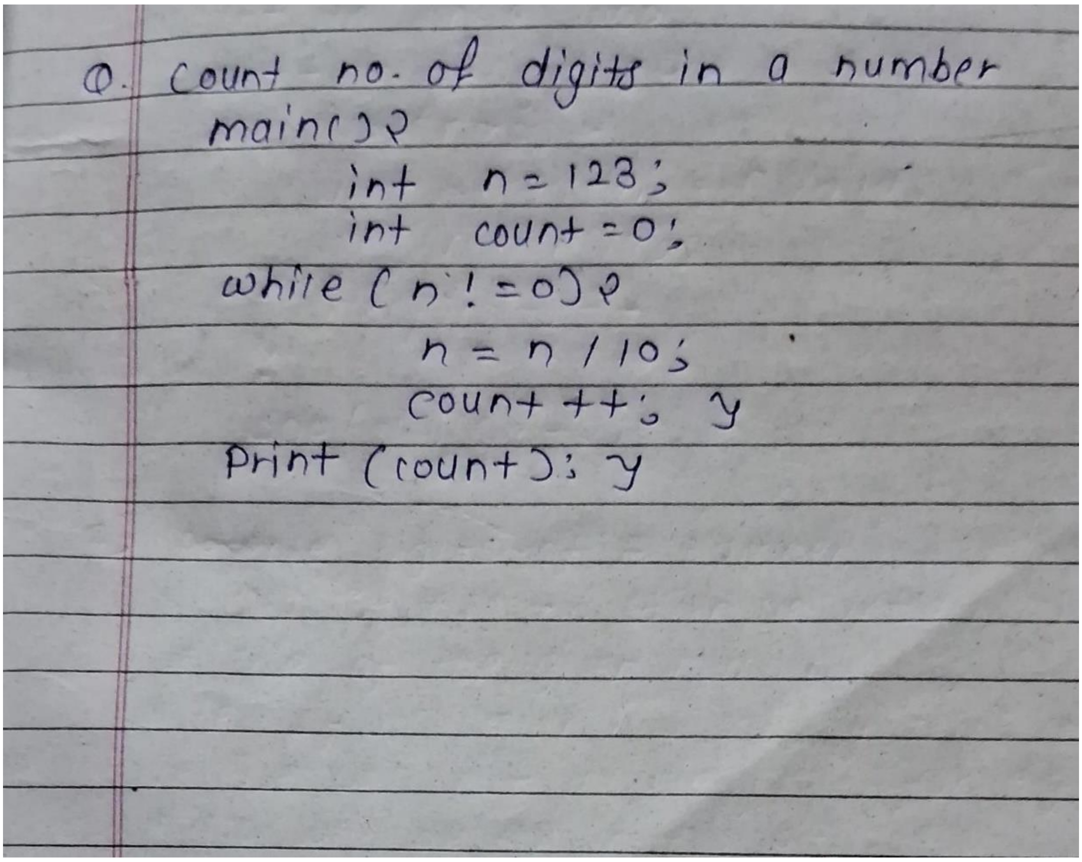
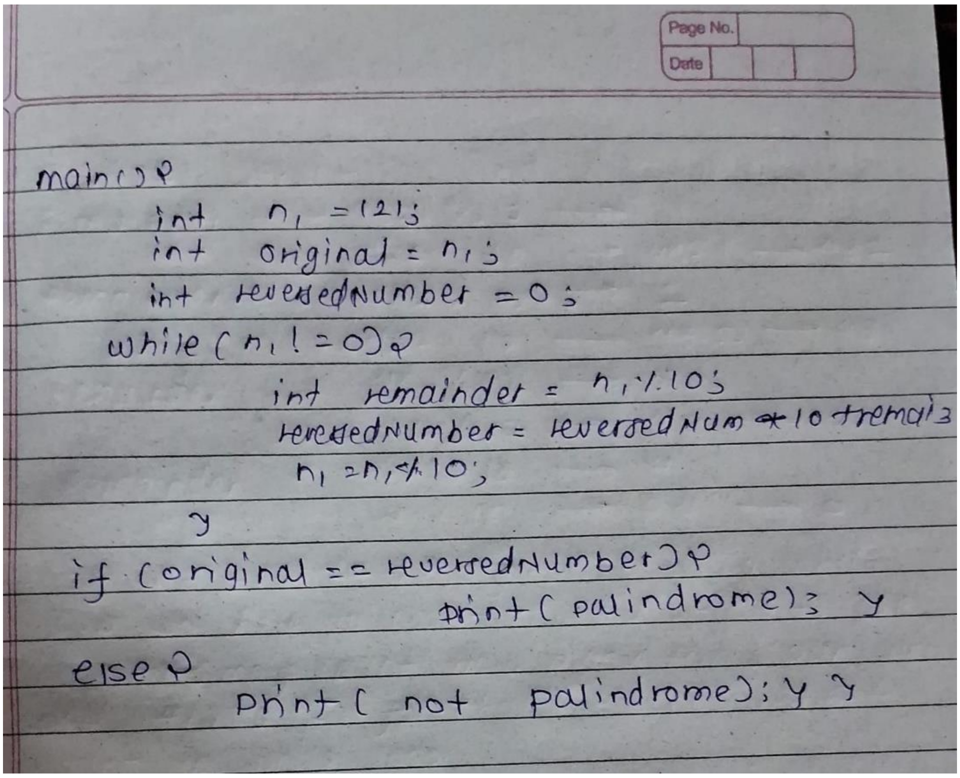
Assignment No. 1



```
Q. Count no. of digits in a number
   main(){
          int   n=123;
          int   count=0;
   while (n!=0){
             n = n/10;
             count++; }
   Print (count); }
```

