ASSIGNMENT-9 SOLUTION

1. What is the worst case complexity of selection sort?
   a) O(nlogn)
   b) O(logn)
   c) O(n)
   d) $O(n^2)$

Solution: (d) $O(n^2)$

Selection sort creates a sub-list, LHS of the 'min' element is already sorted and RHS is yet to be sorted. Starting with the first element the 'min' element moves towards the final element

2. What is the correct order of insertion sort (in ascending order) of the array arr[5]={8 3 5 9 4}?
   a) {3 8 5 9 4}→ {3 5 8 9 4}→ {3 5 8 9 4}→ {3 4 5 8 9}
   b) {3 8 5 9 4}→ {3 5 8 9 4}→ {3 5 8 4 9}→ {3 5 4 8 9}→{3 4 5 8 9}
   c) {3 8 5 9 4}→{3 4 8 5 9}→{3 4 5 8 9}→{3 4 5 8 9}→{3 4 5 8 9}
   d) {8 3 5 4 9}→{8 3 4 5 9}→{3 4 5 8 9}

Solution: (a) In insertion sort, we start from the second number onwards and place it in appropriate position before its current position. Thus, the correct answer is (a).

3. What is the error in the following program?

```
#include<stdio.h>
#define SI(p, n, r) float si; si=p*n*r/100;
int main()
{
   float p=2500, r=3.5, a;
   int n=3;
   a=SI(p, n, r);
   SI(1500, 2, 2.5);
   printf("%f", a);
 return 0;
}
```

   a) Variable 'a' should be replaced by 'si'
   b) Nothing will be printed
   c) Error due to multiple declaration of si
   d) No error

Solution: (c) The macro #define SI(p, n, r) float si; si=p*n*r/100; contains the error. To remove this error, we have to modify this macro to #define SI(p,n,r) p*n*r/100 and declare si as float in the main program.

4. When the Binary search is best applied to an array?
   a) when the array size is large.

b) when the array is in sorted order.
c) when the array elements are mixed data type.
d) when the array is unsorted.

Solution: (b) Binary search is applied for sorted array.

5. Linear searching is generally used?
   a) when the list has only a few elements
   b) when performing a single search in an unordered list
   c) Both (a) and (b)

Solution: (c) Both a and b

It is practical to implement linear search in the situations mentioned in a and b, but for larger elements the complexity becomes larger and it makes sense to sort the list and employ binary search or other methods.

6. Select the code snippet which performs unordered linear search iteratively?

   a)
```
int unorderedLinearSearch(int arr[], int size, int data)
{
   int index;
   for(int i = 0; i < size; i++)
   {
      if(arr[i] == data)
      {
         index = i;
         break;
      }
   }
   return index;
}
```

   b)
```
int unorderedLinearSearch(int arr[], int size, int data)
{
   int index;
   for(int i = 0; i < size; i++)
   {
      if(arr[i] == data)
      {
         break;
      }
   }
   return index;
}
```

   c)
```
int unorderedLinearSearch(int arr[], int size, int data)
```

```
{
   int index;
   for(int i = 0; i <= size; i++)
   {
      if(arr[i] == data)
      {
         index = i;
         continue;
      }
   }
   return index;
}
```

d) None of the above

Solution: (a)

Unordered term refers to the given array, that is, the elements need not be ordered. To search for an element in such an array, we need to loop through the elements until the desired element is found.

7. What is the best case for linear search?
   a) O(log n)
   b) O(n log n)
   c) O(1)
   d) O(n)

Solution: (c) O(1)

This happens when the element is at the head of the array, hence O(1).

8. What is the best case and worst case complexity of ordered linear search?
   a) O(nlogn), O(logn)
   b) O(logn), O(nlogn)
   c) O(n), O(1)
   d) O(1), O(n)

Solution: (d)

Although ordered linear search is better than unordered when the element is not present in the array, the best and worst cases still remain the same, with the key element being found at first position or at last position respectively.

