Max Profit Problem

## Introduction:

The Max Profit Problem is a scenario where Mr. X owns a large strip of land in Mars Land and wants to maximise his earnings by developing different types of properties. Each property type has a specific construction time and earns a certain amount of money when operational. The goal is to find the optimal mix of properties based on a given time unit.

## Problem Description:

Mr. X has the option to build Theatres, Pubs, and Commercial Parks on his land. Each property type has its own construction time and occupies a specific parcel of land. The earning potential for each property type is as follow

* Theatre: $1500
* Pub: $1000
* Commercial Park: $3000

## Constraints:

* Mr. X cannot develop two properties simultaneously in the same time unit.
* The available time unit is provided as an input.

## Approach:

The problem can be solved recursively by considering different time units and calculating the maximum earnings for each time unit. The approach includes the following steps:

* Define a recursive function named find(n) that takes a time unit n as input and returns the maximum earnings for that time unit.
* Initialise a list prof to store the profits for each property type (Theatre, Pub, Commercial Park). Initialise it with zeros.
* Check if the time unit n is less than 4. If so, return 0, as there are no properties that can be developed.
* If the time unit is 4 or greater, calculate the profits for each property type based on the remaining time (temp) after developing the property.
* Determine the index of the property type with the maximum profit using the getMax(prof) function.
* Add the maximum profit to the total earnings pr and increment the count of completed properties for the corresponding property type in the arr list.
* Recursively call the find() function with the reduced time unit by subtracting the time required for the developed property.
* Repeat the above steps until the base case is reached (time unit < 4).
* Implement the getMax(prof) function to determine the index of the property type with the highest profit.
* Implement the solString() function to generate a formatted string representing the number of properties developed for each property type.
* Initialise the array list to keep track of the number of properties completed for each property type.
* Define the time\_unit list that stores the time required for each property type.
* Prompt the user to input the time unit.
* Print the maximum earnings by calling the find() function with the user input.
* Print the solution string using the solString() function.

## Conclusion:

The Max Profit Problem provides an opportunity for Mr. X to maximise his earnings by choosing the right mix of properties to develop based on a given time unit. By implementing a recursive solution, the code