Question 3

```
x \in [a,b]
Number of Intervals : n
Interval Size : rac{b-a}{n}I = \int_a^b \sin x dx
```

Exact solution of the above integral is 2

```
In [1]:
        import numpy as np
        def WriteDataToFile(list,myfile):
            myfile.write(', '.join(str(item) for item in list)+'\n')
        F = lambda x: np.sin(x)
        I_exact = 2.
        xb = np.pi
        xa = 0.
        def TrapezoidalRuleIntegration(n):
            x = np.linspace(xa,xb,n+1) ##(n + 1)Plus one because linspace creates n point i
            5=0
            [s:=s+2*F(i) for i in x]
            return (s-F(xb)-F(xa))*0.5*(xb-xa)/n
        def SimpsonRuleIntegration(n):
            x = np.linspace(xa,xb,n+1) ##(n + 1)Plus one because linspace creates n point i
            s=F(xa)+F(xb)
            [s:=s+4*F(x[i]) for i in range(1,len(x)-1,2)]
            [s:=s+2*F(x[i]) for i in range(2,len(x)-1,2)]
            return s*(xb-xa)/n/3
        myfile = open("output data\\trap.csv", 'a')
        myfile.seek(0)
        myfile.truncate()
        WriteDataToFile(["n","h","EstimateIntegral","eh","e2h/eh"],myfile)
        firstiter = TrapezoidalRuleIntegration(1)
        WriteDataToFile([1,(xb-xa),firstiter,-firstiter+I_exact,"-"],myfile)
        n = np.array([2**i for i in range(0,8)])
        error_vec = np.zeros(len(n))
        for i in n[1:]:
            T_h = TrapezoidalRuleIntegration(i)
            eh = abs(T_h - I_exact)
            np.append(error_vec,eh)
```

```
eh2 = abs(firstiter - I_exact)
   firstiter = T_h
   WriteDataToFile([i,(xb-xa)/i,T_h,eh,eh2/eh],myfile)
myfile.close()
myfile = open("output_data\\SimpingForNami.csv", 'a')
myfile.seek(0)
myfile.truncate()
WriteDataToFile(["n","h","EstimateIntegral","eh","e2h/eh"],myfile)
firstiter = SimpsonRuleIntegration(1)
WriteDataToFile([1,(xb-xa),firstiter,-firstiter+I_exact,"-"],myfile)
n = np.array([2**i for i in range(0,8)])
error_vec = np.zeros(len(n))
for i in n[1:]:
   T_h = SimpsonRuleIntegration(i)
   eh = abs(T_h - I_exact)
   np.append(error_vec,eh)
   eh2 = abs(firstiter - I_exact)
   firstiter = T_h
   WriteDataToFile([i,(xb-xa)/i,T_h,eh,eh2/eh],myfile)
myfile.close()
```

Results

Part(a)

n	h	EstimateIntegral	eh	e2h/eh
1	3.141592653589793	1.9236706937217898e- 16	1.99999999999998	-
2	1.5707963267948966	1.570796326794897	0.429203673205103	4.659792366325491
4	0.7853981633974483	1.8961188979370398	0.1038811020629602	4.131681939078501
8	0.39269908169872414	1.9742316019455508	0.025768398054449193	4.031337215586981
16	0.19634954084936207	1.9935703437723395	0.006429656227660452	4.007741182739018
32	0.09817477042468103	1.998393360970144	0.0016066390298559163	4.001929561139234
64	0.04908738521234052	1.9995983886400375	0.00040161135996252817	4.000482033192044
128	0.02454369260617026	1.9998996001842038	0.00010039981579623714	4.000120486053522

As seen our value of $\frac{e_{2h}}{e_h}$ approaches to **4**. That actually makes sence because order of convergence for Trapozoidal method of integration is 2.

$$Global\ Error = rac{(a-b)h^2f"(\xi)}{12}$$

Beside that Degree of Precesion for Trapozoidal rule is 1

Part (b)

n	h	EstimateIntegral	eh	e2h/eh
1	3.141592653589793	1.2824471291478598e- 16	1.99999999999998	-
2	1.5707963267948966	2.0943951023931953	0.09439510239319526	21.18753991779319
4	0.7853981633974483	2.0045597549844207	0.0045597549844207386	20.701792687482968
8	0.39269908169872414	2.000269169948388	0.0002691699483881038	16.940059660175127
16	0.19634954084936207	2.0000165910479355	1.6591047935499148e- 05	16.223806322213832
32	0.09817477042468103	2.0000010333694127	1.0333694127062643e- 06	16.05529226189237
64	0.04908738521234052	2.0000000645300013	6.453000134243325e-08	16.01378260047776
128	0.02454369260617026	2.0000000040322576	4.032257638897363e-09	16.00344202214203

As seen our value of $\frac{e_{2h}}{e_h}$ approaches to **16**. That actually makes sence because order of convergance for Simpson method of integration is 4.

$$Global\ Error = rac{(a-b)h^4f""(\xi)}{2880}$$

Beside that Degree of Precesion for Simpson rule is 3

Part (c)

From above to analysis it is very evident that convergance condition can be checked by ratio $\frac{e_{2h}}{e_h}$, In case we are not aware of the exact definate integral we can just put stopping condition on that ration

Stopping condition

$$rac{e_{2h}}{e_h}
ightarrow 2^{order}$$

X	Trapezoidal Rule	SipmpsonRule
Order	2	4