9. Which of the following is a disadvantage of linear search?
   a) Requires more space
   b) Greater time complexities compared to other searching algorithms
   c) Not easy to understand

d) All of the mentioned

Solution: (b)

The complexity of linear search as the name suggests is O(n) which is much greater than other searching techniques like binary search(O(logn)).

10. Binary Search can be categorized into which of the following?
    a) Brute Force technique
    b) Divide and conquer
    c) Greedy algorithm
    d) Dynamic programming

Solution: (b) Divide and conquer

Since 'mid' is calculated for every iteration or recursion, we are diving the array into half and then try to solve the problem.

11. Given an array arr = {45, 77, 89, 91, 94, 98,100} and key = 100; what are the mid values (corresponding array elements) generated in the first and second iterations?
    a) 91 and 98
    b) 91 and 100
    c) 89 and 94
    d) 94 and 98

Solution: (a) 91 and 98

12. Select the appropriate pseudocode that performs selection sort

a)
```
int min;
    for(int j=0; j<arr.length-1; j++)
    {
        min = j;
        for(int k=j+1; k<=arr.length-1; k++)
        {
            if(arr[k] < arr[min])
                min = k;
        }
        int temp = arr[min];
        arr[min] = arr[j];
        arr[j] = temp;
}
```

b)
```
int min;
  for(int j=0; j<arr.length-1; j++)
```

```
{
        min = j;
        for(int k=j+1; k<=arr.length; k++)
        {
                if(arr[k] < arr[min])
                        min = k;
        }
        int temp = arr[min];
        arr[min] = arr[j];
        arr[j] = temp;
}
```

c)

```
int min;
for(int j=0; j<arr.length-1; j++)
{
        min = j;
        for(int k=j+1; k<=arr.length-1; k++)
        {
                if(arr[k] > arr[min])
                        min = k;
        }
        int temp = arr[min];
        arr[min] = arr[j];
        arr[j] = temp;
}
```

d)

```
int min;
for(int j=0; j<arr.length-1; j++)
{
        min = j;
        for(int k=j+1; k<=arr.length; k++)
        {
                if(arr[k] > arr[min])
                        min = k;
        }
        int temp = arr[min];
        arr[min] = arr[j];
        arr[j] = temp;
}
```

Solution: (a)
Starting with the first element as ‘min’ element, selection sort loops through the list to select the least element which is then swapped with the ‘min’ element.

13. Consider the array A[]= {5,4,9,1,3} apply the insertion sort to sort the array . Consider the cost associated with each sort is 25 rupees, what is the total cost of the insertion sort when element 1 reaches the first position of the array?
    a) 25
    b) 50
    c) 75
    d) 100

    Solution: (b)

    When the element 1 reaches the first position of the array two comparisons are only required hence 25 * 2= 50 rupees.

    *step 1: 4 5 9 1 3.

    *step 2: 1 4 5 9 3

14. The average case occurs in the Linear Search Algorithm when
    a) The item to be searched is in somewhere middle of the Array
    b) The item to be searched is not in the array
    c) The item to be searched is in the last of the array
    d) The item to be searched is either in the last or not in the array

Solution: (a)

- The average case occurs in the Linear Search Algorithm when the item to be searched is in some where middle of the Array.
- The best case occurs in the Linear Search Algorithm when the item to be searched is in starting of the Array.
- The worst case occurs in the Linear Search Algorithm when the item to be searched is in end of the Array.

So, option (a) is correct.

15. Find the output of the following C program

```
#include<stdio.h>
int main()
{
        int a;
        int arr[5] = {1, 2, 3, 4, 5};
        arr[1] = ++arr[1];
        a = arr[1]++;
        arr[1] = arr[a++];
        printf("%d,%d", a, arr[1]);
        return 0;
}
```

    a) 5,4
    b) 5,5
    c) 4,4

d) 3,4

Solution: (c)