PD2 LAB QUESTIONS

1. Predict the output of following C programs.

```c
#include <stdio.h>
enum day {sunday = 1, tuesday, wednesday, thursday, friday, saturday};

int main()
{
    enum day d = thursday;
    printf("The day number stored in d is %d", d);
    return 0;
}
```

2. Predict the output of following C programs.

```c
#include <stdio.h>
enum State {WORKING = 0, FAILED, FREEZED};
enum State currState = 2;

enum State FindState() {
    return currState;
}

int main() {
   (FindState() == WORKING)? printf("WORKING"): printf("NOT WORKING");
   return 0;
}
```

3. Predict the output of following C programs.

```c
main()
{
    enum resut {pass, fail};
    enum result s1,s2;
    s1=pass;
    s2=fail;
    printf("%d",s1);
}
```

4. Predict the output of following C programs.

```c
#include <stdio.h>
enum example {a = 1, b, c};
enum example example1 = 2;
enum example answer()
{
    return example1;
}

int main()
{
   (answer() == a)? printf("yes"): printf("no");
   return 0;
}
```

5. Predict the output of following C programs.

```c
#include<stdio.h>
#define MAX 4
enum sanfoundry
{
    a,b=3,c
};
main()
{
    if(MAX!=c)
        printtf("hello");
    else
        printf("welcome");
}
```

6. Point out the error(if any) in the following code.

```c
#include<stdio.h>
enum sanfoundry
{
    a,b,c
};
enum sanfoundry g;
main()
{
    g++;
    printf("%d",g);
}
```

7. What will be the output of the following C code if input given is 2?

```c
#include<stdio.h>
enum day
{
    a,b,c=5,d,e
};
main()
{
    printf("Enter the value for a");
    scanf("%d",a);
    printf("%d",a);
}
```

8. What will be the output of the following C code if the code is executed on a 32 bit platform?

```c
#include <stdio.h>
enum sanfoundry
 {
    c = 0,
    d = 10,
    h = 20,
    s = 3
} a;

int main()
{
        a = c;
        printf("Size of enum variable = %d bytes", sizeof(a));
        return 0;
}
```

9. Predict the output of following C programs.

```c
#include<stdio.h>
enum sanfoundry
{
    a=1,b,c,d,e
};
int main()
{
    printf("%d",b*c+e-d);
}
```

10. Predict the output of following C programs.

```c
#include<stdio.h>
enum sanfoundry
{
    a,b,c=5
};
int main()
{
    enum sanfoundry s;
    b=10;
    printf("%d",b);
}
```

11. Predict the output of following C programs.

```c
#include<stdio.h>
enum sanfoundry
{
    a=1,b
};
enum sanfoundry1
{
    c,d
};
int main()
{
    enum sanfoundry1 s1=c;
    enum sanfoundry1 s=a;
    enum sanfoundry s2=d;
    printf("%d",s);
    printf("%d",s1);
    printf("%d",s2);
}
```

Soru:

Ders devam listesinde bulunan öğrenciler için kullanılacak ‘student’ isimli struct yapısını, ‘number’, ‘name’, ‘surname’, ve ‘midtermGrade’ alanlarından oluşacak şekilde tanımlayınız. Tanımladığınız bu yapıyı kullanarak 10 öğrenci için bir struct dizisi oluşturunuz ve klavyeden 3 öğrencinin bilgilerini alınız. Daha sonra bilgileri alınan öğrencilerin bilgilerini ekrana yazdırınız.

Define a struct named ‘student’ for course attendance list of students. This struct consists of ‘number’, ‘name’, ‘surname’, and ‘midtermGrade’ fields. Using this struct you defined, create a struct array for 10 students and get the information of 3 students from the keyboard. Then, print the information of the students whose information was entered.

11. Predict the output of following C programs.

```c
#include <stdio.h>
#include <string.h>

struct Books {
   char  title[50];
   char  author[50];
   char  subject[100];
   int   book_id;
};

int main( ) {

   struct Books Book1;        /* Declare Book1 of type Book */
   struct Books Book2;        /* Declare Book2 of type Book */

   /* book 1 specification */
   strcpy( Book1.title, "C Programming");
   strcpy( Book1.author, "Nuha Ali");
   strcpy( Book1.subject, "C Programming Tutorial");
   Book1.book_id = 6495407;

   /* book 2 specification */
   strcpy( Book2.title, "Telecom Billing");
   strcpy( Book2.author, "Zara Ali");
   strcpy( Book2.subject, "Telecom Billing Tutorial");
   Book2.book_id = 6495700;

   /* print Book1 info */
   printf( "Book 1 title : %s\n", Book1.title);
   printf( "Book 1 author : %s\n", Book1.author);
   printf( "Book 1 subject : %s\n", Book1.subject);
   printf( "Book 1 book_id : %d\n", Book1.book_id);

   /* print Book2 info */
   printf( "Book 2 title : %s\n", Book2.title);
   printf( "Book 2 author : %s\n", Book2.author);
   printf( "Book 2 subject : %s\n", Book2.subject);
   printf( "Book 2 book_id : %d\n", Book2.book_id);

   return 0;
}
```

