Question 1

Let T (n) be the number of <u>additions</u> that the function fct does. Assume n= 2^z(i.e., n = 1, 2, 4, 8, 16, ...),

```
int fct(const int A[], int n){
   if ( n == 1 )
     return A[n]+A[n]
   else
      return  A[n]+A[n]+fct(A, n/2);
}
```

Part A)) Write a recurrence relation for T(n) where n is the number of elements in the array A.

Part B) Solve the recurrence relation to obtain T(n) in terms of n. Show your workings.

Part C) Write the resulting expression of part 2) in big O notation.

Part D) Name an algorithm that has this same running time complexity.

Question 2

a) What is the output of the C++ program below?

```
//assume a,b >= 0
int f(int a,int b){
      if(a==0 && b == 0){
            return 0;
      }
      if(a == 0){
            return f(a,b-1)-1;
      }
      return 1 + f(a-1,b);
}
```

What is returned by the call f(3,8)

What does this function do?

b) Consider the following recursive C++ function.

```
int foo(int n){
  if(n == 0){
    return 1;
  }
  int x = 0;
  for(int i = n; i > 0;i--){
    x+= foo(n-1);
  }
  return x;
}
```

What is returned by the call foo(4)

What does this function do?

The next 3 questions need to be coded and have to compile and run. You should start with mt1started.cpp found in D2L

Question 3 (10 points)

Write a **recursive** function `isSorted` determines if a vector is in ascended order.

Question 4 (10 points)

Write a **recursive** function that is passed a vector of unique positive integers and a sum. Using that vector find all unique combines of integers from that vector that sums to the sum(You can repeat numbers). Use the function prototype

```
int sumThing(std::vector<int> & v,int sum);
```

Do not use globals, nor static varibales, but feel free to create extra strings .

Examples

```
Std::vector<int> A = {8,4,6,2};
```

`sumThing(A,10)` returns `6`

Because all these combine to sum to 10 (`22222 ,2224, 226,244,28,46`)

Question 5 (10 points)

Write a function that will be passed a string. Your program should print out all the letters that were not in the string. This should be case insensitive.