Assignment Solutions | Recursion - 4 | Week 11

1. Given an integer array containing unique numbers, return power set, containing all the subsets of the set. [Leetcode 78]

Solution:

```
void subsets(vector<int> &nums, vector<vector<int>> &ans, vector<int> &temp, int
idx) {
        if(idx == nums.size()) {
            ans.push_back(temp);
            return;
        }
        // not taken
        subsets(nums, ans, temp, idx+1);
        // taken
        temp.push_back(nums[idx]);
        subsets(nums, ans, temp, idx+1);
        temp.pop_back();
    }

    vector<vector<int>> subsets(vector<int>& nums) {
        vector<vector<int>> ans;
        vector<int> temp;
        subsets(nums, ans, temp, 0);
        return ans;
    }
```

2. Given an integer array which may contain duplicate numbers, return power set, containing all the subsets of the set. [Leetcode 90]

Solution:

```
void subsetsWithDup(vector<int> &nums, vector<vector<int>> &ans, vector<int> &temp, int idx) {
       if(idx == nums.size()) {
           ans.push_back(temp);
           return;
       }
       // not taken
       int i = idx+1;
       while(i < nums.size() && nums[i] == nums[i-1]) {
           ++i;
       }
       subsetsWithDup(nums, ans, temp, i);
       // taken
       temp.push_back(nums[idx]);
       subsetsWithDup(nums, ans, temp, idx+1);
       temp.pop_back();
   }

   vector<vector<int>> subsetsWithDup(vector<int>& nums) {
       vector<vector<int>> ans;
       vector<int> temp;
       sort(nums.begin(), nums.end());
       subsetsWithDup(nums, ans, temp, 0);
```

3. Given a string, find the length of the longest common substring from two given strings.

Solution:

```
#include <bits/stdc++.h>
using namespace std;
int longestCommonSubstring(string &a, string &b, int idxA, int idxB) {
    if(idxA == a.size() || idxB == b.size()) {
        return 0;
    }
    if(a[idxA] == b[idxB]) {
        return 1 + longestCommonSubstring(a, b, idxA+1, idxB+1);
    }
    return max(longestCommonSubstring(a, b, idxA+1, idxB), longestCommonSubstring(a,
b, idxA, idxB+1));
}

int main() {

    string a, b;
    cin >> a >> b;

    int ans = longestCommonSubstring(a, b, 0, 0);
    cout << ans << endl;

    return 0;
}
```

4. Program to find the factorial of a given number.

Solution:

```
num = int(input("Enter a number: "))
fact = 1

if num < 0:
    print("Factorial does not exist for negative numbers.")
elif num == 0:
    print("Factorial of 0 is 1.")
else:
    for i in range(1, num+1):
        fact = fact * i
    print("Factorial of", num, "is", fact)
```

5. Program to convert a decimal number to binary.

Solution:

```python
decimal = int(input("Enter a decimal number: "))
binary = ""

while decimal > 0:
    remainder = decimal % 2
    binary = str(remainder) + binary
    decimal = decimal // 2

print("Binary equivalent is", binary)
```
