# CSE1007 – JAVA PROGRAMMING

Lab Exercise on Methods

## Name : Anupam Kunwar

## REG : 19BCE1369

**Question 1**

Write a JAVA program to find the GCD of any two numbers. Your program should have method findGCD() that return the gcd of the two numbers.

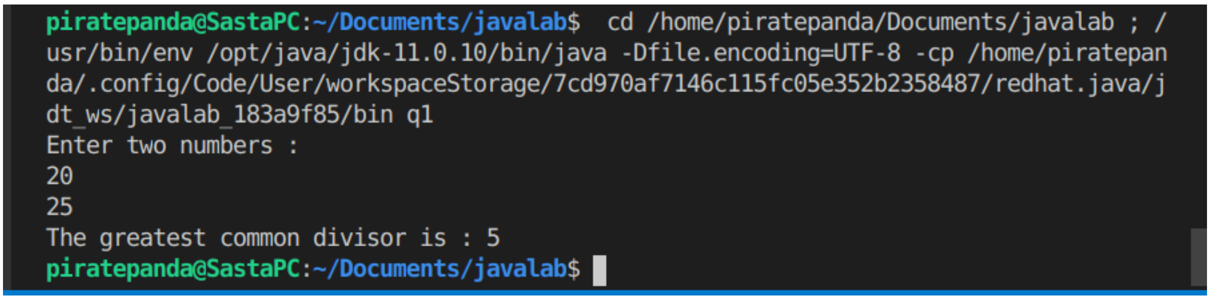
**Solution :**

```
import java.util.Scanner;


public class q1 {


public static int GCD(int n1,int n2){
if(n1==0)
return n2;
if(n2==0)
return n1;
if(n1==n2)
return n1;
if(n1>n2)
return GCD(n1-n2,n2);
return GCD(n1,n2-n1);
}
public static void main(String args[]){
Scanner in = new Scanner(System.in);
System.out.println("Enter two numbers : ");
int a = in.nextInt();
int b = in.nextInt();
int ans = GCD(a,b);
System.out.println("The greatest common divisor is : "+ans);
```

}

}

**Output :**



```
piratepanda@SastaPC:~/Documents/javalab$  cd /home/piratepanda/Documents/javalab ; /
usr/bin/env /opt/java/jdk-11.0.10/bin/java -Dfile.encoding=UTF-8 -cp /home/piratepan
da/.config/Code/User/workspaceStorage/7cd970af7146c115fc05e352b2358487/redhat.java/j
dt_ws/javalab_183a9f85/bin q1
Enter two numbers :
20
25
The greatest common divisor is : 5
piratepanda@SastaPC:~/Documents/javalab$ 
```

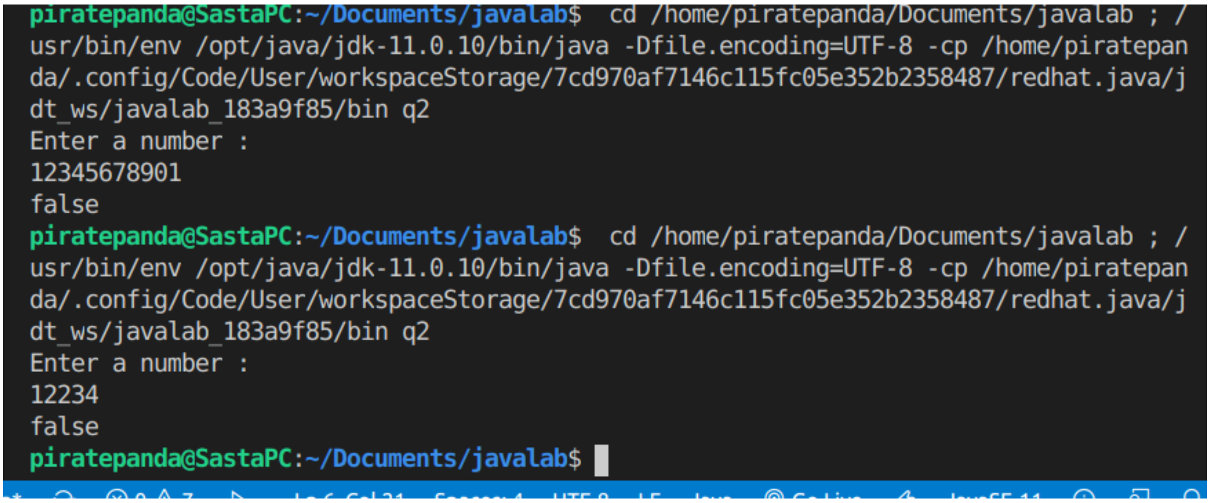
## Question 2

Travel Tickets Company sells tickets for airlines, tours, and other travel-related services. Because long numbers have often been entered incorrectly by agents, Travel Tickets has asked you to code a JAVA program that will indicate if a ticket number entry is invalid. Ticket numbers are 11 digits long. Ticket numbers are designed such that if you drop the last digit of the number, then divide the 10-digit number by 7, the remainder of the division will be identical to the last dropped digit. If ticket number is 10-digits, include the 11th digit or if it is 11-digit long, check for the validity. If the ticket number is any other length, your program should prompt the agent to check and re-enter the ticket number. Include a method **isValidTicket()** that return either true or false depending on the validity of the ticket.

