CIS 1051 Introduction to Programming in Python **Summer II 2014**

Lecture 2: July 10

Lecturer: Anwar Mamat

Disclaimer: *These notes may be distributed outside this class only with the permission of the Instructor.*

2.1 If Else

Listing 1: If Else Example

```
def grade(g):
    if(g >= 90):
        print('A')
    elif(g>=80):
        print('B')
    elif(g >= 70):
        print('C')
    else:
        print('D')
def main():
    g = 60
    grade(g)

main()
```

2.2 Loops

Listing 2: Factorial

```
#!/usr/bin/local/python3.4
#n!= n * (n-1) * (n-2)*... * 1
def fact(n):
    p = 1
    i = 1
    while(i <= n):
        p *= i
        i += 1
    return p
def main():
    f = int(input("Enter␣a␣number:"))
    print(fact(f))
main()
```

Listing 3: Multiplication Table

```
#prints the 9*9 multiplication table
def main():
    for i in range(1,10):
        for j in range(1,i+1):
            print(i*j,end='\t')
        print()
main()
```

2.2.1 Wallis Formula

$$\prod_{n=1}^{\infty}\left(\frac{2n}{2n-1}\cdot\frac{2n}{2n+1}\right)=\frac{2}{1}\cdot\frac{2}{3}\cdot\frac{4}{3}\cdot\frac{4}{5}\cdot\frac{6}{5}\cdot\frac{6}{7}\cdot\frac{8}{7}\cdot\frac{8}{9}\cdots=\frac{\pi}{2}$$

Figure 2.1: Python turtle output

Listing 4: Wallis Formula

```
# Calculate pi using Wallis Formula
def wallis(l):
    p = 2
    for i in range(3,l,2):
        p *= (i-1)/i * (i+1)/i
    return p * 2
def main():
    print(wallis(1000000))
main()
```

2.2.2 Calculate Square root

IN this program, we calculate the square root of a number using Newton's method.

$$x_{n+1}=x_n-\frac{f(x_n)}{f'(x_n)}=x_n-\frac{x_n^2-S}{2x_n}=\frac{1}{2}\left(x_n+\frac{S}{x_n}\right)$$

Figure 2.2: Sine Curve

Listing 5: Square Root

```
#x = (N/x+x)/2
import math
def sqrt2(n):
    root_old=0
    root = 1
    count = 0
    root = (n / root + root) / 2
    while(math.fabs(root-root_old) > 0.001):
        root_old = root
        root = (n / root + root) / 2
        count +=1
    print(count)
    return root
def main():
    n = float(input("Enter a number:"))
    s =round(sqrt2(n),3)
    print('Square root of {} = {}'.format(n,s))
main()
```

2.2.3 Year Calendar

Listing 6: Year Caldendar

```
def calendar():
    days2=[31,28,31,30,31,30,31,31,30,31,30,31]
    #days=(31,28,31,30,31,30,31,31,30,31,30,31)
    months=['January','February','March','April','May','June','July',
    'August','September', 'October','November','December']
    pos = 3
    for month in range(12):
        print(months[month])
        print('------------------------------------------------------------')
        print('S\tM\tT\tW\tT\tF\tS')
        for j in range(pos):    #skip to correct position
            print('\t',end='')
        for day in range(1,days[month]+1):
            print(day,end='\t')
            print('|',end='')
            pos +=1
            if(pos >= 7):        //7 days in a week
                print()
                pos=0
        print('\n------------------------------------------------------------')
def main():
    calendar()
main()
```

2.3 Turtle Graphics

Listing 7: Draw a polygon

```
import turtle
t = turtle.Turtle()
def drawpolygon(size, side):
    t.up()
    t.backward(100)
    t.down()
    t.color(’red’)
    for i in range(side):
        t.forward(size)
        t.left(360/side)
def main():
    drawpolygon(100,8)// octagon
    drawpolygon(100,5)// pentagon
    turtle.exitonclick()
main()
```

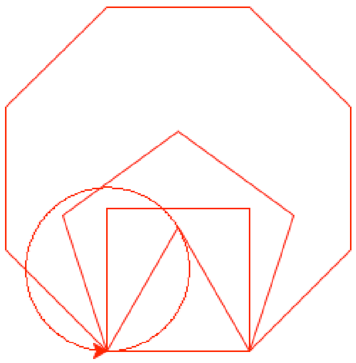

Figure 2.3: Python turtle output

Listing 8: Sine Curve

```
import turtle
import math
t = turtle.Turtle()

def line(x1,y1,x2,y2):
    t.up()
    t.goto(x1,y1)
    t.down()
    t.goto(x2,y2)

def axes():
    line(-180,0,180,0)
    line(0,200,0,-200)

def sine():
    t.up()
    t.goto(-180,0)
    t.down()
    for i in range(360):
        y=math.sin(math.radians(i))
        t.goto(i-180,y*100)
    t.up()
    t.goto(-10,-50)
    t.write("y=sin(x)", True, "left", "Arial␣24")
axes()
sine()
turtle.exitonclick()
```

