

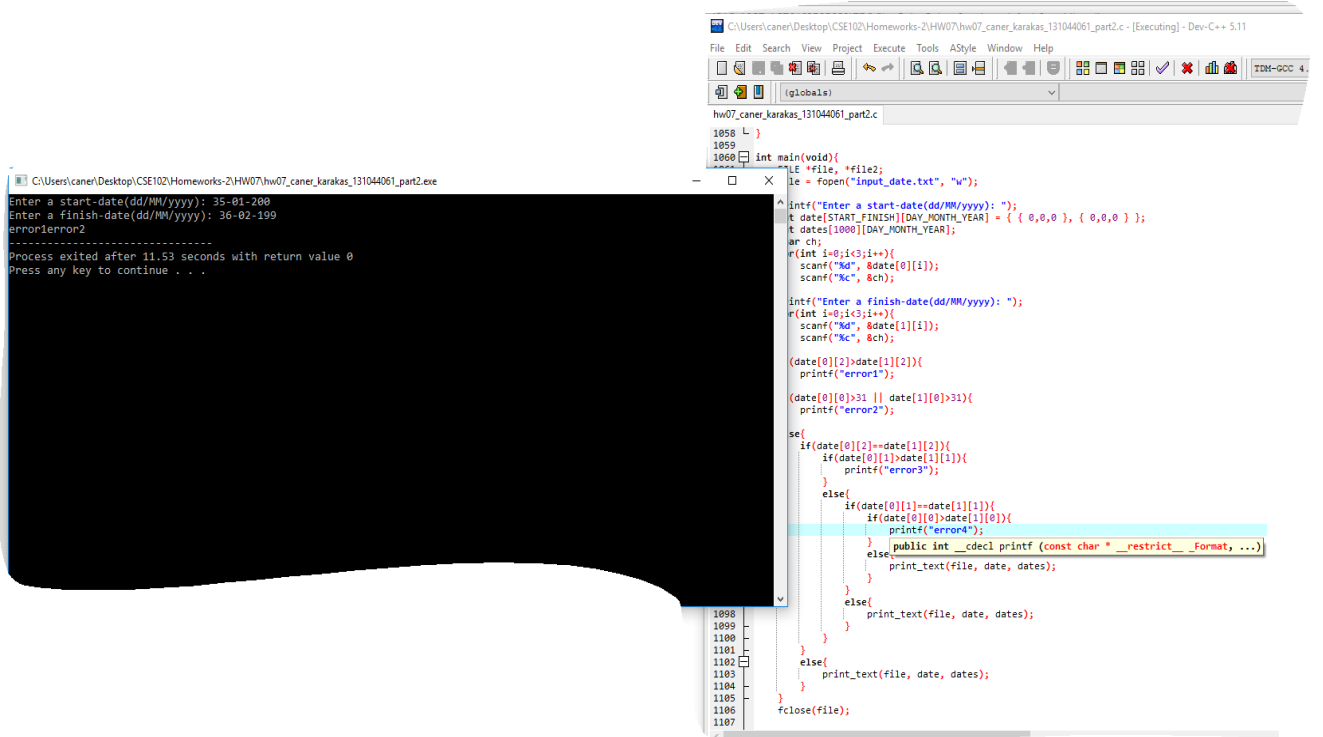
CSE102 - Computer Programming

HW 07

PART-2

Programımızdan ilk beklenen, girilen iki tarihin arasındaki bütün tarihleri; yeni bir dosya açıp içine yazmamız.