1. class Demo1{

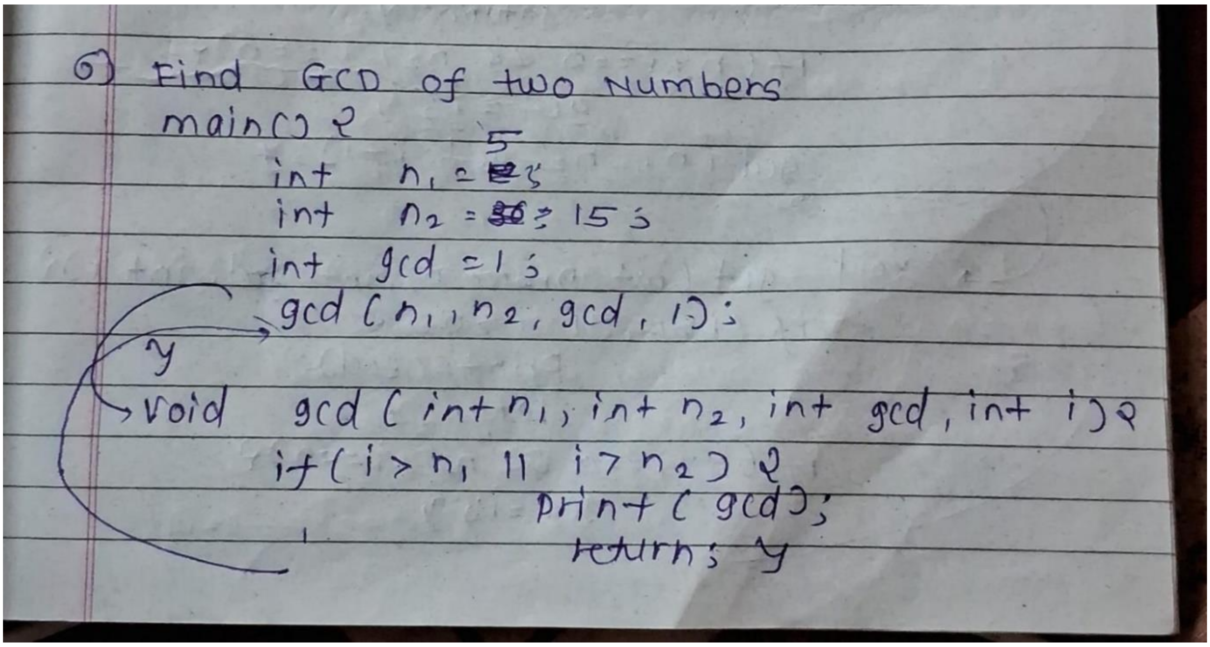
```
       public static void main(String[]args){
               System.out.println(countTheDigits(12345));

}
     public static int countTheDigits(int number){
               int count=0;
               while(number>0){
             int remainder=number%10;
               number=number/10;
               count++;
}
               return count;
}


}
```



