

ÇOK DEĞİŞKENLİ FONKSİYONLAR

Değişken Sayıda Argüman Alabilen Fonksiyonlar

Değişken sayıda argüman alabilen bir fonksiyona **çok değişkenli fonksiyon** (**variadic function**) denir. C dilindeki güçlü ancak çok nadiren kullanılan özelliklerden biridir. **printf()** ve **scanf()** fonksiyonları aslında değişken sayıda argümanı **biçim dizisinden** (**format string**) sonra koyabildiğimiz fonksiyonların en bilinen örnekleridir.

```
printf(const char* format,...);
```

```
printf("%d %c %s %c",i,c,str,c2); //5 argümanı olan printf
printf("%s %d %c",str,i,c); //4 argümanı olan printf
printf("%s",str); //2 argümanı olan printf
```

C Dilinde, değişken sayıda bağımsız değişkeni olan bir fonksiyon; en az bir sabit bağımsız değişkene sahip olacak şekilde tanımlanır ve ardından derleyicinin değişken sayıda bağımsız değişkeni ayrıştırmasını sağlayan bir **üç nokta simgesi** (**elipsis**) (...) eklenir.

```
dönüş-tipi fonksiyonkimliği(veri-tipi birinciargüman, ...);
```

Değişken argümanları işlemek için kodunuza **stdarg.h** başlık dosyasını kodunuza dahil etmeniz gerekir;

Fonksiyon	Açıklama
va_start(va_list ap, arg)	Bu fonksiyon, üç nokta ile verilen argümanları va_list değişkenine aktarır.
va_arg(va_list ap, type)	Her seferinde, üç nokta ile temsil edilen değişken listesindeki bir sonraki argümanı va_list üzerinden işler ve listenin sonuna ulaşana kadar onu type ile verilen veri tipine dönüştürür.
va_copy(va_list dest, va_list src)	va_list 'teki argümanların bir kopyasını oluşturur.
va_end(va_list ap)	Bu, va_list değişkenlerine erişimi sonlandırır.

Tablo 27. Stdarg.h Fonksiyonları

Aşağıda ilk parametre ile belirlenmiş argüman sayısı kadar argüman alan bir fonksiyon tanımlanmıştır.

```
#include <stdio.h>
#include <stdarg.h>

int argümanlarınHepsiniTopla(int kacAdet, ...) {
    va_list argumanlar;
    int sayac, toplam = 0;
    va_start(argumanlar, kacAdet);
    for (sayac = 0; sayac < kacAdet; sayac++)
        toplam += va_arg(argumanlar, int);
    va_end(argumanlar);
    return toplam;
}

int main(){
    printf("3 Argüman Toplamı = %d \n",
        argümanlarınHepsiniTopla(3, 10, 20, 30));
    printf("5 Argüman Toplamı = %d \n",
        argümanlarınHepsiniTopla(5, 10, 20, 30,40,50));
    return 0;
}
```

Ana Fonksiyonun Parametreleri

Ana fonksiyon (**main function**) programın icra edilmeye başladığı fonksiyondur. C Dilinde yazdığımız programlar da çalıştırılırken konsoldan argüman alabilir. Şu ana kadar argümansız ana fonksiyonu gördük. Ana fonksiyon;

- **Satır içi** (**inline**) fonksiyon olarak bildirilemez!
- Adresi alınamaz!
- Programın başka hiçbir yerinden çağrılmaz (**call**)!

