Problem Solving through Programming in C
Week 06 Assignment Solution

1. What is the right way to initialise an array in C?
   a) int arr{}={1,2, 5,6,9}
   b) int arr[5]={1,2, 5,6,9}
   c) int arr{5}={1,2, 5,6,9}
   d) int arr()={1,2, 5,6,9}

Solution: (b)

2. An integer array of dimension 10 is declared in a C program. The memory location of the first byte of the array is 1000. What will be the location of the 8th element of the array? (Assume integer takes 4 bytes of memory and the element stored at 1000 is identified as 1st element)
   a) 1028
   b) 1032
   c) 1024
   d) 1036

Solution: (a) Integer takes four bytes of memory. As the memory assignment to the elements are consecutive and the index starts from 0, the 8th element will be located at 1000+(7×4)

3. What will be the output after execution of the program?

```
#include <stdio.h>
main()
{
   int i,a[4]={3,1,2,4},result;
   result=a[0];
   for(i=1;i<4;i++)
   {
       if(result<a[i])
       continue;
       result=a[i];
   }
 printf("%d",result);
}
```

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

Solution: (a) The program finds the minimum element of an array. Hence, the output is 1.

4. Which of the statements is correct?
   a) An array contains more than one element
   b) All elements of array have to be of same data type
   c) The size of array has to be declared upfront
   d) All of the above

Solution: (d) All of the above

5. To compare two arrays, we can use
   a) Comparison operator ‘==’ directly on arrays
   b) Use switch case
   c) Using for loop
   d) Using ternary operator on arrays

Solution: (c) We can use a ‘for’ loop and equality check operator on each element of the arrays to compare.

6. 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)

7. What will be the output?

```
#include <stdio.h>
int main()
{
   int arr[]={1,2,3,4,5,6};
   int i,j,k;
   j=++arr[2];
   k=arr[1]++;
   i=arr[j++];
   printf("i=%d, j=%d,k=%d",i,j,k);
return 0;
}
```

   a) i=5, j=5, k=2
   b) i=6, j=5, k=3
   c) i=6, j=4, k=2
   d) i=5, j=4, k=2

Solution: (a)

k=arr[1]++ due to post increment operation, assignment is done first. so it actually becomes k=arr[1]=2. j=++arr[2]=++3=4. i=arr[j++]=arr[4++]=arr[4]=5 (as its post increment hence assignment is done first). Due to the post increment in i=arr[j++], the value of j is also incremented and finally becomes 5. So, finally i=5, j=5, k=2.

8. Array elements are stored in memory in the following order

a) Contiguous
b) Random
c) Both contagious and random
d) None

Solution: (a) Contiguous

9. What will be the output?

```
#include<stdio.h>
int main()
{
int n = 2;
int sum = 5;
switch(n)
{
        case 2: sum = sum-2;
        case 3: sum*=5;
        break;
        default:
                sum =0;

}
printf("%d",sum);
   return 0;
}
```

Solution: 15 (Short answer type)
N=2 therefore switch(2) i.e. case 2 will be executed. Inside case 2 sum becomes sum-2 = 5-2 = 3. As there is no break statement after case 2 therefore case 3 is also executed. Inside case 3, sum becomes sum*5= 3*5=15. After that the execution finds a break statement and comes out of the switch. So, finally 15 is printed.

10. How many ‘a’ will be printed when the following code is executed?

```
#include <stdio.h>
int main()
{
   int i = 0;
   char c = 'a';
   while (i < 5)
   {
      i++;
      switch (c)
      {
         case 'a':
         printf("%c ", c);
         break;
      }
   }
   printf("a\n");
return 0;
}
```

Solution: 6 (short answer type)
Initially, i=0, which satisfies the while condition. Case ‘a’ is always executed inside the while loop for i=1 to i=5 i.e., 5 times. Finally, another ‘a’ will be printed that is outside of the while loop. Therefore, a total of 6 times ‘a’ is printed.