Page No.
Date

```
main(){
      int  n1 =121;
      int  original = n1;
      int  reversedNumber = 0;
   while(n1!=0){
             int remainder = n1%10;
             reversedNumber = reversedNum*10+remai3
             n1 =n1/10;
      }
  if (original == reversedNumber){
                       print( palindrome); }
  else{
          print( not   palindrome); } }
```

```
2.  class Demo3{
      public static void main(String[]args){
              isPalindrome(121);

}

  public static void isPalindrome(int number){
              int original=number;
              int reversedNumber=0;
              while(number>0){
             int remainder=number%10;
              number=number/10;
              reversedNumber=reversedNumber*10+remainder;
              }
              if(original == reversedNumber){
                      System.out.println("Number is palindrome");
      }
              else{
                      System.out.println("Number is not palindrome");

      }

}
 }
```



6) Find GCD of two Numbers

```
main() {
        int n1 = 5;
        int n2 = 15;
        int gcd = 1;
        gcd (n1, n2, gcd, 1);
  }
void  gcd (int n1, int n2, int gcd, int i){
        if(i > n1 || i > n2) {
                      print( gcd);
                       return; }
```

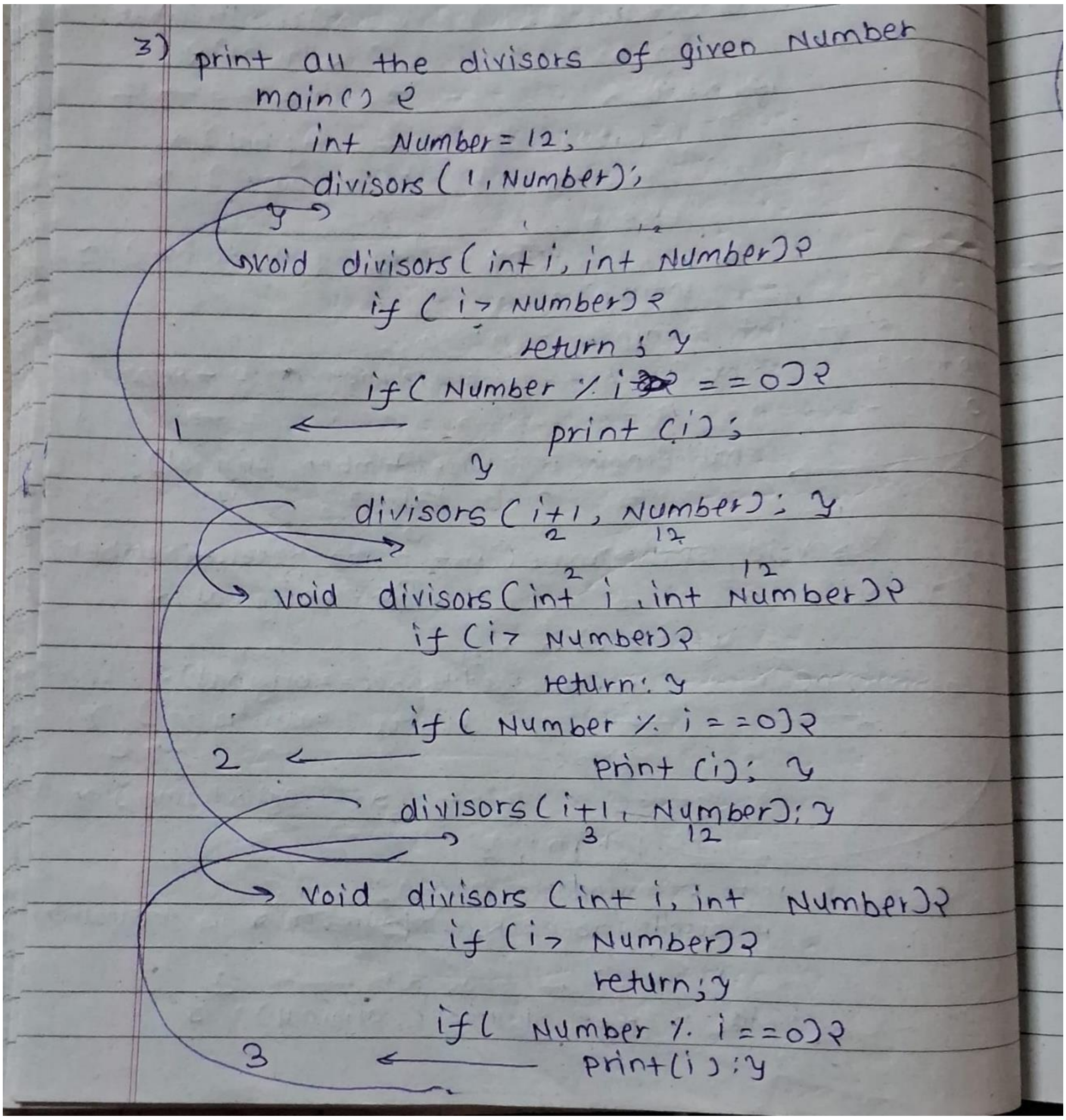
```
if (n1% i==0 && n2% i==0){
            gcd = i; }
gcd(n1, n2, gcd, i+1); }

  void gcd (int n1, int n2, int gcd, int i){
     if( i > n1 || i > n2){
                print (gcd);
                return; }
2    if( n1% i==0 && n2% i==0){
                gcd = i; }
     gcd( n1, n2, gcd, i+1); }

  void gcd( int n1, int n2, int gcd, int i){
     if( i > n1 || i > n2){
                print (gcd);
                return; }
3    if( n1% i==0 && n2% i==0){
                gcd = i; }
     gcd( n1, n2, gcd, i+1); }