**Solution :**

```
import java.util.Scanner;

public class q2 {


public static int lengthOf(long n){

int c=0;

while(n!=0){

n=n/10;

c++;

}

return c;

}
public static boolean isValidTicket(long n){
int length = lengthOf(n);
if(length!=10 && length!=11)
return false;
else if(length==10){
n = (n*10)+(n%7);
System.out.println("The valid ticket number is : "+n);
return true;
}
else{
long last = n%10;
n = n/10;
long rem = n%7;
if(rem==last)
return true;
return false;
}
}

public static void main(String args[]){
Scanner in = new Scanner(System.in);
System.out.println("Enter a number : ");
long a = in.nextLong();
boolean b1 = isValidTicket(a);
System.out.println(b1);
}
}
```

**Output :**



```
piratepanda@SastaPC:~/Documents/javalab$  cd /home/piratepanda/Documents/javalab ; /
usr/bin/env /opt/java/jdk-11.0.10/bin/java -Dfile.encoding=UTF-8 -cp /home/piratepan
da/.config/Code/User/workspaceStorage/7cd970af7146c115fc05e352b2358487/redhat.java/j
dt_ws/javalab_183a9f85/bin q2
Enter a number :
12345678901
false
piratepanda@SastaPC:~/Documents/javalab$  cd /home/piratepanda/Documents/javalab ; /
usr/bin/env /opt/java/jdk-11.0.10/bin/java -Dfile.encoding=UTF-8 -cp /home/piratepan
da/.config/Code/User/workspaceStorage/7cd970af7146c115fc05e352b2358487/redhat.java/j
dt_ws/javalab_183a9f85/bin q2
Enter a number :
12234
false
piratepanda@SastaPC:~/Documents/javalab$ 
```

## Question 3

Assume that you have a list of words and you wish to find how many words are palindrome in the list. Devise a JAVA program that reads several words and displays the palindrome words and count of such words. Your program should have two methods namely **reverseString()** and **isPalindrome()**.

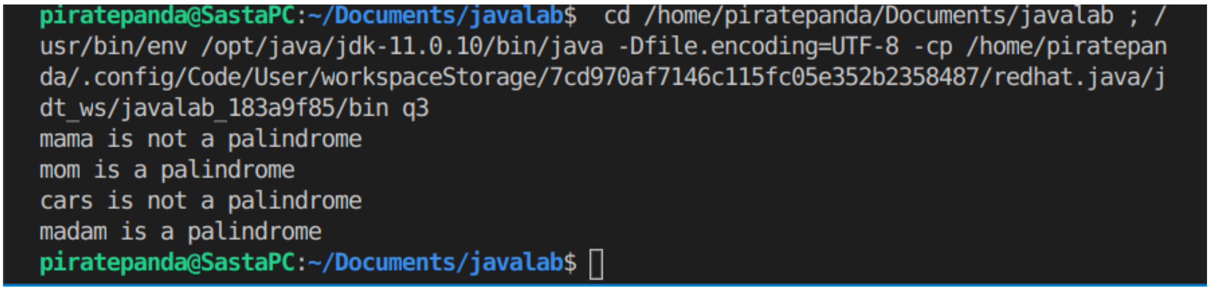
Hint: A word is palindrome if its reverse is same as the original.

**Solution :**

```
public class q3 {

public static String reverseString(String s){

int i;

String s1 = "";

int l = s.length();

for(i=l-1;i>=0;i--){

s1 = s1+s.charAt(i);

}

return s1;

}


public static boolean checkPali(String s,String s1){
```

```
return s.equals(s1);
}

public static void main(String args[]){
String rev;
String[] words = {"mama","mom","cars","madam"};
for(int i=0;i<words.length;i++){
rev = reverseString(words[i]);
if(checkPali(words[i],rev)){
System.out.println(words[i]+" is a palindrome");
}
else{
System.out.println(words[i]+" is not a palindrome");
}
}
}
}
```

**Output :**



```
piratepanda@SastaPC:~/Documents/javalab$  cd /home/piratepanda/Documents/javalab ; /
usr/bin/env /opt/java/jdk-11.0.10/bin/java -Dfile.encoding=UTF-8 -cp /home/piratepan
da/.config/Code/User/workspaceStorage/7cd970af7146c115fc05e352b2358487/redhat.java/j
dt_ws/javalab_183a9f85/bin q3
mama is not a palindrome
mom is a palindrome
cars is not a palindrome
madam is a palindrome
piratepanda@SastaPC:~/Documents/javalab$
```