Homework 1

Problem 1

Suppose we have a hard disk with:

512 B sector size

206 Sectors per track in average

50864 tracks per surface

2 surfaces per platter

14 platters per disk

A. What is the capacity of disk in GB? Please use the unit conversion 1KB=1024B and similar conversions on MB and GB.

   A: 14*2*50864*206*512B = 139.9GB

B. The manufacturer claims it to be a 146.8GB hard disk. If you didn't make anything wrong in the first sub-problem, the actual capacity is smaller than the claimed one.

   a) Why could there be such a difference?

      A: manufactures use the unit of 1KB=1000B, and similar conversions on MB and GB.

   b) How much percentage of capacity is lost between the actual one and the claim one?

      A: (146.8GB – 139.9GB) / 146.8GB = 4.7%

Problem 2

Suppose that a 1MB file consisting of 512-byte logical blocks is stored on a disk drive with the following characteristics:

Rotational rate: 10,000 RPM

$T_{\text{avg seek}}$: 5 ms

Average # sectors/track: 1000

Surfaces: 4

Sector size: 512 B

For each case below, suppose that a program reads the logical blocks of the file sequentially, one after the other, and that the time to position the head over the first block is $T_{\text{avg seek}} + T_{\text{avg rotation}}$.

A. Best case: Estimate the optimal time (in ms) required to read the file given the best possible mapping of logical blocks to disk sectors (i.e., sequential).

   A: 5 + 0.5*60000/10000 + (1024*1024/512) * (1/1000) * 60000/10000 = 20.288 ms

B. Random case: Estimate the time (in ms) required to read the file if blocks are mapped randomly to disk sectors.

   A: (5 + 0.5*60000/10000 + (1/1000) * 60000/10000) * (1024*1024/512) = 16396.288 ms

Problem 3

Permute the loops in the following function so that it scans the three-dimensional array a with a stride-1 reference pattern.

```
#define N 8
int sumarray3d (int a[N][N][N])
{
        int i, j, k, sum = 0;

        for (i = 0; i < N; i++) {
                for (j = 0; j < N; j++) {
                        for (k = 0; k < N; k++) {
                                sum += a[k][i][j];
                        }
                }
        }
```

A:

```
#define N 8
int sumarray3d (int a[N][N][N])
{
    int i, j, k, sum = 0;

    for (i = 0; i < N; i++) {
        for (j = 0; j < N; j++) {
            for (k = 0; k < N; k++) {
                sum += a[i][j][k];
            }
        }
    }
    return sum;
}
```