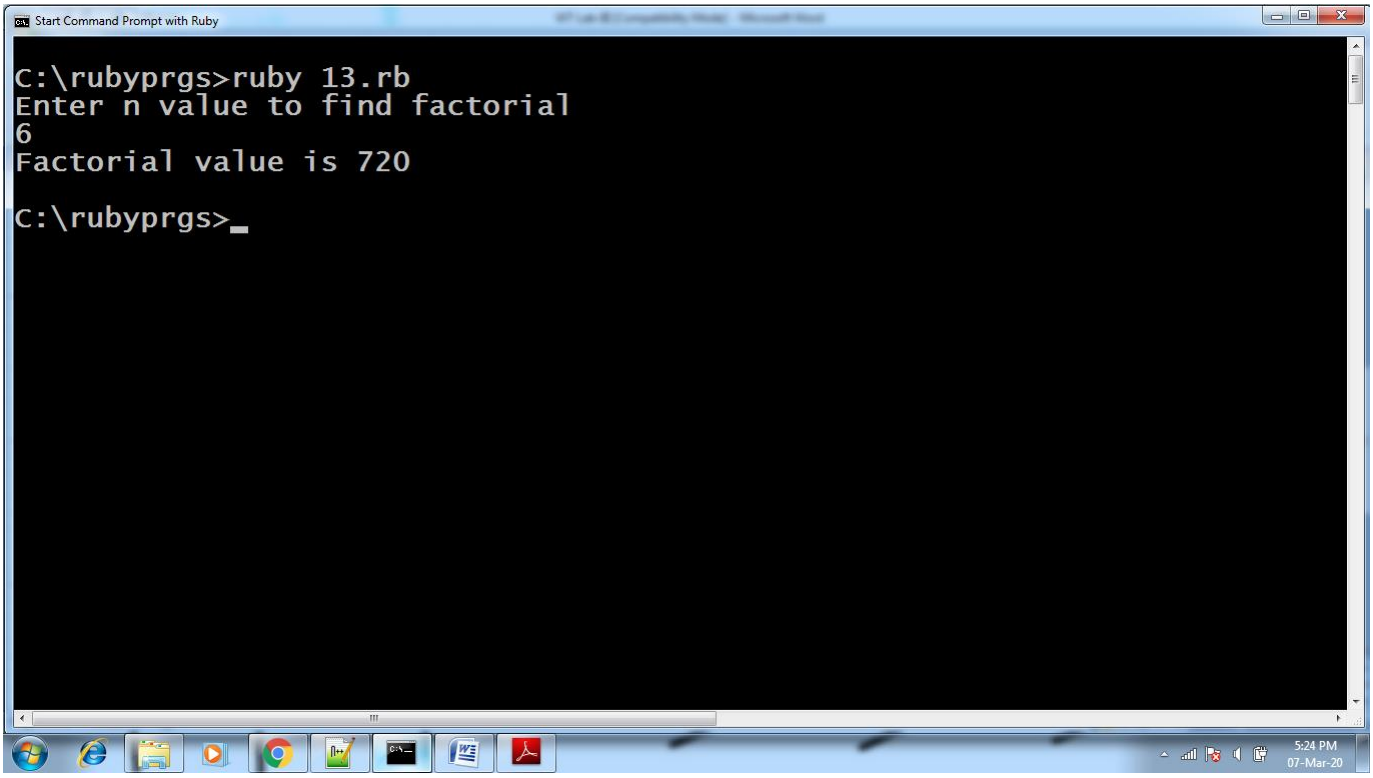
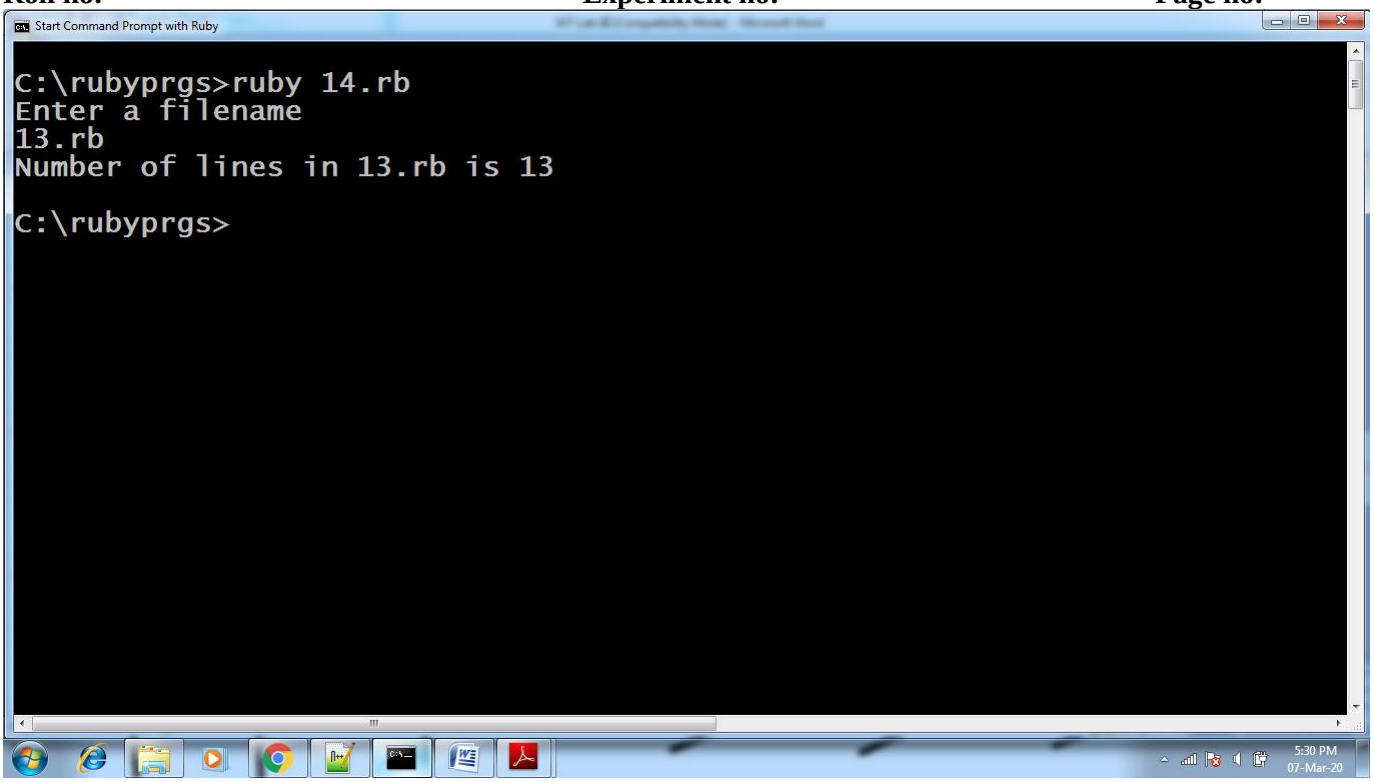
13. Write Ruby program reads a number and calculates the factorial value of it and prints the same.

```
#Ruby program to find factorial of a given number
puts "Enter n value to find factorial"
n=gets.to_i
fact=1
if n<=1
fact=1
else
        while n>=1
                fact=fact*n
                n-=1
        end
end
puts "Factorial value is #{fact}"
```



