STRUCTURES, UNIONS AND ENUMS

1. What is the similarity between a structure, union and an enumeration?

2. Will the following declaration work ?

```
typedef struct s
{
int a ;
float b ;
} s ;
```

3. Can a structure contain a pointer to itself?

4. Point out the error, if any, in the following code.

```
typedef struct
{
int data ;
NODEPTR link ;
} *NODEPTR ;
```

5. How will you eliminate the problem in 4 above ?

6. Point out the error, if any, in the following code.

```
#include<stdio.h>
#include<string.h>
void modify ( struct emp * ) ;
struct emp
{
char name[20] ;
int age ;
};
int main ( )
{
struct emp e = {"sanjay", 35 } ;
modify ( &e ) ;
printf ( "%s %d\n" , e.name , e.age ) ;
return 0 ;
}
void modify ( struct emp *p )
{
strupr ( p -> name );
p -> age = p-> age + 2 ;
}
```

7. Point out the error, if any, in the following code.

```
struct emp
{
int ecode;
struct emp e ;
} ;
```

8. Point out the error, if any, in the following code.

```
struct emp
{
int ecode ;
struct emp *e ;
} ;
```

9. Point out the eroor, if any, in the fllowing code.

```
typedef struct data mystruct ;
struct data
{
int x ;
mystruct *b ;
} ;
```

10. Will the following code work?

```
include<stdio.h>
#include<malloc.h>
#include<string.h>
struct emp
{
int len ;
char name[1] ;
} ;
int main (
{
char newname[ ]= "Rahul" ;
struct emp *p = ( struct emp * ) malloc ( sizeof  (struct emp ) -1 + strlen ( newname ) + 1 ) ;
p -> len = strlen ( newname );
strcpy ( p -> name, newname ) ;
printf (" %d %s\n" , p -> len, p -> name );
return 0 ;
}
```

11. Can you suggest a better way to write the program in 10 above?

12. How will you free the memory allocated in 11 above?

13. Can you rewrite the program in 11 such that while freeing the memory only one call to free( ) will suffice?

14. Which of the following is the correct output for the program given below?

```
#include<stdio.h>
int main ( )
{
struct value
{
int bit1 : 1 ;
int bit3 : 4 ;
int bit4 : 4 ;
} bit ;
printf ( "%d\n", sizeof ( bit ) ) ;
return 0 ;
}
```

A. 1 B.2 C.4 D.9

15. Which of the following is the correct output for the program given below?

```
 #include<stdio.h>
int main( )
{
    struct value
    {
        int bit1 : 1 ;
        int bit3 : 4 ;
        int bit4 : 4 ;
```

```
} bit = { 1,2,2 } ;
printf (“%d%d%5d\n”, bit.bit1, bit.bit3 , bit.bit4 );
return 0 ;
}
```
A. 1 2 3 B.0 2 2 C.-1 2 2 D.1 -2 -2

16. Which of the following is the correct output for the program given below?
```
#include<stdio.h>
int main( )
{
   enum vaule { VAL1 = 0, VAL2, VAL3, VAL4, VAL5 } var ;
   printf (“%d\n” , sizeof ( var ) ) ;
   return 0 ;
}
```
A.1 B.2 C.4 D.10

17. What will be the output of the following program?
```
#include<stdio.h>
#include<<string.h>
int main ( )
{
     struct emp
     {
         char *n ;
         int age ;
            } ;
struct emp e1 = {“Dravid”, 23 } ;
struct emp e2 = e1 ;
strupr ( e2.n ) ;
printf (“%s\n”, e1.n ) ;
return 0 ;
}
```

18. Point out the error, if any, in the following code.
```
 #include<stdio.h>
 int main ( )
{
   struct emp
    {
      char n[ 20 ];
      int age ;
    };
struct emp e1 = {“Dravid”,23} ;
struct emp e2 = e1 ;
if ( e1 == e2 )
     printf(“The structures are equal\n” ) ;
return 0 ;
}
```

19. How will you check whether the contents of two structure variables are same are not?

20. How are structure passing and returning implemented by the compiler?

21. How can I read/write structures from/to data files?

22. Which of the following is the correct output for the program given below?
```
 #include<stdio.h>
 int main( )
{
       enum days { MON = -1, TUE, WED = 6,THU, FRI, SAT } ;
```

```
    prinf ( “%d %d %d %d %d %d\n” ,MON, TUE, WED, THU, FRI, SAT ) ;
    return 0 ;
}
```

A. -1 0 1 2 3 4 B. -1 2 6 3 4 5 C. -1 0 6 2 3 4 D. -1 0 6 7 8 9

23. Which of the following is the correct output for the program given below?

```
#include<stdio.h>
int main( )
{
        enum days { MON = -1, TUE, WED = 6, THU, FRI, SAT };
        printf(“%d %d %d %d %d %d”, ++MON, TUE, WED, THU, FRI, SAT );
        printf ( “\n” ) ;
        return 0 ;
}
```