  void gcd (int n1, int n2, int gcd, int i){
     if( i > n1 || i > n2){
                print (gcd);
                return; }
4    if( n% i==0 && n2% i==0){
                gcd = i; }
     gcd (n1, n2, gcd, i+1); }

  void gcd (int n1, int n2, int gcd; int i){
5    if (i > n1 || i > n2){
                Print (gcd);
                return; }
     if (n% i==0 && n2 % i==0){
                gcd = i; }
```

```
        3. class gcd{
           public static void main(String[]args){
                   int n1=12;
                   int n2=36;
                   int gcd=1;
                   gcdTwo(n1,n1,1,gcd);
   }

         public static void gcdTwo(int n1,int n2,int i,int gcd){
                   if(i>n1||i>n2){
                           System.out.println(gcd);
                           return;}
                   if(n1%i==0 && n2%i==0){
                           gcd=i;}
                   gcdTwo(n1,n2,i+1,gcd);

   }


   }
```

Page No.
Date

```
main(){
        int number = 153;
        int original = number;
        int count = CountDigits (number, 0);
        isArmstrong (number, original, 0, count);
}
static int countDigits( int number, int count){
            if (number == 0) {
                        return count; }
            number = number /10;
            count ++;
            return countDigits( number, count); }
void isArmstrong ( int number, int original,
                           int sum, int count ){
        if (number == 0){
            if (sum == original) {
                               print(" Armstrong"); }
                               return;
            else{
                        Print ("Not Armstrong"); }
                        return; } }

        int remainder = number %10;
        sum = sum + (int) math.pow( remainder, count);
        number = number /10;
        isArmstrong( number, original, sum, count);

}
```