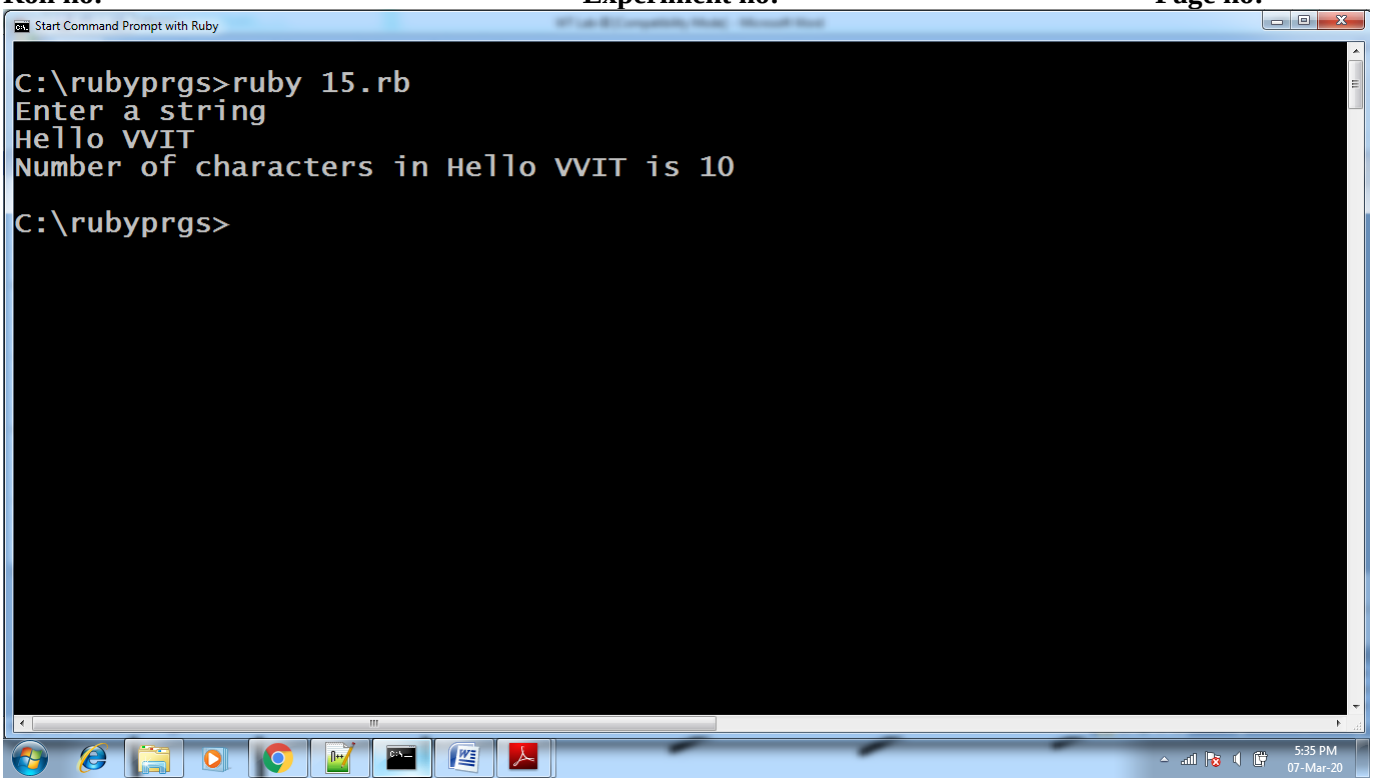
14. Write a Ruby program which counts number of lines in a text files using its regular expressions facility.

```
#Ruby program to find number of lines in a file
puts "Enter a filename"
fname=gets.chop
count=IO.readlines(fname).size
puts "Number of lines in #{fname} is #{count}"
```



