

Week 9

1. #include <somefile.h> are _____ files and #include "somefile.h" _____ files.
- Library, Library
 - Library, user-created header
 - User-created header, library
 - They can include all types of file

Solution: (d) Both of these statement can be used to select any file.

2. Which of the following are C preprocessors?
- #ifdef
 - #define
 - #endif
 - all of the mentioned

Solution: (d)

3. If the file name is enclosed in double quotation marks
- The preprocessor treats it as a user-defined file
 - The preprocessor treats it as a system-defined file
 - The preprocessor treats it as a user-defined file & system-defined file
 - None of the mentioned

Solution: (a)

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

Solution: (a) I

5. What is the error in the following program?

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

- Variable 'a' should be replaced by 'si'
- Nothing will print
- Error in declaration of si
- 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

6. When the Binary search is best applied to an array?
- For very large size array
 - When the array is sorted
 - When the array elements are mixed data type
 - When the array is unsorted

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

7. Linear searching is used?

- a) When the list has only a few elements
- b) When performing a single search in an unordered list
- c) Used all the time
- d) Both a and b

Solution: (d) 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.

8. 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.

9. 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 $O(1)$.

10. What is the worst case for linear search?

- a) $O(\log n)$
- b) $O(n \log n)$
- c) $O(1)$
- d) $O(n)$

Solution: (d) $O(n)$

Worst case is when the desired element is at the end of the array or not present at all, in this case you have to traverse till the end of the array, hence the complexity is $O(n)$.

11. What is the time complexity of binary search?

- a) $O(n \log n)$
- b) $O(\log n)$
- c) $O(n)$
- d) $O(n^2)$

Solution: (b) $O(\log n)$

12. Given an array $arr = \{2, 8, 77, 82, 99\}$ and $key = 82$; How many iterations are done until the element is found using binary search?

- a) 1
- b) 2
- c) 3
- d) 4

Solution: (b) 2

Explanation: Iteration1: $mid = 77$; Iteration2: $mid = 82$;

13. 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

14. If the given input array is sorted or nearly sorted, which of the following algorithm gives the best performance?

- a) Insertion sort
- b) Selection sort
- c) Bubble sort

Week 9

d) Quick sort

Solution: (a) Insertion sort

Insertion sort takes linear time when input array is sorted or almost sorted (maximum 1 or 2 elements are misplaced). All other sorting algorithms mentioned above will take more than linear time in their typical implementation.

15. What is the advantage of selection sort over other sorting techniques?

- a) It requires no additional storage space
- b) It is scalable
- c) It works best for inputs which are already sorted
- d) It is faster than any other sorting technique

Solution: (a)

Since selection sort is an in-place sorting algorithm, it does not require additional storage.