CS 301 Assignment 5

Oytun Kuday Duran (Student) - 28357

January 2023

1 Decision Version of Problem

Inputs:

- set **T** of projects (Assuming e_t and p_t are included in a project t ∈ T like a struct)
- positive integer **c** that is maximum effort can be spent
- positive integer **m** that is minimum profit allowed

Outputs:

↦ Yes, if a subset such that T' ⊆ T where effort is at most **c** and profit is at least **m** exists
↦ No, otherwise (If there is no such subset)

2 Proof

To show this problem is in **NP** , it is needed to show that it is finding a solution in exponential time at worse and there exists a polynomial-time verifiable certificate for the solution.

- **Getting a candidate:** Assuming n=$|T|$, then there are $2^n - 1$ **nonempty** subsets T' ⊆ T. One can simply keep check through subsets of T using a naive algorithm (Or randomly choosing) and check whether it is a valid certificate or not until finding a valid subset. For each T', it is needed to sum the costs and calculate the profits which are constant time operations for which each subset takes polynomial time, In worst case, for checking all subsets, it takes exponential time $O(2^n)$ for checking all subsets. If an arbitrary subset is chosen (Guessing/Non-deterministically), it takes polynomial time.
- **Verifying the certificate:** Correctness of a subset T' ⊆ T can be checked in polynomial time by iterating through the the t ∈ T' and calculating profit by subtracting the effort from the gain, then comparing it with (it should be at least) **m**, and summing the effort to compare with (it should be at most) **c**. These can be done in one iteration through subset each having a constant time, verification step takes $\Theta(|T'|)$ which is polynomial. Therefore, a certificate is verifiable in polynomial time.

Moreover, to prove this is in **NP** using membership proof, we can use a non-deterministic polynomial algorithm as has been discussed in class. This can be done in following steps similarly:

→ **Guessing** (Non-deterministic choosing) a T' ⊆ T.

→ **Iterate** each t ∈ T' and compute total effort required for and the total profit of subset of projects T'.

→ **If** total effort spent of T' is at most c and total profit of T' is at least m, then return (output) **Yes**

→ **Otherwise** return (output) **No**

As can be seen, this decision problem can be solved with a non-deterministic polynomial algorithm / by guessing a candidate, and as also mentioned above, candidate solutions (certificates) can be verified in polynomial time as well. As the input and output are also polynomial sized, this decision problem is in **NP**.