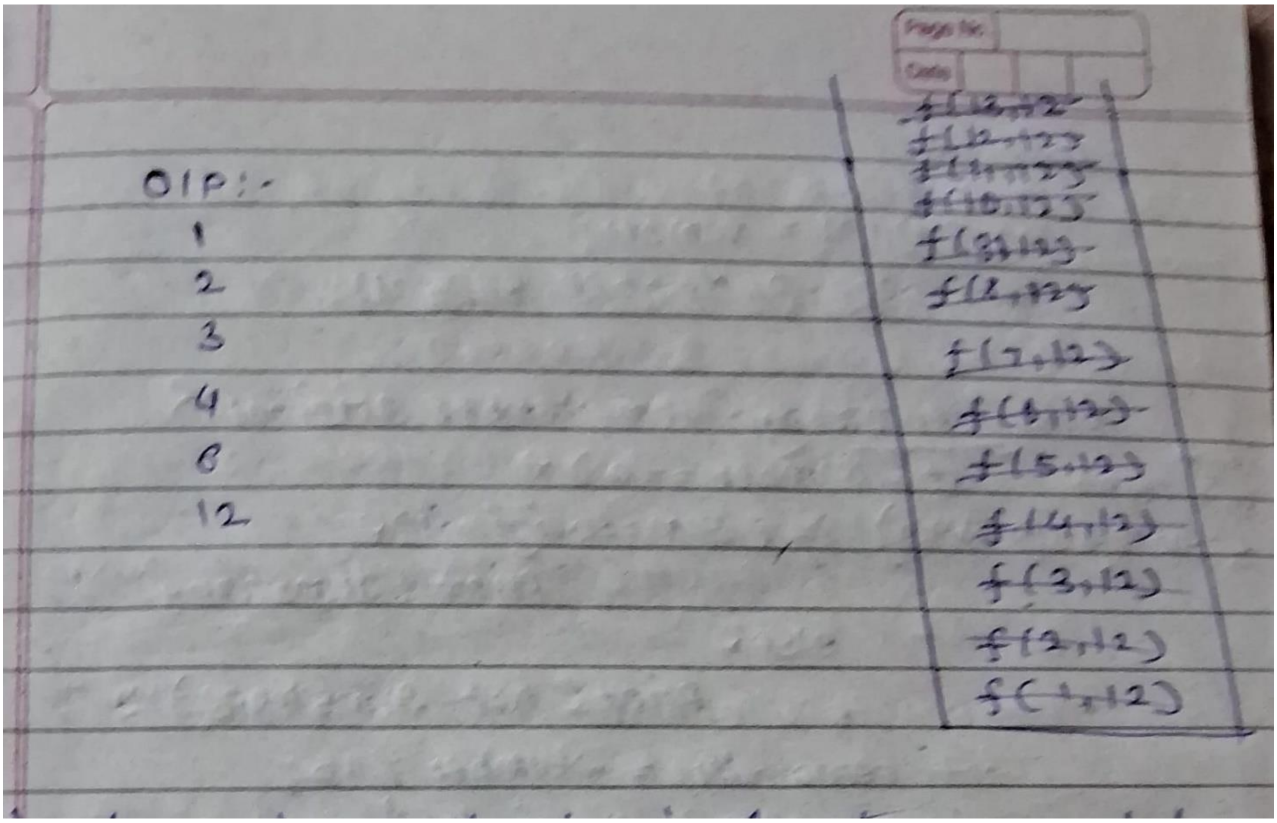
```
4  class Armstrong{
    public static void main(String[]args){
        int Number=153;
        int original=Number;
        int count=countDigits(Number,0);
        isArmstrong(Number,original,0,count);
}
        static void isArmstrong(int number,int original,int sum,int
count){
                if(number==0){
                    if(sum==original){
                        System.out.println("Number is
Armstrong");}
                    else{
                        System.out.println("Number is not
Armstrong");}
                    return;}
                int remainder=number%10;
                sum=sum+(int)Math.pow(remainder,count);
                number=number/10;
                isArmstrong(number,original,sum,count);}

        static int countDigits(int Number,int count){
                if(Number==0){
                    return count;}
                Number=Number/10;
                count++;
                return countDigits(Number,count);
}}
```



