

UBI 534 - Temel Algoritmik Yöntemler-Ödev 3

```
#include <string> // for string ops. (c++)
#include <stdio.h> // for printf (standard c)
#include <stdlib.h> // standart c libs.

using namespace std;

void radix (int byte, long N, long *source, long *dest)
{
    long count[256];
    long index[256];
    int i;
    memset (count, 0, sizeof (count));
    for ( i=0; i<N; i++ ) count[((source[i])>>(byte*8))&0xff]++;

    index[0]=0;
    for ( i=1; i<256; i++ ) index[i]=index[i-1]+count[i-1];
    for ( i=0; i<N; i++ ) dest[index[((source[i])>>(byte*8))&0xff]++] = source[i];
}

void radixsort (long *source, long *temp, long N)
{
    radix (0, N, source, temp);
    radix (1, N, temp, source);
    radix (2, N, source, temp);
    radix (3, N, temp, source);
}

void make_random (long *data, long N)
{
    for ( int i=0; i<N; i++ ) data[i]=rand()|(rand()<<16);
}

long data[100];
long temp[100];

void main (void)
{
    make_random(data, 100);
    radixsort (data, temp, 100);
    for ( int i=0; i<100; i++ ) printf("%d \n",data[i]);
}
```

Yukarıda Radix-Sort algoritmasının C kodu verilmiştir. Bu ödevde istenenler aşağıda sıralanmıştır:

- a) Bu algoritmayı $O(n^2)$ karmaşıklığa sahip bir sıralama ile süre bakımından karşılaştırınız. Bu algoritmalarından birisini (*Insertion-Sort*) derste görmüştük. Sıralanan sayılar 32 bit tamsayılar olduğu için hızlı olacaktır. Süre ölçümü yaparken daha önceki ödevlerde olduğu gibi dizi boyutlarını değiştirip, ölçümlerinizi alınız.
- b) *Radix-Sort*'u daha önce yazdığımız *Randomized-Quick-Sort* ile süre açısından karşılaştırınız.

c) Yukarıda verilen *Radix-Sort* 32 biti 4'e bölmekte ve *Counting-Sort* benzeri bir sıralama ile sıralamayı yapmaktadır.

- 4'e bölmek en iyi (optimum) çözüm müdür? bulunuz. Örneğin 2, 4, 8 ve 16'ya bölmek mümkün, hepsini deneyip en iyisi hangisi bulunuz (Teorik ve Pratik olarak karşılaştırınız).

Rapor için not: Raporunuzda yaptığınız değişiklikleri, kullandığınız dizi boyutlarını, ekran çıktılarını da veriniz. Sonuçlardan grafikler çizerseniz raporun okunurluğu açısından iyi olabilir. Kullandığımız makinenin özelliklerini, hız, bellek, tampon (*cache*) bellek kapasitelerini yazınız.

Ödevin son tarihi: 22.Nisan.2013, 23.59

Ödev gönderimi: Lütfen ödevinizi dersin asistanı olan Öğr. Gör. Kaya Oğuz'un aşağıdaki e-postalarına gönderiniz:

kaya.oguz@ege.edu.tr

kayaoguz@gmail.com

Yrd. Doç. Dr. Orhan Dağdeviren

Uluslararası Bilgisayar Enstitüsü

Ege Üniversitesi