

PD2 LAB3 SORULARI

Kod 1: Aşağıdaki kod yazılarak sonuçlar incelenmelidir (10 dk).

The results should be examined by writing the following code.

```
1 #include <stdio.h>
2
3 int main(void)
4 {
5     int i;
6     int *iptr;
7     i = 5;
8     iptr = &i;
9
10    printf("i adresi    %p\n", &i);
11    printf("iptr degeri %p\n", iptr);
12
13    printf("i degeri    %d\n", i);
14    printf("*iptr degeri %d\n", *iptr);
15
16    getchar();
17    return 0;
18 }
```

Kod 2: Aşağıdaki koda double veri tipi char, int ve float olarak değiştirilerek sonuçlar tekrar alınmalı ve incelenmelidir (15dk).

In the following code, the double data types are changed to char, int and float, and the results must be retrieved and examined again.

```
1 #include<stdio.h>
2 int main()
3 {
4     double i;
5     double *iptr;
6
7     iptr = &i;
8     printf("i boyutu: %d\n", sizeof(i));
9     printf("iptr boyutu: %d", sizeof(iptr));
10
11    getchar();
12    return 0;
13 }
```

Kod 3: Aşağıdaki kodda month dizisine rand fonksiyonu kullanılarak rastgele değerler atanmalı. Daha sonra sırasıyla her bir “elm=...” atamasından sonra printf fonksiyonu çağırılarak sonuçlar ekrana yazdırılmalıdır (20dk).

In the following code, random values must be assigned to the month array using the rand function. After each "elm = ..." assignment, the printf function must be called and the results printed on the screen

```
1  #include<stdio.h>
2  int main( void )
3  {
4      int elm;
5      int month[ 12 ];
6      int *ptr;
7      ptr = month; // month[0] 'ın adresini ptr'ye ata
8      elm = ptr[ 3 ]; // elm = month[ 3 ]
9      ptr = month + 3; // ptr, month[ 3 ] adresini gösterecek
10     ptr = &month[ 3 ]; // ptr, month[ 3 ] adresini gösterecek
11     elm = ( ptr+2 )[ 2 ]; // elm = ptr[ 4 ] ( = month[ 7 ] ).
12     elm = *( month + 3 );
13     ptr = month; // month[0] 'ın adresini ptr'ye ata
14     elm = *( ptr + 2 ); // elm = month[ 2 ]
15     elm = *( month + 1 ); // elm = month[ 1 ]
16
17     return 0;
18 }
```

Kod 4: Aşağıdaki kod yazılarak ekran çıktıları incelenmelidir (20 dk).

The results should be examined by writing the following code.

```

1  #include <stdio.h>
2  int main()
3  {
4      int i[10], j;
5      int *iptr;
6      for (j=0; j<10; j++)
7          i[j]=j;
8
9      iptr = i;
10     for (j=0; j<10; j++) {
11         printf("%d ", *iptr);
12         iptr++;
13     }
14     /* iptr artık dizinin başını göstermez */
15     printf("\n%d \n",*(iptr-1));
16     iptr = i;
17     for (j=0; j<10; j++)
18         printf("%d ", *(iptr+j));
19     /* iptr hala dizinin başını gösterir */
20     printf("\n%d",*iptr);
21     getchar();
22     return 0;
23 }

```

Kod 5: Aşağıdaki koddaki eksiklik giderilerek ekran çıktısının istenildiği gibi olması sağlanmalıdır (15 dk).

The code below should be corrected so that the output of the screen will be as desired.

```

1  #include <stdio.h>
2  int main()
3  {
4      char *a="1234567890";
5      char x[10];
6      char *p1, *p2;
7      printf("%s\n", a);
8      p1 = a;
9      p2 = x;
10     while (*p1 != '\0') {
11         *p2 = *p1;
12         p1++;
13         p2++;
14     }
15     printf("%s\n", x);
16     getchar();
17     return 0;
18 }

```