CS3114 Fall 2012 Homework Assignment 3
Sample Solutions

1. Suppose that a record is 32 bytes, a block is 1024 bytes (thus, there are 32 records per block), and that working memory is 1MB (there is also additional space available for I/O buffers, program variables, etc.). What is the *expected* size for the largest file that can be merged using replacement selection followed by a *single* pass of multiway merge? Explain how you got your answer.

Since working memory is 1MB and the block size is 1KB, the number of blocks in working memory is 1024. The expected runlength is 2MB, since replacement selection will, on average, produce runs that are twice the memory size. 1024 runs can be merged in a single multiway merge operation. Thus, the largest expected file size for a single pass of multiway merge is 2 Gigabytes.

---

2. A typical disk drive from 2004 has the following specifications. The total storage is approximately 120GB on 6 platter surfaces or 20GB/platter. Each platter has 16K tracks with 2560 sectors/track (a sector holds 512 bytes) and 8 sectors/cluster. The disk turns at 7200 rpm. The track-to-track seek time is 2.0 ms, and the average seek time is 10.0 ms. Calculate the time required to read a 16MB file assuming

(a) The file is stored on a series of contiguous tracks, as few tracks as possible.

One track holds 1280K bytes, so the file requires 12.8 tracks, whose read time will be nearly identical to 13 full tracks. Seek time to the first track is 10 ms. Latency and read time together require $1.5 * 8.33$ ms. Thus, the time to read the first track is $10 + 1.5 * 8.33 \approx 22.5$ ms. The time to read each of the next 12 tracks is $2 + 1.5 * 8.33 \approx 14.5$ ms. Thus, the total time required is $22.5 + 12 * 14.5 = 196.5$ ms.

(b) The file is spread randomly across the disk in 4KB clusters.

16MB at 4KB per cluster is 4096 clusters, each requiring an independent seek and rotational delay. The time to do the actual read after rotational delay is only 8/2560 of a read, so we can safely ignore that. Thus, the total time required is $4096 * (10 + 0.5 * 8.33) \approx 58.03$ seconds.

---

3. B^{+}-tree Questions

(a) Show the result of inserting the values 1, 2, 3, 4, 5, and 6 (in that order) into the B^{+}-tree of Figure 10.18 in the textbook.

```
                                18/33
                                / | \
       --------------------------  |  -----------
       |                           |            |
      4/10                         23          same
     / | \                         /\
 -----  |  ------           -------  ---
 |      |      |            |          |
1/2/3  4/5/6  10/12/15  18/19/20/21/22  23/30/31
```

(b) Show the result of deleting first the value 31 and then the value 52 from the B^{+}-tree of Figure 10.18 in the textbook.

I will accept either of the following (though the first one is what should really happen).

```
                 22
                /  \
       ---------    ------------
       |                       |
      18                      33
     /  \                    /  \
   ---    -----          -----    ------
   |          |          |             |
10/12/15  18/19/20/21   22/23/30   33/45/47/48/50
```

```
             18|22|33
            /  |  |  \
           |   |   \  -----------------
   ---------   |    --------          \
   |           |           |           |
10/12/15   18/19/20/21   22/23/30   33/45/47/48/50
```

---

4. Show the shortest paths generated by running Dijkstra's shortest-paths algorithm on the graph of Figure 11.26, beginning at Vertex 5. Show the D values as each vertex is processed, as in Figure 11.19.

Vertex	1	2	3	4	5	6
Initial	∞	∞	∞	∞	0	∞
Process 5	∞	∞	15	11	0	3
Process 6	5	∞	15	11	0	3
Process 1	5	15	15	11	0	3
Process 4	5	15	15	11	0	3
Process 3	5	15	15	11	0	3
Process 2	5	15	15	11	0	3

---

5. Write an algorithm to determine whether a directed graph of $|V|$ vertices contains a cycle. Your algorithm should run in $\Theta(|V| + |E|)$ time.

To solve this problem, simply run the BFS topological sort algorithm. If there are any cycles, then some vertices will remain in the queue.