3) print all the divisors of given Number

```
main() {
    int Number = 12;
    divisors(1, Number);
}

void divisors(int i, int Number){
        if (i > Number){
                return; }
        if( Number % i == 0){
1   <--            print(i);
        }
        divisors(i+1, Number); }
                   2     12
                2           12
void divisors(int i, int Number){
        if (i > Number){
                return; }
        if( Number % i == 0){
2   <--            print(i); }
        divisors(i+1, Number); }
                 3      12
void divisors(int i, int Number){
        if (i > Number){
                return; }
        if( Number % i == 0){
3   <--        print(i); }
```

```
         9       12
divisors( i+1, Number);

void  divisors (~~int~~, int i, int Number){
          if ( i > Number) {
                  return; }
          if ( Number % i==0){
                  print (i); }
          divisors( i+1, Number); }
                    10      12

      void  divisors ( int i, int Number){
            if ( i > Number){
                   return; }
            if ( Number % i ==0){
                  print (i); }
            divisors ( i+1, Number); }
                       11      12

      void   divisors (int i, int Number){
                if (i > Number){
                         return; }
                if ( Number % i ==0){
                         print(i); }
                divisors(i+1, Number); }
                         12      12

      void  divisors (int i, int Number){
                if  (i > Number){
                         return; }
                if ( Number % i==0){
12  <-----                 print(i); }
               divisors(i+1, Number); }
                         13     12

      void   divisors (int i, int Number){
               if( i > Number){
                        return; }
               if ( Number % i==0){
                           print(i); }
               divisors ( i+1, Number); }
```



Page No.

