\right) - \frac{1}{6}\log\left(\frac{1}{6}\right) = 0.6500$$

$$E_{\text{Residence}=Outside} = -\frac{1}{4}\log\left(\frac{1}{4}\right) - \frac{3}{4}\log\left(\frac{3}{4}\right) = 0.8113$$

$$\therefore \quad \text{Information Gain}|_{\text{Residence}} = E_{original} - \frac{6}{10}E_{\text{Residence}=Campus} - \frac{4}{10}E_{\text{Residence}=Outside} = 0.2565$$

Therefore, the attribute Designation will be chosen to split by the decision tree algorithm.

(c) Calculate the Gini-index gain when splitting on attributes, Designation and Residence. Which attribute would the decision tree algorithm choose to split? Show the calculations. **(4)**

Solution:

The Gini-index gain after splitting on the attribute Designation is:

$$\begin{aligned}
G_{\text{Designation}=Faculty} &= 1-\left(\frac{3}{7}\right)^2-\left(\frac{4}{7}\right)^2 = 0.4898 \\
G_{\text{Designation}=Staff} &= 1-\left(\frac{3}{3}\right)^2-\left(\frac{0}{3}\right)^2 = 0 \\
\therefore \quad \text{Information Gain}|_{\text{Designation}} &= G_{original} - \frac{7}{10}G_{\text{Designation}=Faculty} - \frac{3}{10}G_{\text{Designation}=Staff} \\
&= 0.1371
\end{aligned}$$

The Gini-index gain after splitting on the attribute Residence is:

$$\begin{aligned}
G_{\text{Residence}=Campus} &= 1-\left(\frac{5}{6}\right)^2-\left(\frac{1}{6}\right)^2 = 0.2778 \\
G_{\text{Residence}=Outside} &= 1-\left(\frac{1}{4}\right)^2-\left(\frac{3}{4}\right)^2 = 0.3750 \\
\therefore \quad \text{Information Gain}|_{\text{Residence}} &= G_{original} - \frac{6}{10}G_{\text{Residence}=Campus} - \frac{4}{10}G_{\text{Residence}=Outside} \\
&= 0.1633
\end{aligned}$$

Therefore, the attribute Residence will be chosen to split by the decision tree algorithm.

Q4. [Bayesian Learning]

Suppose $x_1, x_2, \ldots, x_n$ denote a set of random i.i.d. samples drawn from a Poisson distribution with mean $\lambda > 0$. The probability mass function of a Poisson is given as, $p(x \mid \lambda) = \frac{e^{-\lambda}\lambda^x}{x!}$. Derive the maximum likelihood estimator $(\hat{\lambda})$ for λ. **(5)**

Solution:

The likelihood function is expressed as,

$$L(x_1, x_2, \ldots, x_n \mid \lambda) = \prod_{k=1}^{n} p(x_k \mid \lambda) = \frac{e^{-n\lambda}\lambda^{\left(\sum_{k=1}^{n} x_k\right)}}{\prod_{k=1}^{n}(x_k!)}$$

The log likelihood function is expressed as,

$$\ln L(x_1, x_2, \ldots, x_n \mid \lambda) = -n\lambda + \left(\sum_{k=1}^{n} x_k\right)\ln\lambda - \sum_{k=1}^{n}\ln(x_k!)$$

To find the maximum log likelihood, we proceed as follows:

$$\begin{aligned}
\frac{\partial}{\partial\lambda}\ln L(x_1, x_2, \ldots, x_n \mid \lambda) &= 0 \\
\implies \quad -n + \frac{1}{\lambda}\left(\sum_{k=1}^{n} x_k\right) &= 0
\end{aligned}$$

Therefore, the Maximum Likelihood Estimator for λ is,

$$\hat{\lambda} = \frac{1}{n}\left(\sum_{k=1}^{n} x_k\right)$$