

Ders 26

Pointerlar

Örnek size of bellek gösterimi

```
int main()
{
    printf("sizeof(char) = %zu\n", sizeof(char));
    printf("sizeof(char*) = %zu\n", sizeof(char*));

    printf("sizeof(short) = %zu\n", sizeof(short));
    printf("sizeof(short*) = %zu\n", sizeof(short*));

    printf("sizeof(int) = %zu\n", sizeof(int));
    printf("sizeof(int*) = %zu\n", sizeof(int*));

    printf("sizeof(double) = %zu\n", sizeof(double));
    printf("sizeof(double*) = %zu\n", sizeof(double*));
}
```

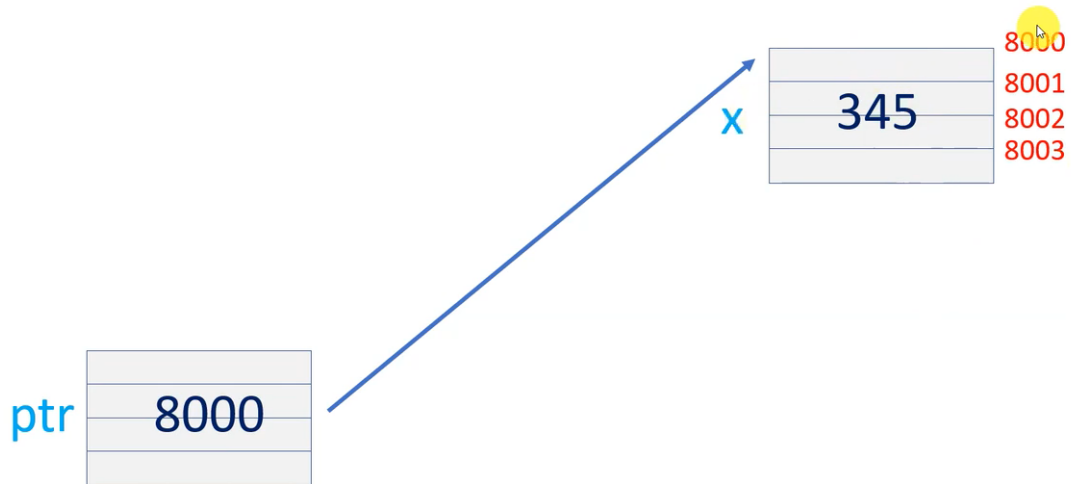
Microsoft Visual Studio Debug Console

```
sizeof(char) = 1
sizeof(char*) = 4
sizeof(short) = 2
sizeof(short*) = 4
sizeof(int) = 4
sizeof(int*) = 4
sizeof(double) = 8
sizeof(double*) = 4
```

D:\KURSLAR\NISAN_2022_C\Debug\NISAN_2022_C.exe (process 14640) exited with code 0.

Örnek

```
int main(void)
{
    int x = 345;
    int* ptr = &x;
```



Örnek ekrana adres yazdıralım.

```
int main(void)
{
    int x = 345;
    int* ptr = &x;

    printf("x = %d\n", x);
    printf("&x = %p\n", &x);
    printf("ptr = %p\n", ptr);
```

&Ptr nin adresini yazdıralım ptr den farklı olacak

```

int main(void)
{
    int x = 345;
    int* ptr = &x;

    printf("x = %d\n", x);
    printf("&x = %p\n", &x);
    printf("ptr = %p\n", ptr);
    printf("&ptr = %p\n", &ptr);
}

```

Microsoft Visual Studio Debug Console

```

x = 345
&x = 012FF7F4
ptr = 012FF7F4
&ptr = 012FF7E8

```

ÖRNEK DİZİNİN ADRESİ YAZDIRMA

```

int main(void)
{
    int a[10] = { 0 };
    int* p = a;

    printf("&a[0] = %p\n", &a[0]);
    printf("a      = %p\n", a);
    printf("p      = %p\n", p);
}

```

Microsoft Visual Studio Debug Console

```

&a[0] = 006FFD1C
a      = 006FFD1C
p      = 006FFD1C

```

D:\KURSLAR\NISAN_2022_C\Debug\NISAN_2022_C.exe (process 5936) exited with code 0.
Press any key to close this window . . .

İLK DEĞER VERMEK

```

int main(void)
{
    //int x = 10, * p1 = &x, * p2 = p1, a[] = { 1, 4, 5 }, * p3 = a;
    int x = 10;
    int* p1 = &x;
    int* p2 = p1;
    int a[] = { 1, 4, 5 };
    int* p3 = a;
}

```

ÖRNEK

```

int main(void)
{
    int x = 34;

    int* p1 = &x;
    int* p2 = &x;
    int* p3 = &x;
    printf("x = %d\n", x);

    ++* p1;
    ++* p2;
    ++* p3;

    printf("x = %d\n", x);
}

```

X =37 OLDU.

ÖRNEK

```

int main(void)
{
    int* p1 = &x, * p2 = &y;

    *p1 = *p2;
}

```

BURDA X=Y OLDU.

Örnek 4

```

void func(int *ptr)
{
    *ptr = 999;
}

int main()
{
    int x = 56;
    int* p = &x;
    int a[] = { 3, 7, 31 };

    func(&x);
    func(p);
    func(a);
}

```

Birinci atama xe

2. atamada x e

3. atama dizinin ilk elemanına func(a)

Swap fonksiyonu ben yazdım

```
void swap(int *px ,int * py){  
    int temp = *px;  
    *px= *py;  
    *py=temp;  
}  
  
int main(){  
    int x= 10;  
    int y = 30;  
    printf("x= %d  y= %d\n",x,y);  
    swap(&x,&y);  
    printf("x= %d  y= %d\n",x,y);  
}
```

Hazır swap fonksiyonuna pointer adreslerini verdik.

```
int main()  
{  
    int a = 56;  
    int b = 93;  
    int* pa = &a;  
    int* pb = &b;  
  
    swap(pa, pb);  
}
```

İlk elemanı değiştirme array decay

```
int main()  
{  
    int a[] = { 1, 4, 7, 2 };  
    int b[] = { 2, 5, 4, 2 };  
  
    swap(a, b);  
}
```

2 değişkenin büyüğü

```
int max2(int a, int b)
{
    if (a > b)
        return a;

    return b;
}

void max_2(int a, int b, int *pmax)
{
    if (a > b)
        *pmax = a;
    else
        *pmax = b;
}
```

```
void max2(int a, int b, int *pmax)
{
    if (a > b)
        *pmax = a;
    else
        *pmax = b;
}

int main()
{
    int x = 10, y = 30;
    int max;

    max2(x, y, &max);

    printf("max = %d\n", max);
}
```

Bu öğretmek için yazıldı normalde bu şekilde yazmayı tercih etmem. Daha zahmetli