

CS 721: Advanced Algorithms & Analysis

Homework 4, Fall 2018, Total 50 points

Assigned on: Tuesday, 10/23/2018

Due on: Tuesday, 10/30/2018

1. (20 points) **Edit distance:** Given two strings $x[1, \dots, m]$ and $y[1, \dots, n]$ of length m and n respectively, a natural way of measuring distance between them is the extent to which they can be aligned or matched up. Edit distance attempts to measure this alignment by applying minimum number of edits to the first string x so that after the edits x is transformed to y . Edits are applied on x in the form of insertion, deletion and substitution. For example, one possible way of aligning the string $x = \text{SNOWY}$ with string $y = \text{SUNNY}$ is writing them as $S * \text{NOWY}$ and $\text{SUNN} * Y$. Now in position 2 we can insert U , in position 4 we can substitute O with N and position 5 we can delete W . These edit operations will transform the first string to the second string. If cost of each of the insertion, deletion and substitution operation is 1 then the edit distance is 3. You will write a dynamic programming solution to find edit distance automatically. Let $D(i, j)$ be the solution of the subproblem of finding edit distance between string $x[1, \dots, i]$ and $y[1, \dots, j]$. Then our goal is to find $D(m, n)$. First write $D(i, j)$ in terms of solution of smaller subproblem (optimal substructure) and give a dynamic programming solution to find minimum edit distance between $x[1, \dots, m]$ and $y[1, \dots, n]$. What is the running time of your solution?
2. (10 points) **Knapsack with repetition:** During a robbery, a burglar finds more loot than he expected and he has to decide what to take. His bag ("knapsack") can hold a total weight of at most W pounds. There are n items to choose from, of weight w_1, \dots, w_n of dollar value v_1, \dots, v_n respectively. What is the maximum dollar value of items he can fit into his bag? Denote by $K(w)$, the maximum dollar value items(s) of weight at most w that can be fit into the knapsack. Our goal is to find $K(W)$. First write $K(w)$ in terms of solution of smaller subproblem (optimal substructure) and give a dynamic programming solution to find $K(W)$. What is the running time of your solution?
3. (20 points) Suppose you are managing construction of billboards on the 21st St. N. which runs from east to west on a straight line. There are n possible sites for billboard construction given in the array $x[1, \dots, n]$, where $0 \leq x[1] < x[2] < \dots < x[n]$ specifies the distance of each possible billboard location from the west side end of 21st St. N. There is also an array $p[1, \dots, n]$ which contains the payment information, i.e., if you place a billboard at location $x[j]$ you receive payment $p[j]$.

Restrictions imposed by the Sedgwick county requires that any pair of billboard must be more than 3 miles apart. You would like place the billboard at a subset of sites so as to maximize your revenue, subject to Sedgwick county's placement restriction. For example, if $n = 4$, $x = [3, 4, 8, 9]$ and $p = [5, 6, 5, 1]$ then optimal solution will place billboards at $x[2]$ and $x[3]$ with revenue $p[2] + p[3] = 6 + 5 = 11$.

Suppose you are also given the array $prev[1, \dots, n]$ where $prev[j]$ stores the location index of the previous billboard site (to the west of $x[j]$) that satisfies Sedgwick county's billboard restriction, i.e., $prev[j] = \max\{i : i < j \text{ and } x[i] < x[j] - 3\}$. Let $R(j)$ denote the optimal revenue obtained by placing billboards at a subset of locations $x[1], x[2], \dots, x[j]$ satisfying Sedgwick county's restriction. Then our goal is to find $R(n)$. First, write $R(j)$ in terms of solution of smaller subproblem (optimal substructure) and give a dynamic programming solution to find $R(n)$ that takes input $x[1, \dots, n], p[1, \dots, n]$ and $prev[1, \dots, n]$. What is the running time of your solution? For convenience assume $prev[1] = 0$ and $R(0) = 0$

Submission:

- All texts and diagrams must be electronically produced.
- Your name and page number should appear on each page.
- Entire assignment should be a single PDF file.
- the PDF file should be named in the format HW04_Lastname_Firstname.pdf, for example HW04_Sinha_Kaushik.pdf.
- Submit the pdf file on blackboard.
- **This homework assignment is due at 11:59 pm on the due date.**