



FATİH SULTAN MEHMET VAKIF ÜNİVERSİTESİ

Öğrencinin;

ADI: Ahmet Emir

SOYADI: Kalafat

NO: 2021241007

BÖLÜM: Bilgisayar Mühendisliği

Projenin;

KONUSU: Çok Basamaklı Sayılar ile İşlemler

Dersin;

ADI: Bilgisayar Programlama III

EĞİTMEN: Musa AYDIN

1- Proje Konusu

C dilinde tanımlayamayacağımız büyüklükteki sayılarla toplama ve çıkarma işlemi yapabilmek.

Bu işlemi yapmak için heap alanında sayılar için array'ler tanımlayıp sayıların her basamağını array'in bir elemanı olarak kaydetmek.

Bu array'lerle işlem yapabilmek.

Makefile oluşturmayı ve kullanmayı öğrenmek.

2- Proje Çıktıları ve Başarı Ölçütleri

Toplama işlemi fonksiyonu:

```
void toplamaIslemi(uint8_t *sayi1, uint8_t *sayi2, int sayi1Boyut, int sayi2Boyut, uint8_t *cevap, int cevapBoyut)
{
    int k = cevapBoyut - 1; // toplama işlemi için birler basamağından başlıyoruz.
    int n = sayi1Boyut - 1;
    int m = sayi2Boyut - 1;
    while (k > 0)
    { // cevap arrayinin ilk indexine kadar dönüyoruz.
        *cevap = 0;

        if(m>=0){
            *(cevap + k) = *(sayi1 + n) + *(sayi2 + m);
            if(*(cevap + k) > 9 ){
                *(sayi1 + n - 1) += 1;
                *(cevap + k) -= 10;
            }
            else{
                *(cevap + k) = *(sayi1 + n);
            }
            if(*sayi1>9){
                *cevap = 1;
            }
            k--;
            n--;
            m--;
        }
    }
}
```

Çıkarma işlemi fonksiyonu:

```
void cikarmaIslemi(uint8_t *sayi1, uint8_t *sayi2, int sayi1Boyut, int sayi2Boyut, uint8_t *cevap, int cevapBoyut){
    int k = cevapBoyut - 1; // toplama işlemi için birler basamağından başlıyoruz.
    int n = sayi1Boyut - 1;
    int m = sayi2Boyut - 1;
    int gecici;
    while (k > 0)
    { // cevap arrayinin ilk indexine kadar dönüyoruz.
        *cevap = 0;

        if(m>=0){
            if(*(sayi1+n) < *(sayi2+m)){
                *(sayi1+n) += 10;
                gecici = *(sayi1 + n) - *(sayi2 + m);
                int geri = 1;
                while(*(sayi1+n-geri)==0){
                    *(sayi1+n-geri)=9;
                    geri++;
                }
                *(sayi1+n-geri) -= 1;
            }
            else{
                gecici = *(sayi1 + n) - *(sayi2 + m);
            }
            *(cevap + k) = gecici;
        }
        else{
            *(cevap + k) = *(sayi1 + n);
        }

        k--;
        n--;
        m--;
    }
}
```

Dosyalardan sayıları çekip array'e yazdığım fonksiyon:

```
void arrayDoldur(FILE *acilmisDosya, uint8_t *sayiArray)
{
    int i = 0;
    char c;
    while (!feof(acilmisDosya))
    {
        c = fgetc(acilmisDosya);
        if ((c == '\0'))
            continue; // array de space olusmaması için
        else if (c == '\n')
            continue; // array e enter girmemesi için
        else if (c >= '0' && c <= '9') // dosyadan gelen deger sayi ise arraye yazilir
        {
            int sayi = (c - '0');
            *(sayiArray + i) = sayi;
            i++;
        }
    }
    rewind(acilmisDosya);
}
```

3- Proje Süresince Yapılanlar

Proje süresince operasyon fonksiyonunda yapmamız gereken toplama çıkarma işlemleri gerçekten zorlandığım alanlardan birisiydi. Özellikle bu konu üzerinde düşünmek için önce rastgele sayılar oluşturmadan kendi sabit sayıların olduğu array'ler hazırladım. Böylece nerede hata yaptığımı keşfedip daha hızlı ilerleyebildim.

4- Ek Açıklamalar

Programımın gelişim ve sürüm takibini GitHub üzerinden yaptım. Linki şu şekildedir: https://github.com/emirkalafat/FSMVU_BPIII_FallTermProject2

Sunum video linki:

<https://youtu.be/WpzNPD40IB0>

5- Kaynakça

Davender S. Malik, C++ Programming Program Design Including Data Structures, 8th Edition

<https://www.javatpoint.com/c-programming-language-tutorial>