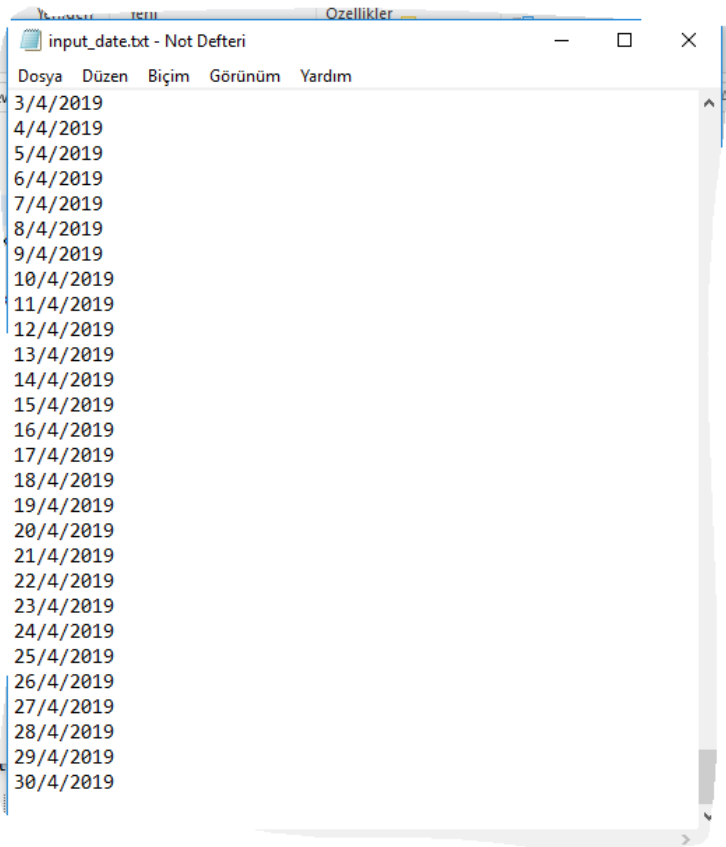
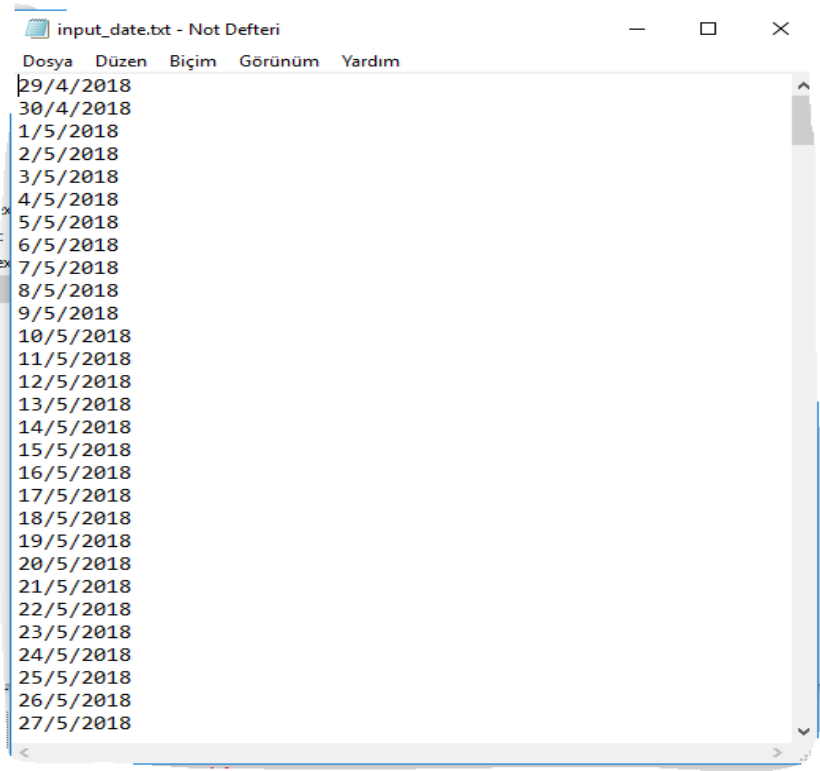
Bunu yaparken kullanıcıdan iki sayı girmesini istedim. Araya hangi karakter koyarsa koysun char değişkeninde tuttum. Bu da kalan tam sayıları rahatlıkla almamı kolaylaştırdı. Ancak yanlış girilebilecek tarihlerin kontrolünü her şeye dikkat etmeden yapmak zorunda kaldım. Kullanıcı 35 gibi bir gün girerse veya bitiş tarihi başlangıçtan daha az ise hata veriyor, diğer hataları kontrol etmiyor. 31 nisan gibi..



```
1058 }
1059
1060 int main(void){
    FILE *file, *file2;
    if (fopen("input_date.txt", "w");
    printf("Enter a start-date(dd/MM/yyyy): ");
    t date[START_FINISH][DAY_MONTH_YEAR] = { { 0,0,0 }, { 0,0,0 } };
    t dates[1000][DAY_MONTH_YEAR];
    char ch;
    for(int i=0; i<3; i++){
        scanf("%d", &date[i][1]);
        scanf("%c", &ch);
    }
    printf("Enter a finish-date(dd/MM/yyyy): ");
    for(int i=0; i<3; i++){
        scanf("%d", &date[i][1]);
        scanf("%c", &ch);
    }
    if(date[0][2]>date[1][2]){
        printf("error1");
    }
    if(date[0][0]>31 || date[1][0]>31){
        printf("error2");
    }
    if(date[0][2]==date[1][2]){
        if(date[0][1]>date[1][1]){
            printf("error3");
        }
        else{
            if(date[0][1]==date[1][1]){
                if(date[0][0]>date[1][0]){
                    printf("error4");
                }
                public int __cdecl printf (const char * __restrict __Format, ...)
            }
            else{
                print_text(file, date, dates);
            }
        }
    }
    else{
        print_text(file, date, dates);
    }
    print_text(file, date, dates);
}
fclose(file);
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
```

```
Enter a start-date(dd/MM/yyyy): 35-01-200
Enter a finish-date(dd/MM/yyyy): 36-02-199
error1error2
.....
Process exited after 11.53 seconds with return value 0
Press any key to continue . . .
```