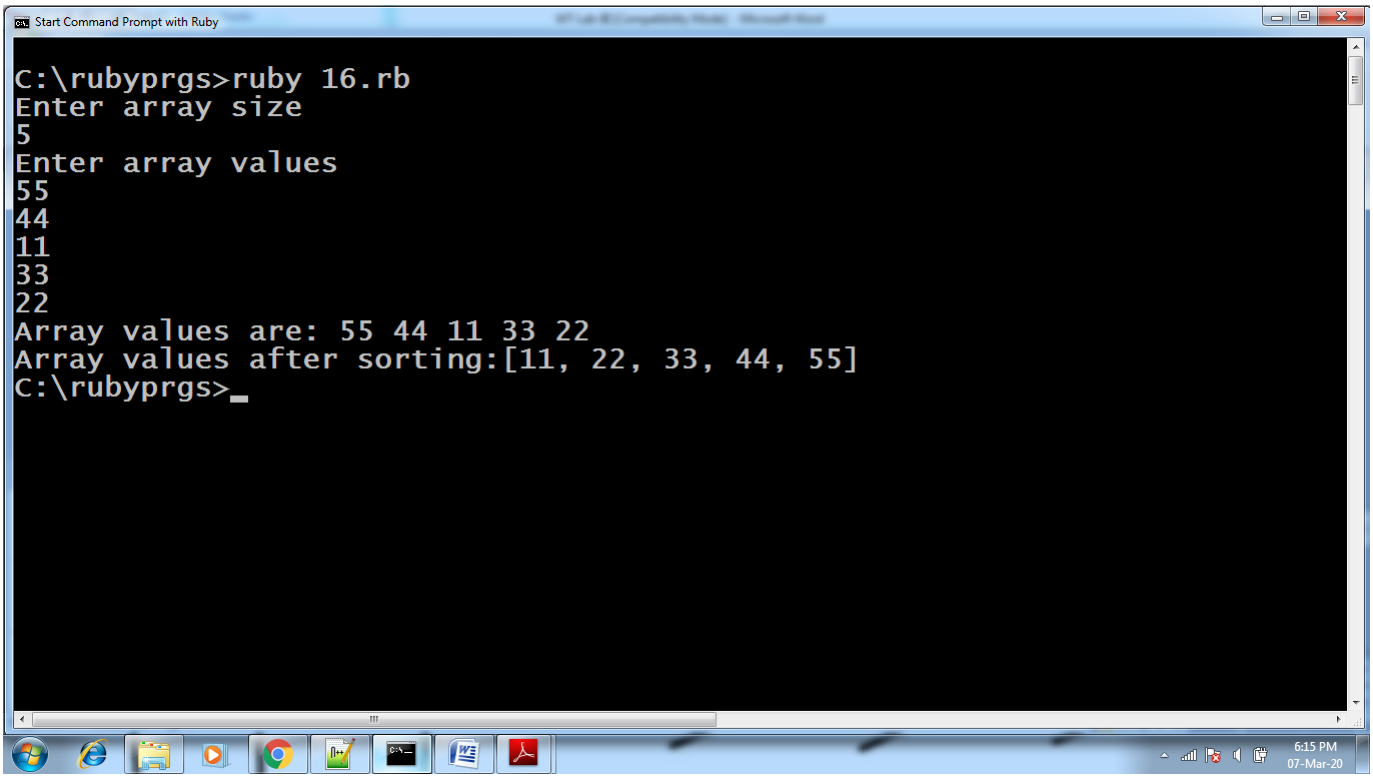
15. Write a Ruby program that uses iterator to find out the length of a string.

```
#Ruby program to find length of a string using iterator
puts "Enter a string"
str=gets.chop
count=0
str.each_char do |i|
        count=count+1
end
puts "Number of characters in #{str} is #{count}"
```



