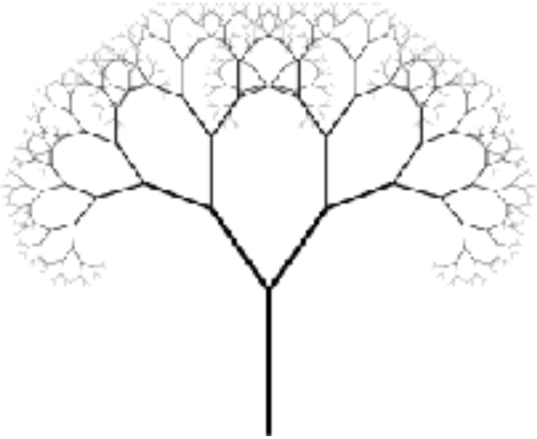
# STRINGS RECURSION

Problem Solving with Computers-I



# Strings

Q1: How are ordinary arrays of characters and C-strings similar and how are they dissimilar?

# The C++ string class methods

```
string fruit = ”Apple";
int len = fruit.length();
int pos= fruit.find(‘l’);
string part= fruit.substr(1,3);
fruit.erase(2,3);
fruit.insert(2,“ricot”);
fruit.replace(2,5,“ple”);
```

Check out cctype for checks and conversions on characters

```
fruit[0]= tolower(fruit[0]);
isalpha(fruit[0])
isalnum(fruit[0])
```

# What is the output of the code?

```
char s1[] = "Mark";
char s2[] = "Jill";
for (int i = 0; i <= 4; i++)
     s2[i] = s1[i];
if (s1 == s2) s1 = "Art";
cout<<s1<<" "<<s2<<endl;
```

A. Mark Jill

B. Mark Mark

C. Art Mark

[✓ D.] Compiler error

E. Run-time error

# What is the output of the code?

```
string s1 = "Mark";
string s2 = "Jill";
for (int i = 0; i <= s1.length(); i++)
     s2[i] = s1[i];
if (s1 == s2) s1 = "Art";
cout<<s1<<" "<<s2<<endl;
```

A. Mark Jill

B. Mark Mark

[✓ C.] Art Mark

D. Compiler error

E. Run-time error

# Lab 08: anagrams and palindromes

```
bool isPalindrome(string s1)
```

deTartraTED
WasItACarOrACatISaw

```
bool isAnagram(string s1, string s2)
```

Diba == Adib
Rats and Mice == In cat's dream
Waitress == A stew, Sir?

Why don’t we pass the length of the string?

```
double sumList(Node* head){
    [Needs a base case up here]
    double sumRest;
    sumRest = sumList(head->next);
    return head->data + sumRest;
}
```