

42 C06 Çözümleri

(+) Argüman Nedir ?

Komut satırından çalıştırılan bir C programına **argüman (parametre)** gönderebiliriz. Bu argümanlar program içinde **argc ve argv** değişkenleriyle alınır.

```
int main(int argc, char **argv)
argc (argument count)  Argüman sayısıdır.
argv (argument vector) Argümanları içeren bir dizidir.
```

Örnek ;

```
./program enes 42
```

```
argc  argv[0]      argv[1]  argv[2]
3      ./program   enes      42
```

Yani argv[0] daima programın adıdır. argc, girilen tüm argümanların sayısını içerir.

ex00;

```
int main(int ac, char **av)
{
    int i = 0;
    (void) ac;
    while (av[0][i])
    {
        write(1,&av[0][i],1);
        i++;
    }
    write(1,"\n",1);
    return (0);
}
```

Program Adını Alır: av[0], programın adını tutar. Örneğin, ./a.out gibi.

- **Her Karakteri Yazdırır:** while (av[0][i]) döngüsü, av[0] dizisinin her karakterini sırayla işler. write(1, &av[0][i], 1) fonksiyonu ile her bir karakter ekrana yazdırılır.
- **Bir Satır Baş Ekler:** Program adı yazdırıldıktan sonra, write(1, "\n", 1) ile bir satır başı karakteri eklenir.
- **Başarıyla Sonlanır:** Program başarılı bir şekilde tamamlanır ve return 0; ile sonlanır.

ex01

argc: Komut satırından girilen argümanların sayısını belirtir.
argv: Argümanları içeren bir dizidir. argv[0] programın adı, argv[1]'den itibaren ise kullanıcı tarafından girilen argümanlardır.

Bu program, argv[1]'den başlayarak her argümanı tek tek alır ve içindeki karakterleri sırayla ekrana yazdırır. Bunu yaparken:

- argv[i] ile argümanlara,
- argv[i][j] ile her argümanın karakterlerine erişilir.
- write fonksiyonu ile her bir karakter ekrana yazdırılır.
- Her argümandan sonra bir \n karakteri eklenir, böylece her argüman ayrı bir satırda görünür.

Kodun İşleyişi

Dış döngü (while(av[i]))

- i değişkeni 1'den başlar (çünkü argv[0] programın adıdır ve yazdırılmak istenmez).
- argv[i] boş olana kadar tüm argümanları işler.

İç döngü (while(av[i][j]))

- j değişkeni her argümanın karakterlerini tek tek yazdırır.
write(1, &av[i][j], 1);
- av[i][j]'nin adresini alarak karakter karakter ekrana yazdırır.

Yeni Satır (write(1, "\n", 1);)

- Her argüman yazdırıldıktan sonra bir yeni satır eklenir.

```
int main(int ac, char **av)
{
    int i = 1;
    while(av[i])
    {
        int j = 0;
        while(av[i][j])
        {
            write(1,&av[i][j],1);
            j++;
        }
        write(1,"\n",1);
        i++;
    }
    return (0);
}
```

ex02

```
#include <unistd.h>

void ft_putstr(*str)
{
    int i = 0;
    while(str[i] != '\0')
    {
        write(1,&str[i],1);
        i++;
    }
}

int main(int ac, char **av)
{
    int i = ac - 1;
    while(i > 0)
    {
        ft_putstr(av[i]);
        write(1, "\n", 1);
        i--;
    }
    return (0);
}
```

ft_putstr fonksiyonu:

Parametre olarak aldığı stringi (char *str) ekrana yazdırır. write(1, str++, 1); ile her karakteri tek tek yazar.

main fonksiyonu:

ac: Argüman sayısını tutar.

av: Argümanların olduğu dizidir.

i = ac - 1; ile **son argümandan** başlayarak geriye doğru gider.

while (i > 0) döngüsü:

argv[i] değerini ft_putstr ile ekrana yazdırır.

Yeni satır eklemek için write(1, "\n", 1); kullanılır.

i-- ile bir önceki argümana geçilir.