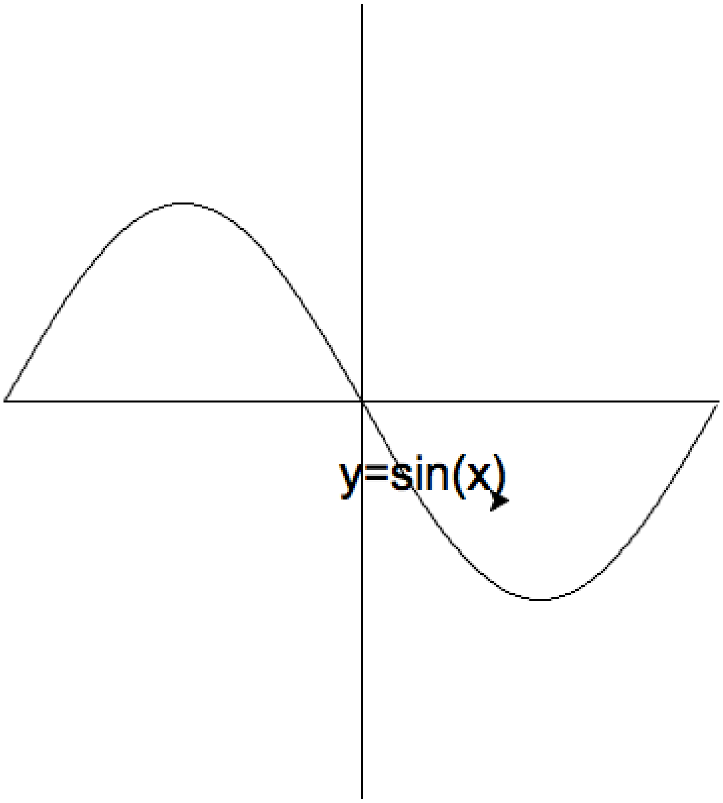

Figure 2.4: Sine Curve

Listing 9: Parametric Curve

```
import turtle
import math
t = turtle.Turtle()

```

```
#draw a line between two points (x1,y1) and (x2,y2)
def line(x1,y1,x2,y2):
    t.up()
    t.goto(x1,y1)
    t.down()
    t.goto(x2,y2)
def axes():
    t.color('blue')
    line(-300, 0 , 200, 0)
    line(-150, -200, -150, 200)

def plot():
    t.up()
    t.goto(-150,0)
    t.down()
    t.color('red')
    for i in range(360*5):
        x= math.radians(i) * math.cos(math.radians(i))
        y = math.radians(i) * math.sin(math.radians(i))
        t.goto(x*5-150,y*5)
def main():
    axes()
    plot()
    turtle.exitonclick()
main()
```

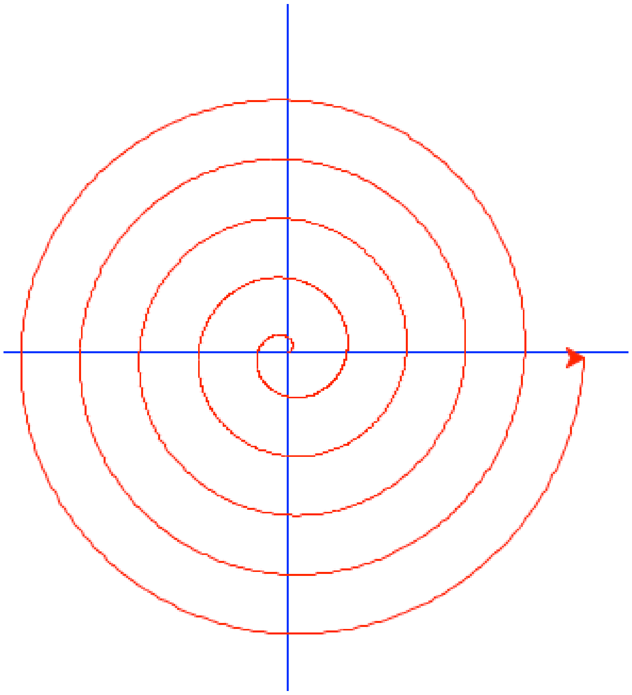

Figure 2.5: Parametric Curve

2.4 List

Listing 10: List Example

```

def grade():
    grades = [] #empty list
    sum = 0
    n = int(input(”Enter␣the␣number␣of␣students:”))
    for i in range(n):
        g = int(input(”enter␣the␣grade”))
        grades.append(g) #add to the list
        #sum += g
    print(grades)
    for i in range(n):
        sum += grades[i]
    avg = sum /n
    print(avg)

grade()
```