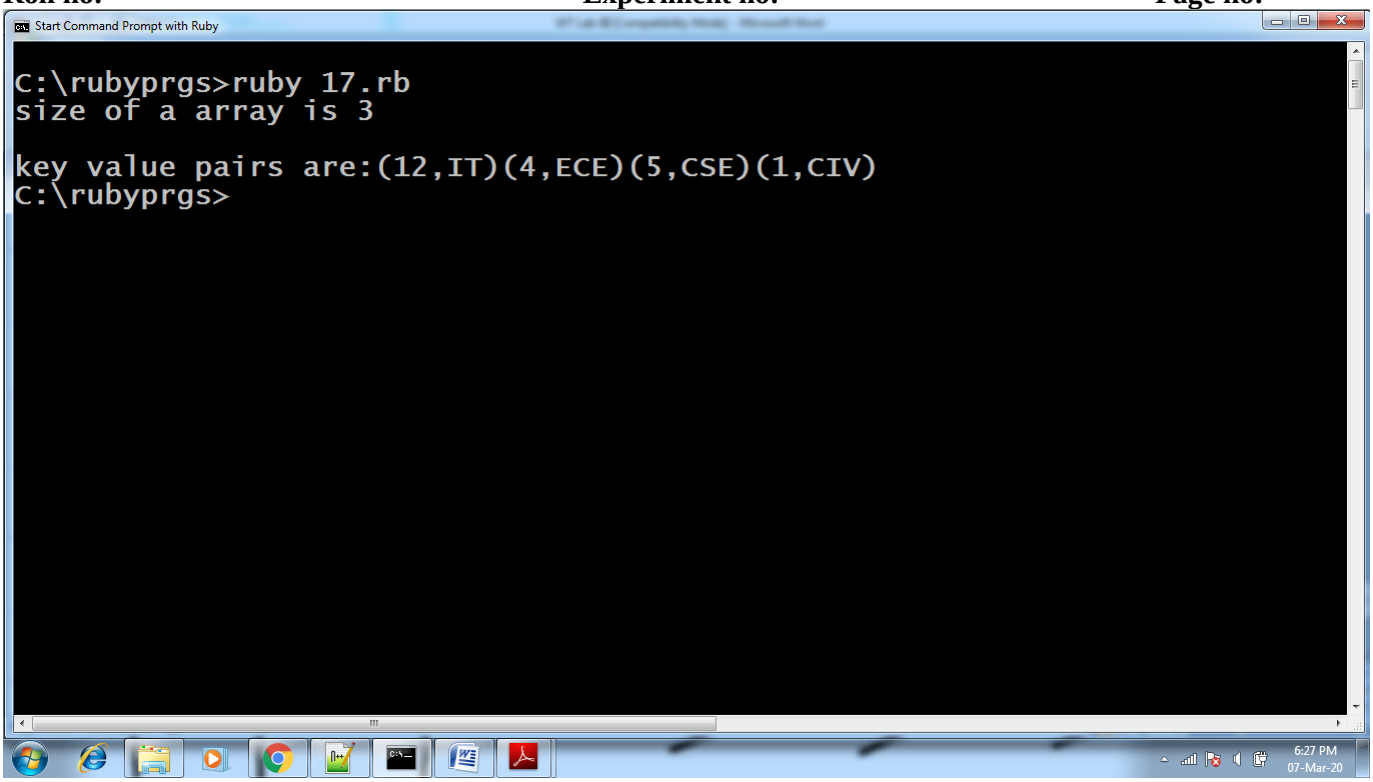
16. Write simple Ruby programs that uses arrays in Ruby.

```
#Ruby program to demonstrate working of Arrays
#Reading array size
puts "Enter array size"
n=gets.to_i
#Declaration of array
a=Array.new(n)
#Reading values into array
puts "Enter array values"
for i in 0..n-1
        a[i]=gets.to_i
end
#Displaying array values
print "Array values are:"
for i in 0..n-1
        print " #{a[i]}"
end
#Sorting array values
for i in 0..n-2
        for j in i+1..n-1
                if a[i]>a[j]
                        t=a[j]
                        a[j]=a[i]
