Problem Solving through Programming in C
Week 12 Assignment Solution

1. Which of the following are themselves a collection of different data types?

   a) string
   b) structure
   c) char
   d) All of the above mentioned

Solution (b) Structure is a collection of different datatypes.

2. Which of the following comments about the usage structures is true?

   a) Storage class can be assigned to individual member
   b) Individual members can be initialized within a structure type declaration
   c) The scope of the member name is confined to the particular structure, within which it is defined
   d) None

Solution: (c) The scope of the member name is confined to the particular structure, within which it is defined

3. What is actually passed if you pass a structure variable to a function?

   a) Copy of structure variable
   b) Reference of structure variable
   c) Starting address of structure variable
   d) Ending address of structure variable

Answer: (a)

If you pass a structure variable by value without & operator, only a copy of the variable is passed. So changes made within that function do not reflect in the original variable.

4. The program will allocate …….bytes to ptr. Assume sizeof(int)=4.

```
#include<stdio.h>
#include<stdlib.h>

  int main()
  {
     int *ptr;
     ptr = (int*)malloc(sizeof(int)*4);
     ptr = realloc(ptr,sizeof(int)*2);
     return 0;
  }
```

Solution: 8

First, using malloc 16 bytes were allocated. Thereafter, applying calloc the size is adjusted and made 8.

5. Find the output of the following program

```
#include<stdio.h>
int main()
{
   char A[] = {'a','b','c','d','e','f','g','h'};
   char *p = A;
   ++p;
   while(*p != 'e')
   printf("%c", *p++);
   return 0;
}
```

a) abcd
b) bcd
c) cd
d) abcdfgh

Solution: (b)
First, the pointer points to the base of A. Then it's incremented by one to point to the element b. While p is not pointing e, the loop keeps prints the elements one after another. Therefore, the final answer is bcd.

6. Match the following

A. Newton Method	1. Integration
B. Lagrange Polynomial	2. Root finding
C. Trapezoidal Method	3. Differential Equation
D. Runge Kutta Method	4. Interpolation

a) A-2, B-4, C-1, D-3
b) A-3, B-1, C-2, D-4
c) A-1, B-4, C-3, D-2
d) A-2, B-3, C-4, D-1

Solution: (a) Appropriate methods for the problems need to be matched.

7. What is the output of the following C code? Assume that the address of x is 2000 (in decimal) and an integer requires four bytes of memory.

```
int main()
{
   unsigned int x[4][3] = {{1, 2, 3}, {4, 5, 6}, {7, 8, 9}, {10, 11, 12}};
   printf("%u,%u, %u", x+3, *(x+3),*(x+2)+3);
}
```

(a) 2036, 2036, 2036
(b) 2012, 4, 2204

(c) 2036, 10, 10
(d) 2012, 4, 6

Solution: (a)
All these points to the same element. Therefore, the same value will be printed. This is an example of pointer arithmetic in an 2D array.

8. What does fp point to in the program?

```
#include<stdio.h>
int main()
{
   FILE *fp;
   fp=fopen("hello", "r");
   return 0;
}
```

a) The first character in the file
b) A structure which contains a char pointer which points to the first character of a file.
c) The name of the file.
d) The last character in the file

Solution: (b) The fp is a structure which contains a char pointer which points to the first character of a file.

9. Choose a correct statement about C structure?

```
int main()
{
struct hello{};
return 0;
}
```

a) It is wrong to define an empty structure
b) Member variables can be added to a structure even after its first definition
c) There is no use of defining an empty structure
d) None of the above

Solution: (b)Member variables can be added to a structure even after its first definition

10. What is the output of the following C program?

```
#include <stdio.h>
struct p
{
   int x;
   char y;
};

int main()
{
   struct p p1[] = {1, 90, 62, 33, 3, 34};
   struct p *ptr1 = p1;
   int x = (sizeof(p1) / 3);
   if (x == sizeof(int) + sizeof(char))
      printf("True");
   else
      printf("False");
   return 0;
}
```

a) True
b) False
c) No output
d) Compilation error

Solution: (b) Size of the structure is the maximum size of the variable inside structure. Thus, the size of each element of structure p is 4 bytes (in gcc compiler, it can vary based on compiler). Thus, sizeof(p1) is 6*4=24. x will be 24/3=8. In the next step, sizeof(int)+sizeof(char) is 5 which is not equal to x. Hence, false will be printed.