

```

import timeit
import random
import matplotlib.pyplot as plt

#input array elements
def Input(Array, n):
    #iterating till the range
    for i in range(0, n):
        ele = random.randrange(1, 50)
        #adding the elements
        Array.append(ele)

#selection sort
def selectionSort(Array, size):
    for ind in range(size):
        min_index = ind
        for j in range(ind + 1, size):
            #select the minimum element in every iteration
            if Array[j] < Array[min_index]:
                min_index = j
        #swapping the elements to sort the array
        (Array[ind], Array[min_index]) = (Array[min_index], Array[ind])

#main block
N = []
CPU = []
trail = int(input("Enter no. of trails: "))
for t in range(0, trail):
    Array = []
    print("-----> TRAIL NO:", t + 1)
    n = int(input("Enter number of elements: "))
    Input(Array, n)
    start = timeit.default_timer()
    selectionSort(Array, n)
    end = timeit.default_timer()
    print("Sorted Array:")
    print(Array)
    N.append(n)

```

```
CPU.append(round(float(end - start)*1000000, 2))

print("N CPU")
for t in range(0, trail):
    print(N[t], CPU[t])

#plotting Graph

plt.plot(N, CPU)
plt.scatter(N, CPU, color = "red", marker="*", s = 50)
#name the x axis
plt.xlabel('Array Size - N')
#naming the y axis
plt.ylabel('CPU Processing Time')
#giving title to the graph
plt.title('Selection Sort Time Efficiency')
#function to show the plot
plt.show()
```