NAME: SUSHANT GAWADE

ROLL NO.: 118

BATCH: B3

ASSIGNMENT 1

CODE:

```
#include <bits/stdc++.h>
#include <iostream>
#include <stdlib.h>
#include <time.h>
#include <algorithm>
#include <chrono>
using namespace std;
using namespace std::chrono;

long int binary_search(long long int arr1[], long long int minm, long long int maxm, long long int num)
{
   long long int arr2[maxm+1];
   for(int i=0;i<maxm+1;i++){
      arr2[i]  = arr1[i];
   }
   auto start = chrono::high_resolution_clock::now();
   sort(arr2,arr2+(maxm+1));
   int mid;
   if (maxm >= minm)
   {
      mid = (minm + maxm) / 2;
```

```
        if (arr2[mid] == num)
        {
            cout << "\n KEY ELEMENT FOUND at INDEX: " << mid<<endl;
            auto stop = chrono::high_resolution_clock::now();

            double duration = chrono::duration_cast<chrono::nanoseconds>(stop - start).count();

            duration *= 1e-9;

            cout << " Time taken by Binary Search: " << fixed << duration << setprecision(9) << "
microseconds" << endl;
        }
        else if (arr2[mid] > num)
        {
            return binary_search(arr2, minm, mid - 1, num);
        }
        else
            return binary_search(arr2, mid + 1, maxm, num);
    }
    else
        cout << "\n KEY ELEMENT NOT FOUND";
}

void linear_search(long long int arr[], long long int n, long long int key)
{
    auto start = chrono::high_resolution_clock::now();

    int temp = 0;
    for (int i = 0; i < n; i++)
    {
```

```
        if (arr[i] == key)
        {
            cout << "\n KEY ELEMENT FOUND at INDEX " << i << endl;
            temp = 1;
            break;
        }
    }
    if (temp == 0)
    {
        cout << "\n KEY ELEMENT NOT FOUND";
    }
    else
    {
        auto stop = chrono::high_resolution_clock::now();

        double duration = chrono::duration_cast<chrono::nanoseconds>(stop - start).count();

        duration *= 1e-9;

        cout << " Time taken by Linear Search: " << fixed << duration << setprecision(9) << "
microseconds" << endl;
    }
}

int main()
{
    srand(time(0));

    long long int n;
```

```
//n = rand();

cout<<"\nEnter the size of array :";
cin>>n;

long long int arr[n] = {0};

for (int i = 0; i < n; i++)
{
   arr[i] = rand();
}

cout << "\n Your Random Array is: \n";

for (int i = 0; i < n; i++)
{
   cout << " " << arr[i];
}

int k = 0;

while (k != 10)
{
   k++;

   long long int key ;

   cout<<"\n\n Enter Key Element: ";
   cin>>key;
```

```
        cout<<"\n****************************************";

        cout<<"\n\n USING LINEAR SEARCH";

        linear_search(arr, n, key);

        cout<<"\n\n USING BINARY SEARCH";

        int result = binary_search(arr, 0, n - 1, key);

        if(result==-1){
           cout << "\n KEY ELEMENT FOUND" << endl;
        }
        cout<<"\n****************************************\n\n";
    }
}
```

Output Clear

```
/tmp/AFubGHGxYt.o
Enter the size of array :5
Your Random Array is:
 855977957 1031746772 1840688927 612786650 1702697909

 Enter Key Element: 1840688927
 ****************************************

 USING LINEAR SEARCH
 KEY ELEMENT FOUND at INDEX 2
 Time taken by Linear Search: 0.000016 microseconds


 USING BINARY SEARCH
 KEY ELEMENT FOUND at INDEX: 4
 Time taken by Binary Search: 0.000005462 microseconds

****************************************



 Enter Key Element:
```