Date

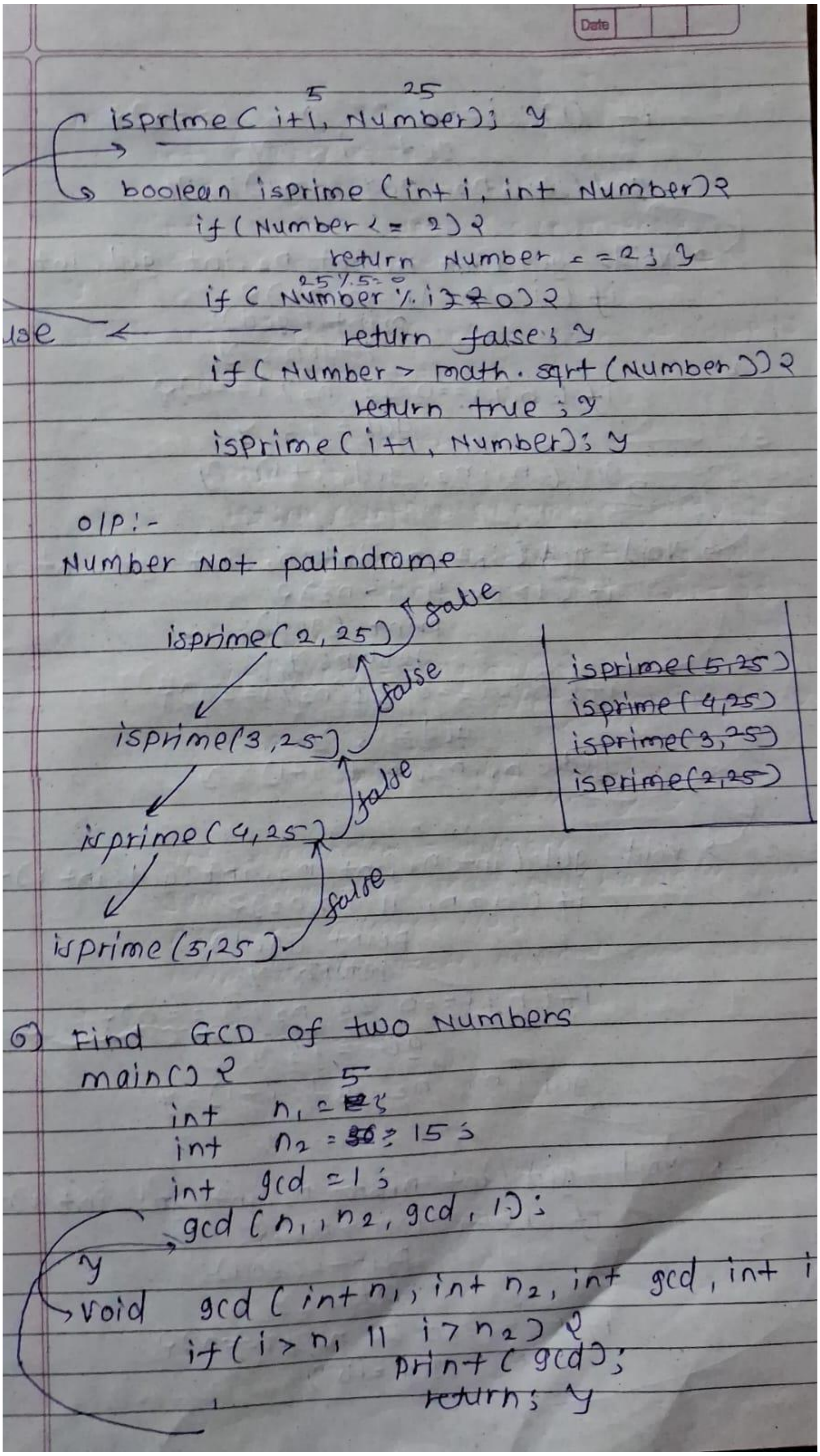
O/P:-
1
2
3
4
6
12

~~f(13,12~~
~~f(12,12)~~
~~f(11,12)~~
~~f(10,12)~~
~~f(9,12)~~
~~f(8,12)~~
~~f(7,12)~~
~~f(6,12)~~
~~f(5,12)~~
~~f(4,12)~~
~~f(3,12)~~
~~f(2,12)~~
~~f(1,12)~~

```
5. class factors{
        public static void main(String[]args){
                int Number=12;
                allFactors(Number,1);

}
        static void allFactors(int number,int i){
                        if(i>number){
                                return;}
                        if(number%i==0){
                                System.out.println(i);}
                        allFactors(number,i+1);}
}
```

5) check if a Number is prime or not.

```
——> main() {
            int Number = 25;
boolean check= isprime(2, Number);
  false     if( ~~isprime~~ check){
                       print(" Number is prime");}
            else {
                   print(" Number not prime");}
    }
                          2        25
    boolean isprime(int i, int Number){
           if (Number<=2){
                     return number==2;  }
           if (Number% i ==0){
                     return false; }
           if (Number > math.sqrt(Number){
                     return true; }
           isprime(i+1, Number);
    }               3      25

        boolean isprime(int i, int Number){
           if( Number <= 2){
                     return Number==2; }
           if (Number % i ==0){
                     return false; }
           if( Number > math.sqrt(Number)){
                    return true; }
          isprime(i+1, Number);
                    4     25

          boolean isprime( int i, int Number){
             if (Number <= 2){
                     return Number ==2; }
             if (Number % i ==0){
                        return false; }
             if (Number > Math.sqrt (Number) {
                     return true; }
```



```
              5      25
    isprime( i+1, Number); }

    boolean isprime (int i, int Number){
         if ( Number <= 2) {
                return Number == 2; }
              25 % 5 = 0
         if ( Number % i ) == 0 ) {
use             return false; }
         if ( Number > math. sqrt (Number )) {
                return true ; }
         isprime( i+1, Number); }
```

O/P:-
Number Not palindrome

isprime(2, 25) ↑ false

isprime(3 ,25) ↑ false

isprime(4,25) ↑ false

isprime (5,25)

~~isprime(5,25)~~
~~isprime(4,25)~~
~~isprime(3,25)~~
~~isprime(2,25)~~

6) Find GCD of two Numbers

```
main() {
       int  n1 = ~~2~~ 5;
       int  n2 = ~~30~~ 15 ;
       int  gcd = 1 ;
       gcd ( n1 , n2 , gcd , 1);
}
void   gcd ( int n1 , int n2 , int gcd , int i
       if( i > n1 || i > n2 ) {
                 print ( gcd);
                 return; }
```

6. class prime{

```
    public static void main(String[]args){
        int number=1;
        if(isPrime(number,2)){
            System.out.println("Number is prime");
            }
        else{
            System.out.println("Number is not prime");}
}
        static Boolean isPrime(int number,int i){
            if(number<=2){
                return number==2;}
            if(number%i==0){
                return false;}
            if(number>Math.sqrt(number)){
                return true;}
            return isPrime(number,i++);}




}
```