ui22cs03-lab5

September 6, 2023

Q1) To determine whether the character entered is in lowercase, uppercase, digit or a special character.

[30]:
```
print("Program to Determine the Character cases:")
ch = input("Please Enter Your Own Character : ")
if(ch.isupper()):
    print("The Given Character ", ch, "is an Uppercase Alphabet")
elif(ch.islower()):
    print("The Given Character ", ch, "is a Lowercase Alphabet")
else:
    print("The Given Character ", ch, "is Not a Lower or Uppercase Alphabet")
```

```
Program to Determine the Character cases:
Please Enter Your Own Character : g
The Given Character  g is a Lowercase Alphabet
```

Q2) Find the roots of the quadratic equation

[9]:
```
print("Program to Calcualte the Roots of Quadratic equation")
print("A Quadratic Equation is in form of ax^2+bx+c ")
print('Enter the a, b and constant c below for calucation:-')
i=10
while i!=0:
  a= int(input("Enter the coefficient of x^2: "))
  b= int(input("Enter the coefficient of x: "))
  c = int(input("Enter the Constant c: "))
  D = ((b*b)-(4*a*c))
  x1= (-b+D)/(2*a)
  x2= (-b-D)/(2*a)
  print("The below is the Quadartic Equation:")
  print(a,"x^2+",b,"x+",c)
  if D==0:
    print("The Both Roots are same for above quadratic Equation is:",-b/(2*a))
    break
  elif D>0:
    print("The first quadratic root is: ",x1)
    print("The second quardatic root is: ",x2)
    break
  else:
```

```
    print("The above Quadaratic Equation have no real roots!")
    break
```

```
Program to Calcualte the Roots of Quadratic equation
A Quadratic Equation is in form of ax^2+bx+c
Enter the a, b and constant c below for calucation:-
Enter the coefficient of x^2: 1
Enter the coefficient of x: 3
Enter the Constant c: 6
The below is the Quadartic Equation:
1 x^2+ 3 x+ 6
The above Quadaratic Equation have no real roots!
```

Q3) Given three sides, write a program to check whether the triangle can be formed for the following conditions.

No triangle if a>= b+c

Right Angled triangle if a2=b2+c2

Obtuse triangle if a2>b2+c2

Acute triangle if a2<b2+c2

Equilateral triangle if all sides of the triangle are the same Isosceles triangle if two sides of the triangle are the same Scalene triangle otherwise

[18]:
```
print("Program to idenetify the Triangle")
a = int(input("Enter the first side of triangle:"))
b =int(input("Enter the second side of triangle"))
c = int(input("Enter the third side of triangle"))
a2=a*a
b2=b*b
c2=c*c
if a>=b+c or b>=a+c or c>= a+b:
  print("No Triangle Possible")
elif a2==b2+c2 or b2== a2+c2 or c2== a2+b2:
  print("It's a Right angled Triangle")
elif a2>b2+c2 or b2>c2+a2 or c2>a2+b2 :
  print("It's an Obtuse angled Triangle")
elif a2<b2+c2 or b2<c2+a2 or c2<a2+b2:
  print("It's an Acute angled Triangle")
elif a==b==c:
  print("It's an Equilateral Triangle")
elif a==b or b==c or a==c:
  print("It's an Isosceles Triangle")
else:
  print("It's Scalene Triangle")
```

```
Program to idenetify the Triangle
Enter the first side of triangle:4
```

```
Enter the second side of triangle3
Enter the third side of triangle5
It's a Right angled Triangle
```

Q4) 4. Write a menu-driven program to demonstrate the simple arithmetic calculator (if else)

[ ]:
```
print("Calculator Made by Aditya")
i=10
while(i!=0):
  print("Below are the folliowing operations:")
  print("1) Addition")
  print("2) Substraction")
  print("3) Multiplication")
  print("4) Division")
  print("5) Exit")
  i = int(input("Enter the operation to perform: "))
  a= int(input("Enter the first number :  "))
  b = int(input("Enter the second number : "))
  if i==1:
    print("Addition of",a,"and",b,"is : ",a+b)
  if i==2:
    print("Subtraction of",a,"and",b,"is : ",a-b)
  if i==3:
    print("Multiplication of",a,"and",b,"is : ",a*b)
  if i==4:
    print("Division of",a,"and",b,"is : ",a/b)
  if i==5:
    break
```

```
Calculator Made by Aditya
Below are the folliowing operations:
1) Addition
2) Substraction
3) Multiplication
4) Division
5) Exit
Enter the operation to perform: 1
Enter the first number :  5
Enter the second number : 6
Addition of 5 and 6 is :  11
Below are the folliowing operations:
1) Addition
2) Substraction
3) Multiplication
4) Division
5) Exit
Enter the operation to perform: 2
Enter the first number :  10
```

```
Enter the second number : 4
Subtraction of 10 and 4 is :  6
Below are the folliowing operations:
1) Addition
2) Substraction
3) Multiplication
4) Division
5) Exit
Enter the operation to perform: 3
Enter the first number :  5
Enter the second number : 8
Multiplication of 5 and 8 is :  40
Below are the folliowing operations:
1) Addition
2) Substraction
3) Multiplication
4) Division
5) Exit
Enter the operation to perform: 4
Enter the first number :  22
Enter the second number : 11
Division of 22 and 11 is :  2.0
Below are the folliowing operations:
1) Addition
2) Substraction
3) Multiplication
4) Division
5) Exit
```

Q5) Program to display the grade obtained by a student based on the marks. The relation between the grades and marks is – (switch)

[24]:
```python
marks = float(input("Enter the marks obtained: "))
grade_mapping = {
(0, 39.99): "Fail",
(40, 49.99): "E",
(50, 59.99): "D",
(60, 69.99): "C",
(70, 79.99): "B",
(80, 89.99): "A",
(90, 100): "A+"
}
grade = "Not Found"
for range_, g in grade_mapping.items():
  if range_[0] <= marks <= range_[1]:
    grade = g
    break
# Display the grade
```

```
print("Grade:", grade)
```

```
Enter the marks obtained: 5
Grade: Fail
```

Q6) Program to find the date on the next day if today's date is given. For example, 28/02/2013 next date is 01/03/2013. (else if ladder)

[29]:
```
# Input the date in the format DD/MM/YYYY
date_str = input("Enter a date (DD/MM/YYYY): ")
# Split the input date into day, month, and year
day, month, year = map(int, date_str.split('/'))
# Function to check if a year is a leap year
def is_leap_year(year):
  if (year % 4 == 0 and year % 100 != 0) or (year % 400 == 0):
    return True
  else:
    return False
# List of maximum days in each month
max_days_in_month = [0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30,
31]
# Check if it's a leap year and update February's maximum days
if is_leap_year(year):
  max_days_in_month[2] = 29
# Check if the input date is valid
if (month >= 1 and month <= 12) and (day >= 1 and day␣
 ↪<=max_days_in_month[month]):
  day += 1
# Check if the day exceeds the maximum for the month
  if day > max_days_in_month[month]:
    day = 1
    month += 1
# Check if the month exceeds 12
    if month > 12:
      month = 1
      year += 1

  next_date = f"{day:02d}/{month:02d}/{year:04d}"
  print("Next date:", next_date)
else:
    print("Invalid date entered.")
```

```
Enter a date (DD/MM/YYYY): 12/03/2003
Next date: 13/03/2003
```

[ ]: