Exercises Set – 4

1. In the following program, replace the iterative function by an equivalent recursive function:

```
#include<iostream>
using namespace std;

void iterative_fn(int n);

int main() {
  int n;
  cout<<"Enter a positive  int\n";
  cin>>n;
  cout<<"\n";
  iterative_fn(n);

}


void iterative_fn(int n){
   for (int i=n; i>0;i--){
     cout<<i<<endl;
   }
}
```

2. Write a recursive function gcd that returns the greatest common divisor of x and y, which is defined recursively as follows: If y is equal to 0, then $gcd(x, y)$ is x; otherwise $gcd(x, y)$ is $gcd(y, x\%\ y)$, where % is the modulus operator.

3. Write a function triangle(unsigned int m, unsigned int n) with the following properties:

```
  void triangle(unsigned int m, unsigned int n)

  // Precondition: m <= n
  // Postcondition: The function has printed a pattern of 2*(n-m+1) lines
  // to the output stream outs. The first line contains m asterisks, the next
  // line contains m+1 asterisks, and so on up to a line with n asterisks.
  // Then the pattern is repeated backwards, going n back down to m.

  /* Example output:
     triangle(3, 5) will print this to cout:
     ***
     ****
     *****
     *****
     ****
     ***
  */
```