1. **Longest Common Subsequence.** Given 2 sequences $X = (x_1, \ldots, x_m)$ and $Y = (y_1, \ldots, y_n)$, find a common subsequence whose length is maximum. The subsequence need not be consecutive but must be in order. For example, the LCS of "springtime" and "printing" is "printi" with length 6.

s	p	r	i	n	g	t	i	m	e
	|	|	|	|			|	|		
	p	r	i	n		t	i	n	g

2. **Weighted Interval Scheduling.** There are n jobs. Job i starts at startTime[i], finishes at endTime[i] and has a profit of profit[i]. Two jobs are *compatible* if they don't overlap. Our goal is to find maximum profit of mutually compatible jobs. For example, Figure 1 has 5 jobs $\{a, b, c, d, e\}$, starting at $\{1, 2, 3, 4, 6\}$, ending at $\{3, 5, 9, 6, 9\}$ with profits $\{20, 20, 100, 70, 60\}$. The maximum profit that can be achieved is 150 when choosing $\{a, d, e\}$.

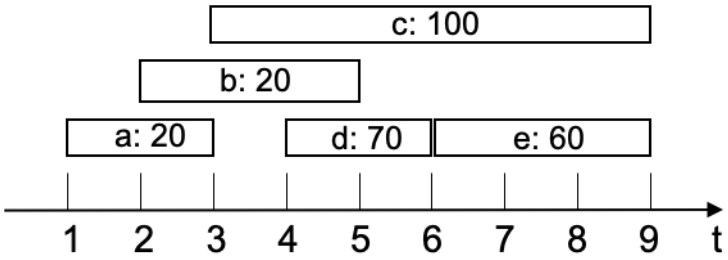

Figure 1

3. **Knapsack Problem.** Given n objects and a "knapsack". Item i weighs $w_i > 0$ kilograms and has value $v_i > 0$. Knapsack has capacity of W kilograms. Our goal is to fill knapsack so as to maximize total value. For example, 5 items are listed below and the capacity of our knapsack is 11. The optimal set is $\{3, 4\}$ with value 40.

#item	value	weight
1	1	1
2	6	2
3	18	5
4	22	6
5	28	7

Note: *main.cpp* is given by us, you can implement and test your code in it.