12. Predict the output of following C programs.

```c
include <stdio.h>
#include <string.h>

struct Books {
   char  title[50];
   char  author[50];
   char  subject[100];
   int   book_id;
};

/* function declaration */
void printBook( struct Books book );

int main( ) {

   struct Books Book1;        /* Declare Book1 of type Book */
   struct Books Book2;        /* Declare Book2 of type Book */

   /* book 1 specification */
   strcpy( Book1.title, "C Programming");
   strcpy( Book1.author, "Nuha Ali");
   strcpy( Book1.subject, "C Programming Tutorial");
   Book1.book_id = 6495407;

   /* book 2 specification */
   strcpy( Book2.title, "Telecom Billing");
   strcpy( Book2.author, "Zara Ali");
   strcpy( Book2.subject, "Telecom Billing Tutorial");
   Book2.book_id = 6495700;

   /* print Book1 info */
   printBook( Book1 );

   /* Print Book2 info */
   printBook( Book2 );

   return 0;
}

void printBook( struct Books book ) {

   printf( "Book title : %s\n", book.title);
   printf( "Book author : %s\n", book.author);
   printf( "Book subject : %s\n", book.subject);
   printf( "Book book_id : %d\n", book.book_id);
}
```

13. Predict the output of following C programs.

```c
#include <stdio.h>
#include <stdlib.h>
struct person {
   int age;
   float weight;
   char name[30];
};

int main()
{
   struct person *ptr;
   int i, n;

   printf("Enter the number of persons: ");
   scanf("%d", &n);

   // allocating memory for n numbers of struct person
   ptr = (struct person*) malloc(n * sizeof(struct person));

   for(i = 0; i < n; ++i)
   {
       printf("Enter first name and age respectively: ");

       // To access members of 1st struct person,
       // ptr->name and ptr->age is used

       // To access members of 2nd struct person,
       // (ptr+1)->name and (ptr+1)->age is used
       scanf("%s %d", (ptr+i)->name, &(ptr+i)->age);
   }

   printf("Displaying Information:\n");
   for(i = 0; i < n; ++i)
       printf("Name: %s\tAge: %d\n", (ptr+i)->name, (ptr+i)->age);

   return 0;
}
```

14. Write C program to add two Complex Numbers by passing structure to a function. (In this example, you should take two complex numbers as structures and add them by creating a user-defined function.)

15. Write C program to store information of students using structure. (In this example, you should store the information of 5 students by using an array of structures.)

16. Write C program to store data in structures dynamically. (In this example, you should to store the information entered by the user using dynamic memory allocation.)

17. What will be the output of the C program?

```
#include<stdio.h>
#include<string.h>
struct player
{
        char pname[20];
}pl;
char* play(struct player *temp_pl)
{
        strcpy(temp_pl->pname, "kohli");
        return temp_pl->pname;
}
int main()
{
        strcpy(pl.pname, "dhoni");
        printf("%s %s", pl.pname, play(&pl));
        return 0;
}
```

kohli kohli

18. What will be the output of the C program?

```
#include <stdio.h>
struct student
{
    char name[50];
    int age;
};

// function prototype
void display(struct student s);

int main()
{
    struct student s1;

    printf("Enter name: ");
    scanf("%[^\n]%*c", s1.name);

    printf("Enter age: ");
    scanf("%d", &s1.age);

    display(s1);   // passing struct as an argument

    return 0;
}
void display(struct student s)
{
  printf("\nDisplaying information\n");
  printf("Name: %s", s.name);
  printf("\nAge: %d", s.age);
}
```

19. What will be the output of the C program?

```c
#include <stdio.h>
struct student
{
    char name[50];
    int age;
};

// function prototype
struct student getInformation();

int main()
{
    struct student s;

    s = getInformation();

    printf("\nDisplaying information\n");
    printf("Name: %s", s.name);
    printf("\nRoll: %d", s.age);

    return 0;
}
struct student getInformation()
{
  struct student s1;

  printf("Enter name: ");
  scanf ("%[^\n]%*c", s1.name);

  printf("Enter age: ");
  scanf("%d", &s1.age);

  return s1;
}
```

20. What will be the output of the C program?

```c
#include <stdio.h>
typedef struct Complex
{
    float real;
    float imag;
} complex;

void addNumbers(complex c1, complex c2, complex *result);

int main()
{
    complex c1, c2, result;

    printf("For first number,\n");
    printf("Enter real part: ");
    scanf("%f", &c1.real);
    printf("Enter imaginary part: ");
    scanf("%f", &c1.imag);

    printf("For second number, \n");
    printf("Enter real part: ");
    scanf("%f", &c2.real);
    printf("Enter imaginary part: ");
    scanf("%f", &c2.imag);

    addNumbers(c1, c2, &result);
    printf("\nresult.real = %.1f\n", result.real);
    printf("result.imag = %.1f", result.imag);

    return 0;
}
void addNumbers(complex c1, complex c2, complex *result)
{
     result->real = c1.real + c2.real;
     result->imag = c1.imag + c2.imag;
}
```