```
*********Program 38.c ******************************
#include<stdio.h>
//iterative and recursive factorial
int ifact(int);
int rfact(int);
int main() {
  int i;
  for (i = 0; i < 10; i++) {
    printf("fact of %d = %d %d\n", i, ifact(i), rfact(i));
  }

}

int ifact(int n) {
  int i, rv = 1;
  for (i = 1; i <= n; i++) {
     rv = rv * i;
  }
  return rv;
}

int rfact(int n) {
  if (n <= 1) {
    return 1;
  } else {
    return n * rfact(n-1);
  }
}
*********Program 39.c ******************************
#include<stdio.h>
//recursive fibonacci series...
long fib(long);
int main() {
  long x, res;
  scanf("%ld",&x);
  res = fib(x);
  printf("fib(%ld) = %ld\n", x, res);
}

long fib(long n) {
  if (n == 0 || n == 1) {
    return n;
  } else {
    return (fib(n-1) + fib(n-2));
  }
}
*********Program 40.c ******************************
#include<stdio.h>
//recursive gcd...
int gcd(int, int);
int main() {
  int x, y, res;
  //assume x > y
  scanf("%d%d",&x,&y);
  res = gcd(x,y);
  printf("gcd of %d %d is %d\n", x, y, res);
}

int gcd(int x, int y) {
 if (y == 0) {
    return x;
  } else {
    return gcd(y, x%y);
  }
  //use ternary operator, a single line of code..
  //return (y == 0) ? x : gcd(y,x%y);
}

*********Program 41.c ******************************
#include<stdio.h>
int main() {
 static int x = 1;
 printf("value of x is %d\n", x);
 x++;

 if (x > 100) {
  return 0;
 }
 main();
}
```