                        a[i]=t
                end
        end
end
#Array values after sorting
print "\nArray values after sorting:#{a}"
```



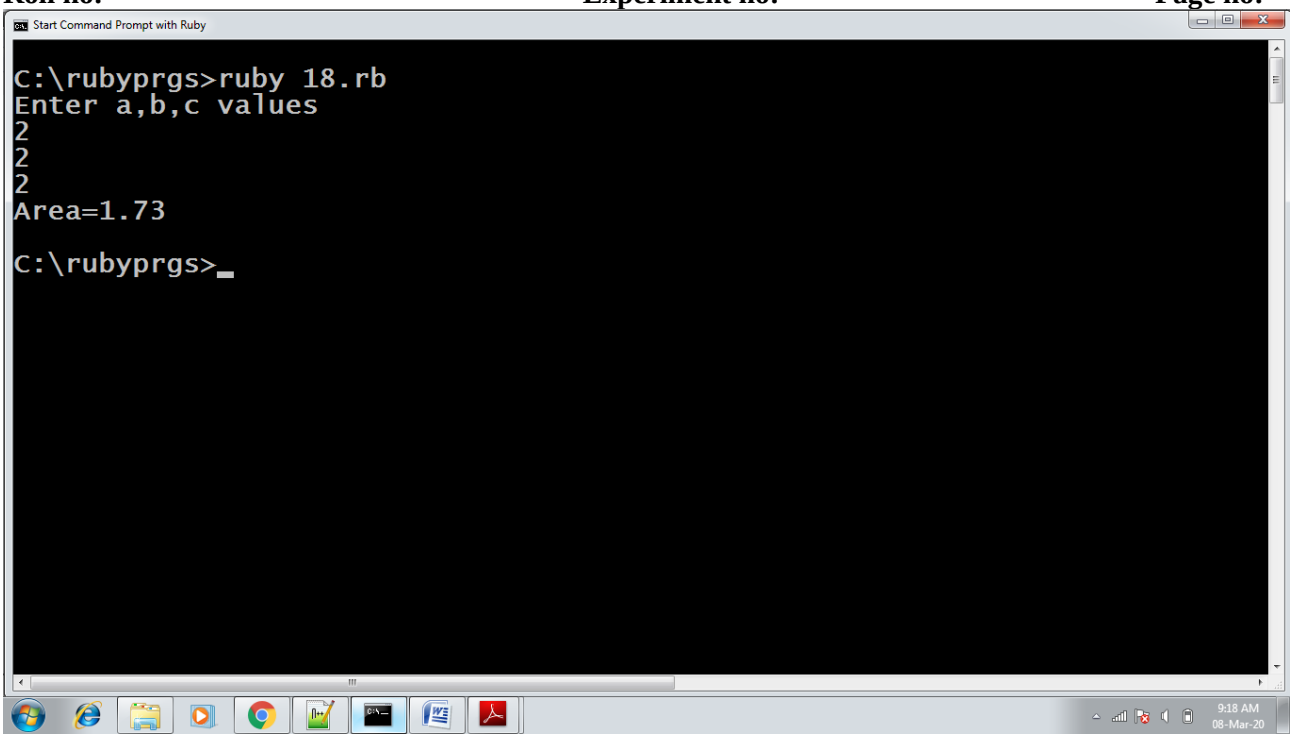
17. Write programs which uses associative arrays concept of Ruby.

```
#Ruby program to demonstrate working of associative arrays
#Declaring associative arrays
depts={12=>"IT",4=>"ECE",5=>"CSE"}
puts "size of a array is #{depts.length}"
#Adding new value into associative array
depts[1]="CIV"
#Displaing keys and values in associative array
print "\nkey value pairs are:"
depts.each do |key,val|
        print "(#{key},#{val})"
end
```