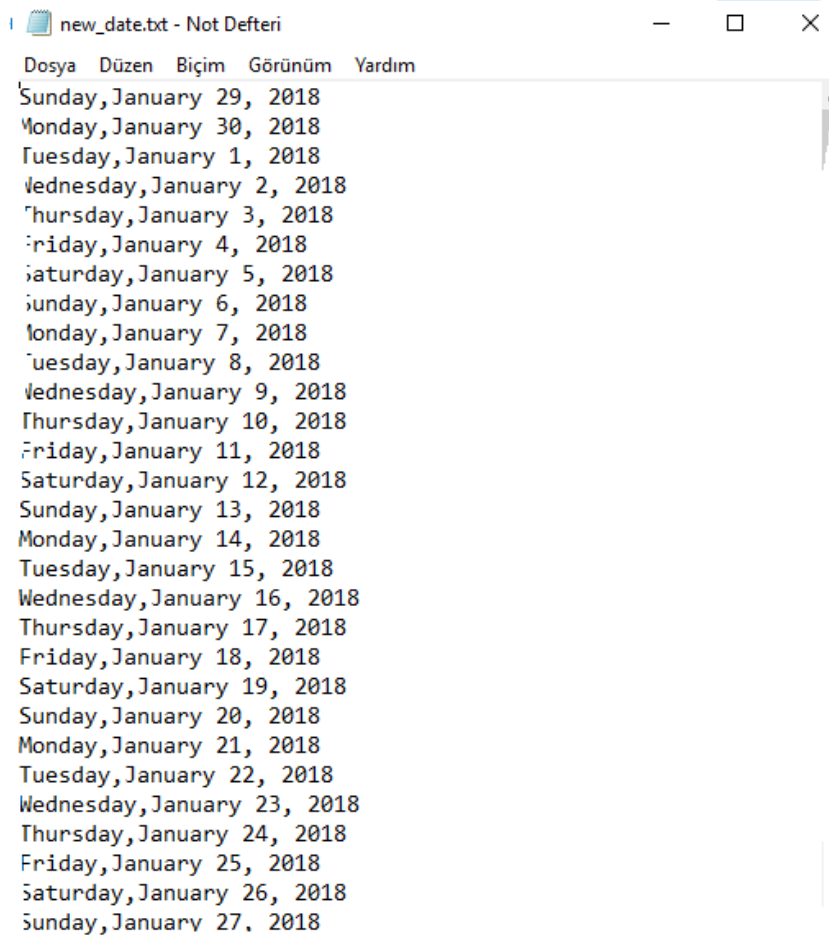
Bütün tarihleri hesaplarken ayların uzunluğunu ve artık yılları kontrol ederek yapıyor ve dosyaya yazıyor.



İkinci kısım için beklenen, ilk kısımda yazılan dosyadan okunup formatı değişip yeni bir dosyaya yazmak.

Bunu da dosyadan bir karakter bir sayı şeklinde tek satırı alarak döngü içinde hallettim. İlk önce bütün alınan satırları sayı dizisinde tuttum. Sonra işlemlere tabi tuttum.

Yılları güne çevirme işlemini de; 365 gün 6 saat mantığından 1 senenin içindeki saat ile istenilen yıla kadar toplam saati (2017 girilince 2016*saat şeklinde), istenilen ayın gününe kadar bulduğum günün toplam saati ile çarpıp, bulduğum sonuç ile toplam güne ulaştım(toplam saat / 24). Toplam günü mod 7 işlemine sokarak hangi güne denk geldiğini buldum ve bunu her tarih için tekrarladım. Artık yıla göre şubat ayından sonraki aylara bir gün ekledim. Sonra her bulduğum güne göre istenilen formatta yeni açtığımız dosyaya yazdım.



```
new_date.txt - Not Defteri
Dosya Düzen Biçim Görünüm Yardım
Sunday, January 29, 2018
Monday, January 30, 2018
Tuesday, January 1, 2018
Wednesday, January 2, 2018
Thursday, January 3, 2018
Friday, January 4, 2018
Saturday, January 5, 2018
Sunday, January 6, 2018
Monday, January 7, 2018
Tuesday, January 8, 2018
Wednesday, January 9, 2018
Thursday, January 10, 2018
Friday, January 11, 2018
Saturday, January 12, 2018
Sunday, January 13, 2018
Monday, January 14, 2018
Tuesday, January 15, 2018
Wednesday, January 16, 2018
Thursday, January 17, 2018
Friday, January 18, 2018
Saturday, January 19, 2018
Sunday, January 20, 2018
Monday, January 21, 2018
Tuesday, January 22, 2018
Wednesday, January 23, 2018
Thursday, January 24, 2018
Friday, January 25, 2018
Saturday, January 26, 2018
Sunday, January 27, 2018
```

