1.

a) Let $\mathrm{LSP}(s,t) = s'$ denote the vertex directly connected to s on the longest simple path from s to t.

Let $w(s,s')$ where $(s,s') \in E$, denote the weight put on the edge.

Let $W(s,t)$ denote the total weight of the longeest simple path from s to t.

Then we have $\mathrm{LSP}(s,t) = \mathrm{argmax}_{\{s' | (s,s') \in E\}}(w(s,s') + W(s',t))$

$W(s,t) = \max_{\{s' | (s,s') \in E\}} (w(s,s') + W(s',t)) = w(s, \mathrm{LSP}(s,t)) + W(\mathrm{LSP}(s,t), t)$

b) Let $V_s = \{v \in V | \exists \text{path from } s \text{ to } v\}$, $E_s = \{(u,v) \in E | u, v \in V_s\}$

There are at most $|V_s|$ subproblems $W(s',t)$, $\mathrm{LSP}(s',t)$, where $s' \in V_s$

Suppose we know whether a vertex is connected with t, then the total number of subproblems is $|\{v \in V_s | v \text{ is connected to } t\}|$

2.

a) Let s denote the string we deal with.

Let $W(i,j)$ denote the length of the longest palindrome subsequence of the string $s(i{:}\, j)$.

$$W(i,j) = \begin{cases} 2 + W(i+1, j-1) & \text{if } s(i) == s(j) \\ \max(W(i+1,j), W(i,j-1)) & \text{otherwise} \end{cases}$$

Let $p(i,j)$ denote the corresponding longest palindrome subsequence of the string $s(i{:}\, j)$.

$$p(i,j) = \begin{cases} s(i)p(i+1,j-1)s(j) & \text{if } s(i) == s(j) \\ p(i+1,j) & \text{if } W(i+1,j) > W(i,j-1) \\ p(i,j-1) & \text{if } W(i+1,j) \leqslant W(i,j-1) \end{cases}$$

b) The subproblems are all $W(i,j)$, $p(i,j)$ s.t. $i \leqslant j$. Therefore, worst case we have $\Theta(n^2)$.

3.

4.

a) Let $l_1, l_2, \ldots, l_n$ denote the length of strings we are dealing with.

Let s denote the string we deal with.

Let $W(i,j)$ denote the minimal cost of printing word i to j starting on a new line.

If $s(i{:}\, n)$ can be printed on one line, then certainly the cost is zero. Otherwise, the minimum cost is the minimum (over j) of the cost of this line $s(i{:}\, j)$ plus the minimum cost of the rest lines $s(j{:}\, n)$.

$W(i,n) =$

$$\begin{cases} 0 & \text{if } M - n + i - \sum_{k=i}^{n} l_k \geqslant 0 \\ \min\limits_{M - j + i - \sum_{k=i}^{j} l_k \geqslant 0} [(M - j + i - \sum_{k=i}^{j} l_k)^3 + W(j+1, n)] & \text{otherwise} \end{cases}$$

And we then print the sequence of words accordingly.

b) The subproblems are all $W(i,n)$, s.t. $i < n$. Therefore, worst case we have $\Theta(n)$.

5.

6.

a) Let s denote the string we deal with.

Let $C(N)$ denote the conviviality of node N.

Let $W(N)$ denote the maximal conviviality of the subtree rooted at node N.

If the person represented by node N is to be selected, then the total maximum conviviality is just the sum of this person's conviviality and the maximum conviviality starting from one of the node's grand children. Otherwise, the total maximum conviviality is the maximum conviviality starting from one of the node's children.

$W(N) = \max\left(C(N) + \max_{P\in \text{GrandChildren of } N}(W(P)), \max_{P\in \text{Children of } N}(W(P))\right)$

Note if N is leaf, then $W(N) = C(N)$.

And we then select the nodes accordingly.

b) The subproblems are all $W(N)$, where N is any node. Therefore, worst case we have $\Theta(|\text{Nodes}|)$ subproblems.

7.

a) Let $P(w, i)$ denote the maximum possibility of starting from node w, to achieve sequence σ_i, $\sigma_{i+1}, \ldots, \sigma_n$

$P(w, w) = 1$

We denote $P(w, x) = P(w, x)$, if $w\,x \in E$, the probability put on the edge $w\,x$.

The maximum possibility is simply the maximum (over next vertex x) of the product of possibility of edge $w\,x$ and the maximum possibility of starting from node x, to achieve sequence $\sigma_{i+1}, \sigma_{i+2}, \ldots, \sigma_n$.

$P(w, i) = \max_{\{x | wx\in E, \sigma(w,x) = \sigma(i)\}}\left(P(w, x)P(x, i+1)\right)$

Note that if w is a leaf/end vertex, and the sequence has not been achieved, then the possibility becomes zero.

And we select the argmax as the path nodes.

b) The subproblems are all $P(w,\, i)$, where w is a node connected to v_0, and i is from 1 to n. Though the real size will definitely be smaller, we can estimate as $\Theta(|V|n)$

8.

a) Let $M(i, j)$ denote the minimum disruption for removing $A(i, j)$ and removing pixels on row $i+1$ to m.

Then the minimum total disruption is simply the disruption of this pixel plus the minimum total disruption starting from the feasible pixels on the next row.

$$M(i, j) = \begin{cases} d(i, j) & \text{if } i == m \\ d(i, j) + \max_{k=0,\pm 1}\left(d(i+1, j+k)\right) & \text{otherwise} \end{cases}$$

Note that if $j + k$ is out of range, then we simply don't consider such case.

And then we select the pixels accordingly.

b) The subproblems are all $M(i, j)$ where $i \leqslant m$, $j \leqslant n$. Therefore $\Theta(\mathrm{mn})$.

9.

a) Let $l_1, l_2, ..., l_m$ be the break points, let $l_0 = 0$, $l_{m+1} = n$

Let $C(i, j)$ denote the minimum cost of breaking the string $S(l_i + 1{:}\, l_j)$, which contains the break points $l_{i+1}, ..., l_{j-1}$.

$C(i, i+1) = 0$

$C(i, j) = l_j - l_i + \min_{k=i+1:j-1} (C(i, k) + C(k, j))$

And we then determine the order of breaking accordingly.

b) The subproblems are $C(i, j)$, where $0 \leqslant i < j \leqslant m + 1$. Therefore, worst case we have $\Theta(m^2)$ subproblems.

10.

11.

12.

a) Let $p(i, j)$ denote the j^{th} player for the i^{th} position.

Let $M(i, j)$, $V(i, j)$ denote the money, and VORP, respectively, of the j^{th} player for the i^{th} position.

Let $P(i, Y)$ denote the maximum sum of VORP for position $i, i+1, ..., N$, with money Y.

$P(i, Y) = \max_{\{p(i,j) | Y - M(i,j) \geqslant 0\}} (V(i, j) + P(i+1, Y - M(i, j)))$

We then select the argmax as the player.

b) One possible estimation is $\Theta(NX)$, since the subproblems $P(i, Y)$ ranges over i and Y, where they take values from 1 to N, and 1 to X.

We acknowledge that such estimation may be overestimation, since it's unlikely that the subproblems actually go over each value of X (even if we normalize by 10000). We could also estimate the number of possible values of X with the number of all possible combinations of players on positions, which is P^N; however, considering the magnitude, there is expected to be significant overlapping, which makes P^N an even worse estimation of the possible values of X.

Therefore, in conclusion, $\Theta(NX)$.