A. -1 0 1 2 3 4 B. errors C.0 1 6 3 4 5 D. 0 0 6 7 8 9

24. Which of the following is the correct output for the program given below?

```
#include<stdio.h>
int main( )
{
    union var
      {
          int a, b ;
      } ;
union var v ;
v.a = 10 ;
v.b = 20 ;
printf (“%d\n”, v.a );
return 0;
}
```

A. 10 B.20 C.30 D.0

25. If the following structure is written to a file using fwrite( ), can fread( ) read it back successfully?

```
Struct emp
{
    char *n ;
    int age  ;
};
struct emp e = { “sujay”, 15 };
FILE *fp ;
fp = fopen ( “names.dat”, “wb”) ;
fwrite ( &e, sizeof (e), 1, fp )  ;
```

26. If a char is one byte wide, an integer is 2 bytes wide and a long integer is 4 byte wide, then will the following structure always occupy 7 bytes?

```
Struct ex
{
 char ch ;
 int i ;
long int a ;
} ;
```

27. Point out the error, if any, in the follwing code.

```
#include <stdio.h>
int main ( )
{
 struct bits
 {
   float f : 2 ;
```

```
 } bit ;
printf ( “%d\n” ,sizeof ( bit ) ) ;
return 0 ;
}
```

28. Point out the error, if any, in the following code.

```
  #include<stdio.h>
   int main ( )
    {
    struct bits
    {
        int i : 40 ;
    } bit ;
    printf(“%d\n”, sizeof ( bit ) );
return 0 ;
}
```

29. What are the error does the following program give and what is the solution for it?

```
 #include<stdio.h>
 int main ( )
  {
       struct emp
       {
            char name[20] ;
            float sal ;
   } ;
    struct emp e[10] ;
     int i ;
     for ( i=0 ; i<= 9 ; i++ )
         scanf(“%s %f”,e[ i].name,&e[i].sal );
return 0 ;
}
```

30. How can I determine the byte offset of a field within a structure?

31. The way mentioning the array name or function name without [ ] or ( ) yields their base addresses, What do you obtain on mentioning the structure name?

32. What is main( ) returning in the following program?

```
 #include<stdio.h>
 struct transaction
  {
       int sno ;
      char desc[30] ;
     char dc ;
     float amount ;
}
/* here is the main program. */
int main ( int argc, char *argv[ ] )
{
     struct transaction t ;
   scanf(“%d %s %c %f\n”, &t.sno, t.desc, &t.dc, &t.amount );
   printf (“%d %s %c %f\”,t.sno, t.desc, t.dc, t.amount ) ;
   return 0 ;
}
```

33. Point out the error, if any, in the following program?

```
  #include<stdio.h>
  int main ( )
  {
```

```
      struct a
      {
          category : 5 ;
          scheme : 4 ;
      } ;
 printf ( " size = %d\n" , sizeof ( struct a ) ) ;
 return 0;
}
```

34. What is the difference between a structure and a union?

35. Is it necessary that size of all elements in a union should be same?

36. Point out the error, if any, in the following code.

```
#include<stdio.h>
int main ( )
{
    union a
    {
        int i ;
        char ch[2] ;
    } ;
    union a z1 = { 512 } ;
    union a z2 = { 0,2 } ;
    return 0 ;
}
```

37. What is the difference between an enumeration and a set of preprocessor #defines ?

38. Is there easy way to print enumeration values symbolically?

39. What is the use of bit fields in a structure declaration?

40. Can we have an array of bit fields? [Yes/No]

41. Which of the following statements are correct about the program given below?

```
#include<stdio.h>
int main( )
{
      union a
      {
          int i ;
          char ch [ 2 ] ;
        } ;
        union a u1 = { 512 } ;
        union 1 u2 = { 0,2 } ;
        return 0 ;
}
```

A. u2 cannot be initialized as shown.
B. u1 can be initialized as shown.
C. To initialize char ch[ ] of u2 '.' operator should be used.
D. The code causes an error 'Declaration syntax error'.

42. Which of the following is the correct output for the program given below?

```
#include<stdio.h>
#include<malloc.h>
int main ( )
{
    struct node
    {
```

```
            int data ;
            struct node *link ;
          } ;
        struct node *p, *q ;
        p = ( struct node *) malloc ( sizeof ( struct node ) ) ;
        q = ( struct node *) malloc ( sizeof ( struct node ) ) ;
        printf( "%d %d\n", sizeof ( p ) , sizeof ( q ) ) ;
        return 0 ;
}
```

A. 2 2 B. 8 8 C.5 5 D.4 4

43. Point out the error, if any, in the following code.

```
#include<stdio.h>
int main ( )
{
   struct  emp
   {
      char name[ 25 ] ;
      int age ;
      float bs ;
    } ;
    struct emp e ;
    e.name = " rahul" ;
    e.age = 25 ;
    printf ( "%s %d\n", e.name, e.age ) ;
    return 0 ;
}
```

44. Which of the following statements are correct about the program given below?

```
#include<stdio.h>
int main( )
{
      struct emp
      {
            char name[20] ;
            int age ;
            float sal ;
   } ;
   struct emp e[2] ;
   int i = 0 ;
   for( i=0 ; i<2 ; i++ )
       scanf( "%s %d %f", e[i].name, &e[i].age, &e[i].sal ) ;
   for( i=0 ; i<2 ; i++)
        printf( "%s %d %f\n", e[i].name, e[i].age, e[i].sal ) ;
   return 0 ;
}
```

A. Error: ' scanf( ) functions cannot be used for structure elements'.
B. The code runs successfully.
C. Error: 'floating-point formats not linked. Abnormal program termination'.
D. Error: 'structure variable must be initialized'.

