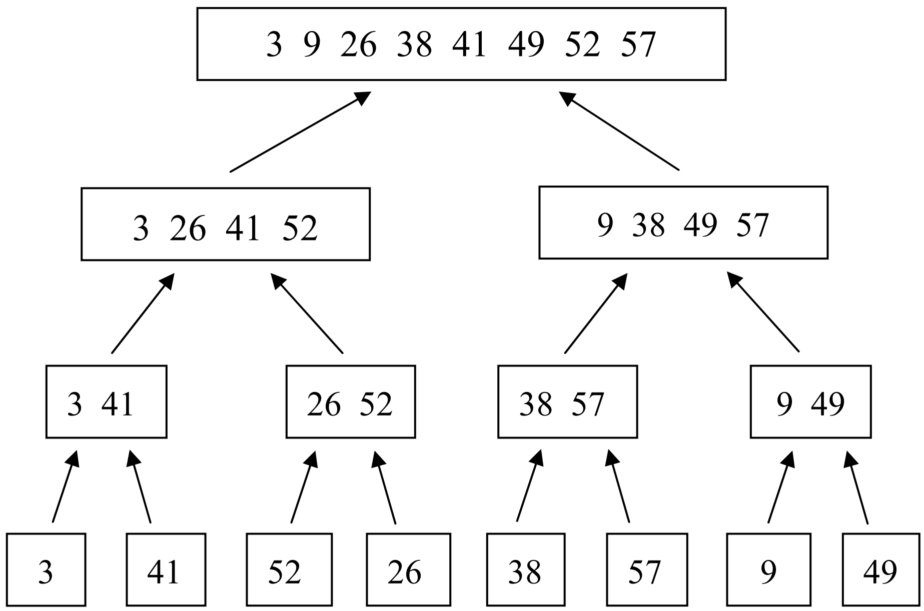
Solutions to CS / MCS 401 Week #3-4 Exercises (Spring 2008)

Exercise 2.3-1



Exercise D.

$C(0) = 0 = d \cdot 0$, so the result holds when $n = 0$.

Let $n \geq 1$, and assume the result holds for all i with $i < n$. $C(n) = d + C(k) + C(n-1-k)$ where $0 \leq k \leq n-1$. Note $n-k-1 \leq n-1$. By the inductive hypothesis,

$$C(n) = d + dk + d(n-k-1) = d(1 + k + (n-k-1)) = dn,$$

so the result also holds for n. By induction it holds for all nonnegative integers.

Exercise E.

Let L, M, and R be sorted arrays of length $n/3$ (possibly $\lfloor n/3 \rfloor$ or $\lceil n/3 \rceil$, so the sum of the lengths is n). For simplicity, assume n is a power of 3, so always L, M, and R have length $n/3$. Assume that each array has an extra element ∞ at the end. We can merge L, M, and R into a single sorted array A of length n using the algorithm below. Here i, j, and k represent the positions of the current elements in L, M, and R respectively; and x represents the smallest element not yet merged from M or R, provided $xValid$ is true. As usual, indentation indicates nesting of blocks.

```
i = 1;  j = 1;  k = 1;
xValid = false;
for ( q = 1,2, ..., n )
    if ( not xValid )
        if ( M[j] ≤ R[k] )          (*)
            x = M[j];
            j = j + 1;
        else
            x = R[k];
            k = k + 1;
        xValid = true;
    if ( L[i] ≤ x )                 (**)
        A[q] = L[i];
        i = i + 1;
    else
        A[q] = x;
        xValid = false;
```

Comparisons are performed in the lines (*) and (**). The comparison in line (**) is performed on each pass through the loop — a total of n times. The comparison on line (*) is always performed on the first pass ($q = 1$). On the remaining passes, it is performed if the element merged to A on the previous pass came from M or R, but *not* if it came from L. Thus the total number of comparisons in line (*) is

$$\begin{aligned} & n - (\text{number of elements merged from L on the first } n{-}1 \text{ passes}) \\ &\quad = n - (n/3 \text{ or } n/3 - 1) \\ &\quad = 2/3\,n \ \text{ or } \ 2/3\,n + 1 \end{aligned}$$

times. The total number of comparisons performed by the algorithm is $5/3\,n$ or $5/3\,n + 1$.

