

C Programlama

Öğretim Elemanı Bilgileri

- Dr. Öğr. Üyesi Sema ATASEVER
- Nevşehir Hacı Bektaş Veli Üniversitesi, Mühendislik Mim.Fak. Bilgisayar Mühendisliği
- Web sayfası : <https://biz.nevsehir.edu.tr/sema/tr>
- Email : sema@nevsehir.edu.tr | s.atasever@gmail.com

Ölçme Yöntemi

ARA SINAV

- Ara sınav : 100 puan üzerinden değerlendirilecektir , Katkı : %40

FİNAL ÖDEVİ

- Final Sınavı : 100 puan üzerinden değerlendirilecektir , Katkı : %60
- Nihai ders notu hesabı : Ara sınavın %40'ı, Final notunun %60'ı alınarak hesaplanmaktadır!

14. Hafta Konuları

- Rastgele sayı üretimi (`rand()` ve `srand()` fonksiyonları kullanımı), dosya işlemleri (yazma, okuma, dosya kopyalama), iki boyutlu dizi değişkenler, standart giriş çıkış işlevleri, örnek kod uygulamaları.

Rand() Fonksiyonu Kullanımı

ARALIK

- `[0,RAND_MAX]`
- `RAND_MAX`: $\text{min} = 32767$ ve $\text{max} = 2^{16}$ olabilir.
- Bu aralıktaki her sayı eşit seçilme olasılığına sahiptir.

ÖLÇEKLEME

`rand() % a + b`

- `a` : ölçekleme faktörü
- `b` : kaydırma değeri

ÖRNEK:

```
rand() % 6          // [0,5] aralığında sayı üretir.  
  
rand() % 6 + 1      // [0+1,5+1] = [1,6] aralığında sayı üretir.
```

srand() Fonksiyonu Kullanımı

- srand() rastgele sayı dizisi üretmek için başlangıç değeri vermemizi sağlayan fonksiyondur.
- Eğer srand() fonksiyonunu kullanmazsak program rand()'ın seed değerini srand(1) olarak ayarlar ve bu da kodun her seferinde aynı sayı dizisini üretmesine neden olur.

<https://medium.com/@elifnurber/rastgele-sayi-%C3%BCreti%CC%87mi%CC%87-c-e1782ffebfaf>

```
#include <stdio.h>
int main (void)
{
    int i;
    int diceRoll;

    srand(time(NULL));

    for(i=0;i<2;i++){
        diceRoll = (rand()%6) + 1;
        printf("%d \n", diceRoll);
    }

    return 0;
}
```

Örnek :

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

int main(void)
{
    // This program will create same sequence of
    // random numbers on every program run

    for(int i = 0; i<5; i++)
        printf(" %d ", rand());

    return 0;
}
```

Örnek :

```
#include <stdio.h>
#include <stdlib.h>
int main() {
    int c, n;

    printf("Ten random numbers in [1,100]\n");

    for (c = 1; c <= 10; c++) {
        n = rand() % 100 + 1;
        printf("%d\n", n);
    }

    return 0;
}
```

Output

```
Ten random numbers in [1,100]
42
68
35
1
70
25
79
59
63
65
```


fopen işlevi

- Bir dosya üzerinde herhangi bir G / Ç işlemi yapmaya başlamadan önce, önce dosya açılmalıdır.
- Bir dosyayı açmak için dosyanın adını belirtmelisiniz.
- Sistem daha sonra bu dosyanın gerçekten var olup olmadığını kontrol eder ve belirli durumlarda, dosya açılmazsa sizin için dosyayı oluşturur.
- Bir dosya açıldığında, sisteme dosya ile gerçekleştirmeyi planladığınız G / Ç işlemlerinin türünü de belirtmeniz gerekir. Dosya verileri okumak için kullanılacaksa, normalde dosyayı okuma modunda açarsınız. Dosyaya veri yazmak istiyorsanız, dosyayı yazma modunda açarsınız.

Dosya işlemleri : Metin Dosyasına Yazmak

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

int main()
{
    int num;
    FILE *fptr;

    // use appropriate location if you are using MacOS or Linux
    fptr = fopen("C:/HAFTA14/program.txt", "w");

    if (fptr == NULL)
    {
        printf("Error!");
        exit(1);
    }

    printf("Enter num: ");
    scanf("%d", &num);

    fprintf(fptr, "%d", num);
    fclose(fptr);

    return 0;
}
```

Dosya işlemleri : Metin Dosyasından Okumak

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

int main()
{
    int num;
    FILE *fptr;

    if ((fptr = fopen("C://HAFTA14/program.txt", "r")) == NULL) {
        printf("Error! opening file");

        // Program exits if the file pointer returns NULL.
        exit(1);
    }

    fscanf(fptr, "%d", &num);

    printf("Value of n=%d", num);
    fclose(fptr);

    return 0;
}
```

Dosya işlemleri : Kopyalama

```
#include <stdio.h>
int main (void)
{
    char inName[64], outName[64];
    FILE *in, *out;
    int c;

    // get file names from user
    printf ("Enter name of file to be copied: ");
    scanf ("%63s", inName);
    printf ("Enter name of output file: ");
    scanf ("%63s", outName);

    // open input and output files
    if ( (in = fopen (inName, "r")) == NULL ) {
        printf ("Can't open %s for reading.\n", inName);
        return 1;
    }

    if ( (out = fopen (outName, "w")) == NULL ) {
        printf ("Can't open %s for writing.\n", outName);
        return 2;
    }

    // copy in to out
    while ( (c = getc (in)) != EOF )
        putc (c, out);

    // Close open files
    fclose (in);
    fclose (out);

    printf ("File has been copied.\n");

    return 0;
}
```

İki Boyutlu Dizi Değişkenler

```
#include<stdio.h>
int main(){
    /* 2D array declaration*/
    int disp[2][3];
    /*Counter variables for the loop*/
    int i, j;
    for(i=0; i<2; i++) {
        for(j=0; j<3; j++) {
            printf("Enter value for disp[%d][%d]:", i, j);
            scanf("%d", &disp[i][j]);
        }
    }
    //Displaying array elements
    printf("Two Dimensional array elements:\n");
    for(i=0; i<2; i++) {
        for(j=0; j<3; j++) {
            printf("%d ", disp[i][j]);
            if(j==2){
                printf("\n");
            }
        }
    }
    return 0;
}
```

Standart Giriş/Çıkış İşlevleri

Character I/O

- getchar, putchar

Formatted I/O

- scanf, printf

Karakter G/Ç Örnek

```
/* echo.c -- repeats input */  
#include <stdio.h>  
int main(void) {  
    char ch;  
    while ((ch = getchar()) != '\n')  
        putchar(ch);  
    return 0;  
}
```

EOF (End of File)

- Bir dosyayı okuyan bir programın, okumayı nerede durduracağını bilmesi için dosyanın sonunu algılamamanın bir yoluna ihtiyacı vardır. Bu nedenle, C giriş işlevleri yerleşik bir dosya sonu algılayıcıyla birlikte gelir. Klavye girişi bir dosya gibi işlendiğinden, klavye girişini sonlandırmak için bu dosya sonu algılayıcısını da kullanabilmelisiniz.
- CTRL-Z, klavye girişi için EOF'dir

Karakter G/Ç örnek + EOF

```
/* echo_eof.c -- repeats input */
#include <stdio.h>
int main(void) {
    char ch;
    while ((ch = getchar()) != EOF)
        putchar(ch);
    return 0;
}
```