1.

a. the modified file is called Index - yt2690.py

b. all results (selected stocks, percentages and running time) are in

$n=20 \rightarrow$ LP - num20.log

$n=40 \rightarrow$ LP - num40.log

$n=60 \rightarrow$ LP - num60.log

2. (a) the objective function means to maximize the total profit of selected objects.

the constraint means the total weights of selected objects cannot exceed the capacity B.

(b) the objective function means to maximize the total profit of selected objects.

the constraint means there exist a cover I in which is a set of selected objects

1. $\sum_{i \in I} w_i > B$: the total weight of I exceed the capacity B.
2. $\sum_{i \in I} x_i \le |I| - 1$: at most $|I| - 1$ number of objects can be selected, when this condition is satisfied I is called minimal cover.

the idea of this formulation is to find a minimal cover that maximizes the objective function when we don't select one of the objects in the cover.

2. when we reach the optimal solution, then both $LP_1 \subseteq LP_2$ and $LP_2 \subseteq LP_1$ hold

for example, when there are 4 objects with weight [1, 1, 1, 1], profit [1, 2, 3, 4] and the capacity B is 3

ⓒ ⓓ the profit/weight = [1, 2, 3, 4],
the LR solution for IP_1 is [0, 1, 1, 1]
the LR solution for IP_2 is [0, 1, 1, 1]
in this instance $LP_1 \subseteq LP_2$ and $LP_2 \subseteq LP_1$

ⓔ weight = [1, 2, 3, 4, 5]
profit = [3, 4, 5, 6, 7]
profit / weight = [3, 2, 1.67, 1.5, 1.4]
capacity = 4.

- the LR solution for IP_1 is $[1, 1, \frac{1}{3}, 0, 0]$.
this is invalid in IP_2 since the total weight of this solution does not exceed the capacity
- the LR solution for IP_2 is [1, 1, 0.5, 0.5, 0]
this is invalid in IP_1 since the total weight exceed the capacity

2. (a) graph : Σ^n, 0-1 vectors.

edge weight : hamming distance

terminal nodes : the vectors that represent the genes of every species

steiner nodes : the longest common subsequence between two nodes and the rest positions are all 0's.

root : the root of RMST will be the longest common subsequence among all genes and might be a $\vec{0}$ vector.

for example the root among [(11100), (11000), (00011)] is (00000)

To formulate OMPTP as a RMST, we have to allow the r node can have multiple edges that connect either steiner nodes or terminal nodes

The rest part of OMPTP is the same as RSMT that has to reach all terminal nodes from the r rooted node.

And, because of the rule of root and using hamming distance as edge weight function, we can make sure once a character has been developed, it will not be undeveloped in the future species.

(b) $f(i,j)$: flow from node i to node j.

c_{ij} : edge weight of edge (i,j)

x_{ij} : whether connect node i and node j. $= \begin{cases} 1, & \text{connected} \\ 0, & \text{ow.} \end{cases}$

$\delta^-(i)$: out flow of node i

$\delta^+(i)$: in flow of node i

$$\min \sum_i \sum_j c_{ij} x_{ij}$$

$f(\delta^-(r)) - f(\delta^+(r)) = 1$ 1 unit flow out from node r

$f(\delta^-(i)) - f(\delta^+(i)) \geq 0 \quad i \in V \setminus \{r\} \cup T$ steiner nodes have at least one out-flow when have one in flow.

$f(i,j) \leq x_{ij}$.

$f(i,j) \geq 0$

$\sum_i x_{ij} \geq 1 \quad \forall j \in T$ all terminal node should be connected.

$\sum_j x_{rj} \geq 1$ root node should be the root

$x_{ij} \in \mathbb{Z}$

(c) no species evolve from the modern species:

$f(\delta^-(i)) - f(\delta^+(i)) = -1 \quad i \in T$

this means no out-flow and there should be a in-flow for each terminal node.

each character can only be developed once:

$\sum_A x_e \leq 1 \quad \forall e$ $A = \{ x_{ij}$ that connects node i and node j who has hamming distance equals 1 on character $e \}$.