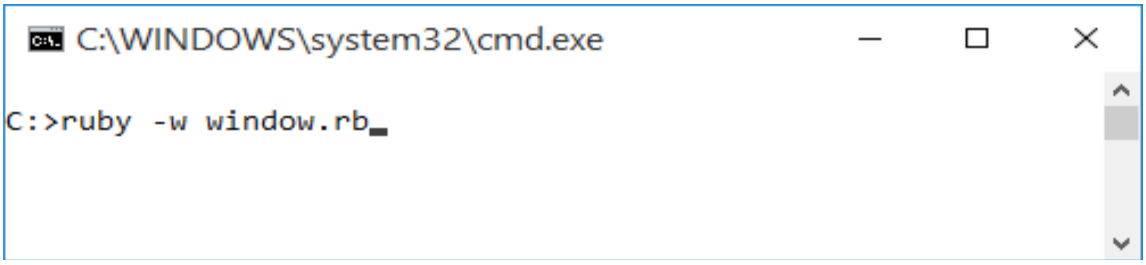
18. Write Ruby program which uses Math module to find area of a triangle.

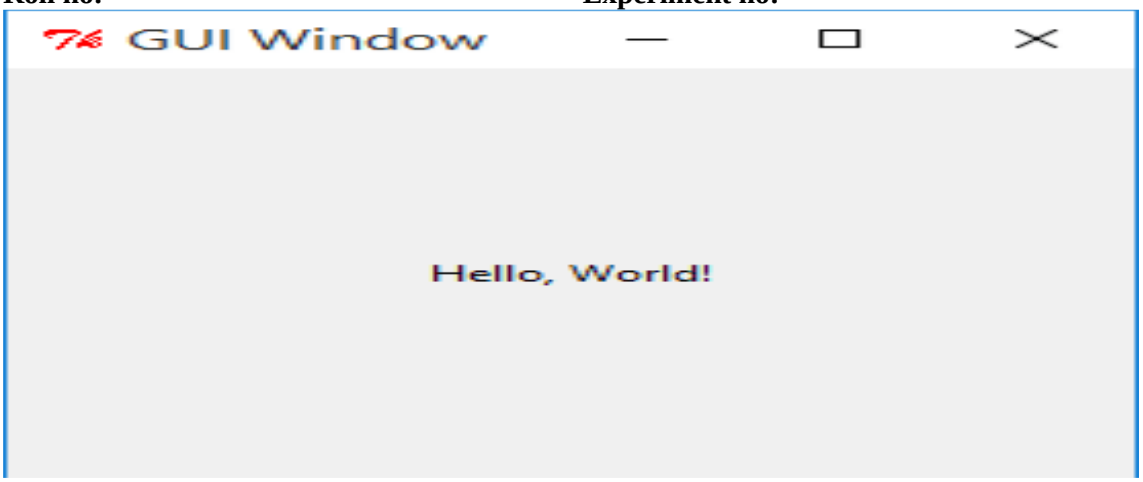
```
#Ruby program to find area of Triangle
puts "Enter a,b,c values"
a=gets.to_i
b=gets.to_i
c=gets.to_i
#perimeter of triangle
p=(a+b+c)/2
#Area of triangle
area=Math.sqrt(p*(p-a)*(p-b)*(p-c)).round(2)
puts "Area=#{area}"
```



19. Write Ruby program which uses tk module to display a window

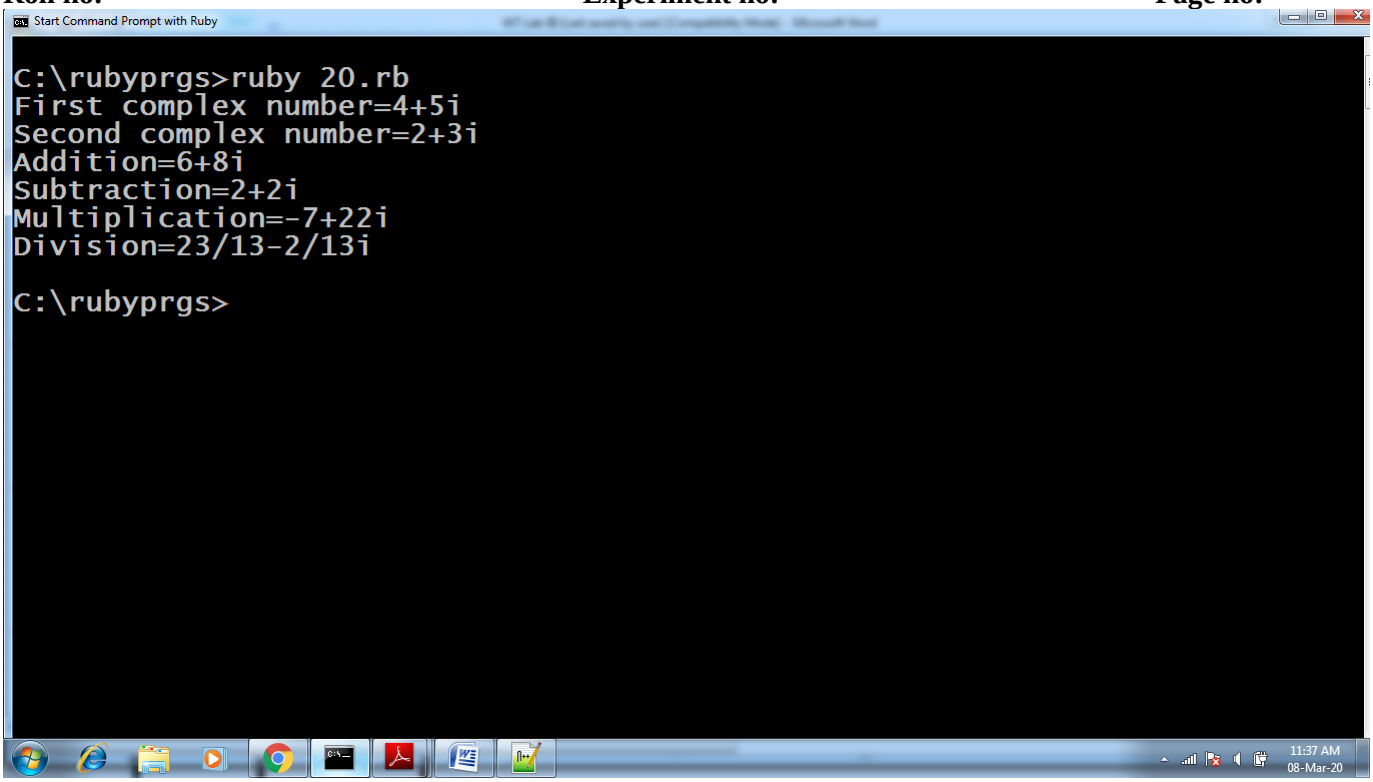
```
#Ruby program to demonstrate tk  module
root = TkRoot.new { title "GUI Window" }
TkLabel.new(root) do
text 'Hello, World!'
pack("side" => "right", "padx"=> "100", "pady"=> "100")
end
Tk.mainloop
```





20. Define Complex class in Ruby and do write methods to carry operations on Complex objects.

```
#Ruby program to define class complex
#Declaring first complex number
c1=Complex(4,5)
#Declaring second complex number
c2=Complex(2,3)
puts "First complex number=#{c1}"
puts "Second complex number=#{c2}"
#Addition of complex numbers
c3=Complex(0,0)
c3=c1+c2
puts "Addition=#{c3}"
#Subtraction of complex numbers
c3=Complex(0,0)
c3=c1-c2
puts "Subtraction=#{c3}"
#Multiplication of complex numbers
c3=Complex(0,0)
c3=c1*c2
puts "Multiplication=#{c3}"
#Division of complex numbers
c3=Complex(0,0)
c3=c1/c2
puts "Division=#{c3}"
```


Start Command Prompt with Ruby
C:\rubyprgs>ruby 20.rb
First complex number=4+5i
Second complex number=2+3i
Addition=6+8i
Subtraction=2+2i
Multiplication=-7+22i
Division=23/13-2/13i
C:\rubyprgs>
11:37 AM
08-Mar-20