45. Which of the following statement is correct about the statement given below?

```
Maruthi.engine.bolts = 25 ;
```

A. structure bolts is nested within structure engine.
B. structure engine is nested within structure maruti.
C. structure maruti is nested within structure engine.
D. structure maruti is nested within structure bolts.

46. Which of the following is the correct output for the program given below?

```
#include<stdio.h>
int main ( )
{
      struct byte
      {
            int one: 1 ;
       } ;
        struct byte var = { 1 } ;
        printf( "%d\n" , var.one ) ;
       return 0 ;
}
```

A. 1 B. -1 C. 0 D. Error

47. Which of the following statement correctly assigns 45 to month using pointer variables pdt ?

```
#include<stdio.h>
struct date
{
    int day ;
    int month ;
    int year ;
} ;

int main( )
{
    struct date d ;
    struct date *pdt ;
    pdt = &d ;
    return 0 ;
}
```

A. pdf.month = 12; B. &pdf.month = 12;
C. d.month=12; D. pdf→month=12;

48. Which of the following statement is true?

A. User has to explicitly define the numeric values of enumerations.
B. User has a control over the size of enumeration variables.
C. Enumerations can have an effect local to the block, if desired.
D. Enumerations have a global effect throughout the file.

49. Which of the following is correct output for the program given below?

```
#include<stdio.h>
int main ( )
{
    enum status { pass, fail, atkt };
    enum status stud1, stud2, stud3 ;
    stud1 = pass ;
   stud2 = atkt ;

   stud3 = fail ;
    printf( "%d %d %d\n", stud1, stud2, stud3 ) ;
    return 0 ;
}
```

A. 0 1 2 B.1 2 3 C.0 2 1 D.1 3 2

50. Which of the following is the correct output for the program given below?

```
#include<stdio.h>
int main ( )
{
      int i= 4, j = 8 ;
      printf ( "%d %d %d\n", i | j & j | i , i | j &&j | i , i^j ) ;
```

```
    return 0 ;
}
```

A. 12 1 12 B. 112 1 12 C. 32 1 12 D. -64 1 12

51. Which of the following is the correct output for the program given below, if size of int is 4 bytes?

```
#include<stdio.h>
int main ( )
 {
    union a
    {
        int  i;
        char ch[2] ;
   };
   union a u ;
  u. ch[0] = 0 ;
  u. ch[1] = 2 ;
  u. ch[1] = 0 ;
  u. ch[1] = 0 ;
 printf( "%d\n" , u. i ) ;
 return 0 ;
}
```

A.512 B.2 C.0 D. None of the above

52. Point out the error, if any, in the following code.

```
#include<stdio.h>
struct tryit
{
    char *str ;
    int i ;
} t ;
int main ( )
{
        scanf("%s %d" , t.str,  &t.i );
        printf("%s\n" , t.str ) ;
        return 0 ;
}
```

53. Which of the following is the correct output for the program given below?

```
#include<stdio.h>
struct course
{
      int courseno ;
      char coursename [25] ;
  } ;
  int main( )
   {
      struct course c[ ]  = {
                              { 102 , " thermal " },
                              { 103 , " manufacturing " },
                               { 104 , " design " }
                                } ;
        printf( "%d" , c[1] .courseno ) ;
        printf( "%s\n" , ( * ( c+2 ) ).coursename );
        return 0 ;
}
```

A. 103 design B. 102 Thermal C.103 manufacturing D.104 design

54. State True or False:

A. A structure can contain similar or dissimilar elements.

B. The '.' operator can be used to access structure elements using a structure variable.

C. A structure can be nested inside another structure.
D. It is not possible to create an array of pointer to structures.
E. One of the elements of a structure can be a pointer to the same structure.
F. On declaring a structure zero bytes are reserved in memory.
G. A union cannot be nested in a structure.
H. The ‘ - >’ operator can be used to access structure elements using a pointer to a structure variable.
I. The alignment of structure members in memory varies from one compiler to another.
J. Memory gets allocated for the members of a structure only when a structure variable is defined.
K. Nested unions are not allowed.
L. The scope of a structure should be global if it is to used in all functions in the program.
M. The alignment of structure members in memory can be controlled using the #pragma pack preprocessor directive.
N. Bit fields cannot be used in union.
O. By default a structure variable will be f auto storage class.
P. Union elements can be of different sizes.
Q. Size of the union is size of the longest element in the union.
R. If we initialize one element of the union it also initializes other elements of the union.
S. Union variables can have local scope or global scope.
T. The elements of a union are always accessed using & operator.
U. A Pointer to a union cannot be created.