**Sardar Patel Institute of Technology**

Bhavan’s Campus, Munshi Nagar, Andheri (West), Mumbai-400058-India

(Autonomous College Affiliated to University of Mumbai)

|  |  |
| --- | --- |
| **PRACTICAL NO. 1-B** | **DESIGN AND ANALYSIS OF ALORITHMS** |
| **NAME** | **VEDANTI ANIL WADATKAR** |
| **UID** | **2021700072** |
| **BATCH** | **D4** |
| PROBLEM STATEMENT | For this experiment, you need to implement two sorting algorithms namely  Insertion and Selection sort methods. Compare these algorithms based on time and space complexity. Time required  to sorting algorithms can be performed using high\_resolution\_clock::now() under namespace std::chrono.  You have togenerate1,00,000 integer numbers using C/C++ Rand function and save them in a text file. Both  the sorting algorithms uses these 1,00,000 integer numbers as input as follows. Each sorting algorithm sorts a block  of 100 integers numbers with array indexes numbers A[0..99], A[0..199], A[0..299],..., A[0..99999]. You need to use  high\_resolution\_clock::now() function to find the time required for 100, 200, 300.... 100000 integer numbers. Finally,  compare two algorithms namely Insertion and Selection by plotting the time required to sort 100000 integers using  LibreOffice Calc/MS Excel. The x-axis of 2-D plot represents the block no. of 1000 blocks. The y-axis of 2-D plot  representsthe tunning time to sort 1000 blocks of 100,200,300,...,100000 integer numbers. |
|  | **Series1- selection sort**  **Series 2 - insertion sort** |
| **Conclusion** | Among both of the sorting algorithm, the insertion sort is fast, efficient, stable while selection sort only works efficiently when the small set of elements is involved or the list is partially previously sorted. |