Yazdığımız programlar da çalıştırılırken konsoldan argüman alabilir. Şu ana kadar argümansız ana fonksiyonu gördük. Argüman alan **ana fonksiyon** (**main function**) aşağıdaki iki şekilde gibi tanımlanır;

```
int main(int argc,
        char* argv[]) {
    // Ana fonksiyon Gövdesi
}
```

Ya da aşağıdaki şekilde tanımlanır;

```
int main(int argc,
        char* argv[],
        char ** envp) {
    // Ana fonksiyon Gövdesi
}
```

Aşağıda konsoldan çalıştırılırken alınan argümanları ve işletim sisteminin **ortam değişkenleri** (**environment variable**) konsola yazan bir program örneği verilmiştir;

```
#include <stdio.h>
int main( int argc, // argüman sayısı
char* argv[],      // Metin dizisi olarak konsoldan girilen komut ve argümanlar
char** envp)       // Metin dizisi olarak ortam (environment) değişkenleri
{
    int sayac;

    printf_s( "\nKonsoldan Girilen Argümanlar:\n" );
    for( sayac = 0; sayac < argc; sayac++ )
        printf_s( "  argv[%d]   %s\n", sayac, argv[sayac] );

    printf_s( "\nOrtam Değişkenleri:\n" );
    while( *envp != NULL )
        printf_s( "  %s\n", *(envp++) );

    return 0;
}
```

Programın derlenmesi ve çalıştırılması sonrası çıktı aşağıda verilmiştir;

```
C:\Users\ilhan>notepad main.c
C:\Users\ilhan>gcc main.c -o main.exe
C:\Users\ilhan>main 10 20 otuz elli

Konsoldan Girilen Argümanlar:
argv[0]   main
argv[1]   10
argv[2]   20
argv[3]   otuz
argv[4]   elli

Ortam Değişkenleri:
ALLUSERSPROFILE=C:\ProgramData
APPDATA=C:\Users\ilhan\AppData\Roaming
CommonProgramFiles=C:\Program Files\Common Files
```

```
CommonProgramFiles(x86)=C:\Program Files (x86)\Common Files
CommonProgramW6432=C:\Program Files\Common Files
COMPUTERNAME=DAMLANURO
ComSpec=C:\WINDOWS\system32\cmd.exe
configsetroot=C:\WINDOWS\ConfigSetRoot
DriverData=C:\Windows\System32\Drivers\DriverData
EFC_8812=1
FPS_BROWSER_APP_PROFILE_STRING=Internet Explorer
FPS_BROWSER_USER_PROFILE_STRING=Default
HOMEDRIVE=C:
HOMEPATH=\Users\ilhan
LOCALAPPDATA=C:\Users\ilhan\AppData\Local
LOGONSERVER=\\DAMLANURO
NUMBER_OF_PROCESSORS=8
OneDrive=C:\Users\ilhan\OneDrive
OneDriveConsumer=C:\Users\ilhan\OneDrive
OS=Windows_NT
Path=C:\WINDOWS\system32;C:\WINDOWS;Files\dotnet\;C:\Users\ilhan\Downloads\codeblocks\MinGW\bin;C:\Users\ilhan\AppData\Local\Microsoft\WindowsApps;
PATHEXT=.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
PROCESSOR_ARCHITECTURE=AMD64
PROCESSOR_IDENTIFIER=Intel64 Family 6 Model 142 Stepping 12, GenuineIntel
PROCESSOR_LEVEL=6
PROCESSOR_REVISION=8e0c
ProgramData=C:\ProgramData
ProgramFiles=C:\Program Files
ProgramFiles(x86)=C:\Program Files (x86)
ProgramW6432=C:\Program Files
PROMPT=$P$G
PSModulePath=C:\Program
Files\WindowsPowerShell\Modules;C:\WINDOWS\system32\WindowsPowerShell\v1.0\Modules
PUBLIC=C:\Users\Public
SESSIONNAME=Console
SystemDrive=C:
SystemRoot=C:\WINDOWS
TEMP=C:\Users\ilhan\AppData\Local\Temp
TMP=C:\Users\ilhan\AppData\Local\Temp
USERDOMAIN=DAMLANURO
USERDOMAIN_ROAMINGPROFILE=DAMLANURO
USERNAME=ilhan
USERPROFILE=C:\Users\ilhan
windir=C:\WINDOWS
ZES_ENABLE_SYSMAN=1

C:\Users\ilhan>
```