<u>Exercise F.</u>

$C(n) = 3C(n/3) + 5/3n$, $C(1) = 0$. We assume $n = 3^k$, so $k = \log_3(n)$.

$$\begin{aligned} C(n) &= 3C(n/3) + 5/3\,n \\ &= 3\big(3C(n/3^2) + (5/3)(n/3)\big) + 5/3\,n \\ &= 3^2C(n/3^2) + 2(5/3\,n) \\ &= 3^2\big(3C(n/3^3) + (5/3)(n/3^2)\big) + 2(5/3\,n) \\ &= 3^3C(n/3^3) + 3(5/3\,n) \\ &\ \ \vdots \\ &= 3^kC(n/3^k) + k(5/3\,n) \\ &= nC(1) + 5/3\,n\log_3(n) \\ &= 5/3\,n\log_3(n) \end{aligned}$$

The exact solution when n is a power of 3 is $\mathbf{C(n) = 5/3\,n\log_3(n) \approx 1.052\,n\lg(n)}$.

By contrast, ordinary (2-way) mergesort uses approximately $n\lg(n)$ comparisons.

Exercise G In each part, we assume $n = 2^k$, so $k = \lg(n)$.

a)
$$
\begin{aligned}
C(n) &= C(n/2) + 2n + 3 \\
&= \big(C(n/2^2) + 2(n/2) + 3\big) + 2n + 3 \\
&= C(n/2^2) + 2(n/2 + n) + 2\cdot 3 \\
&= \big(C(n/2^3) + 2(n/2^2) + 3\big) + 2(n/2 + n) + 2\cdot 3 \\
&= C(n/2^3) + 2(n/2^2 + n/2 + n) + 3\cdot 3 \\
&\ \vdots \\
&= C(n/2^k) + 2(n/2^{k-1} + \ldots + n/2^2 + n/2 + n) + k\cdot 3 \\
&= C(1) + 2n(1/2^{k-1} + \ldots + 1/2^2 + 1/2 + 1) + k\cdot 3 \\
&= 1 + 2n(2 - 1/2^{k-1}) + 3\lg(n) \\
&= 1 + 4n - 4 + 3\lg(n) \qquad \text{(since } n/2^{k-1} = 2^k/2^{k-1} = 2\text{)} \\
&= \mathbf{4n - 3 + 3\lg(n)}
\end{aligned}
$$

b)
$$
\begin{aligned}
C(n) &= 2C(n/2) + n\lg(n) \\
&= 2\big(2C(n/2^2) + n/2\cdot\lg(n/2)\big) + n\lg(n) \\
&= 2^2 C(n/2^2) + n(\lg(n) - 1) + n\lg(n) \\
&= 2^2 C(n/2^2) + 2n\lg(n) - n \\
&= 2^2\big(2C(n/2^3) + n/2^2\cdot\lg(n/2^2)\big) + 2n\lg(n) - n \\
&= 2^3 C(n/2^3) + n(\lg(n) - 2) + 2n\lg(n) - n \\
&= 2^3 C(n/2^3) + 3n\lg(n) - n(1+2) \\
&\ \vdots \\
&= 2^k C(n/2^k) + kn\lg(n) - n(1+2+\ldots+k-1) \\
&= nC(1) + n(\lg(n))^2 - nk(k-1)/2 \\
&= n(\lg(n))^2 - n\lg(n)(\lg(n)-1)/2 \\
&= \mathbf{n(\lg(n))^2/2 + n(\lg(n))/2}
\end{aligned}
$$

c)
$$
\begin{aligned}
C(n) &= 4C(n/2) + 3n^2 && (*) \\
&= 4\big(4C(n/2^2) + 3(n/2)^2\big) + 3n^2 \\
&= 4^2 C(n/2^2) + 4\cdot 3(n/2)^2 + n^2 \\
&= 4^2 C(n/2^2) + 3(2n^2) && (**) \\
&= 4^2\big(4C(n/2^3) + 3(n/2^2)^2\big) + 3(2n^2) \\
&= 4^3 C(n/2^3) + 4^2\cdot 3(n/2^2)^2 + 3(2n^2) \\
&= 4^3 C(n/2^3) + 3(3n^2) && (***) \\
&\ \vdots \\
&= 4^k C(n/2^k) + 3(kn^2) && \text{following pattern on lines } (*), (**), (***) \\
&= 4^k C(1) + 3\lg(n)\, n^2 \\
&= 4^k 0 + 3n^2\lg(n) = \mathbf{3n^2\lg(n)}
\end{aligned}
$$