Computing Science 204 (B1)

Analysis of Merge Sort

Wednesday, August 2

Recall that when we use merge-sort to sort a list of n numbers, where $n > 1$, we divide the list into (almost) equal parts of length

$$\left\lfloor \frac{n}{2} \right\rfloor \qquad \text{and} \qquad \left\lceil \frac{n}{2} \right\rceil .$$

After each part has been sorted (by applying the same method recursively), we merge the numbers into one sorted list by doing at most $n-1$ comparisons. The procedure is given below.

```
mergesort(first,last)
{
   if (first < last)
   {
       mergesort(first,[(first+last)/2]);
       mergesort([(first+last)/2]+1,last);
       merge(first,[(first+last)/2],[(first+last)/2]+1,last);
   }
}
```

It is easy to see that if we have two lists A and B with n and m entries, respectively, then we do at most $n+m-1$ comparisons when we merge them into one sorted list C. The worst case, using all $n+m-1$ comparisons actually occurs when $A[n]$ and $B[m]$ belong in the last two positions in the merged list C.

Therefore, if $W(n)$ equals the number of comparisons made in the worst case when mergesort is called with `first` $= 1$ and `last` $= n$, then it is clear from the algorithm that

$$W(n) = W\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + W\left(\left\lfloor \frac{n+1}{2} \right\rfloor\right) + \left\lfloor \frac{n}{2} \right\rfloor + \left\lfloor \frac{n+1}{2} \right\rfloor - 1, \qquad n > 1.$$

since, as above, the procedure **`merge`** performs

$$\left\lfloor \frac{n}{2} \right\rfloor + \left\lfloor \frac{n+1}{2} \right\rfloor - 1$$

comparisons in the worst case to merge two ordered lists of size

$$\left\lfloor \frac{n}{2} \right\rfloor \qquad \text{and} \qquad \left\lfloor \frac{n+1}{2} \right\rfloor .$$

Now, since

$$\left\lfloor \frac{n+1}{2} \right\rfloor = \left\lceil \frac{n}{2} \right\rceil$$

and

$$\left\lfloor \frac{n}{2} \right\rfloor + \left\lceil \frac{n}{2} \right\rceil = n$$

for $n > 1$, and since $W(1) = 0$, then $W(n)$ satisfies the discrete initial value problem

$$\begin{aligned} &W(n) = W\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + W\left(\left\lceil \frac{n}{2} \right\rceil\right) + n - 1, \qquad n > 1 \\ &W(1) = 0. \end{aligned}$$

A recursive algorithm for computing $W(n)$ for each positive integer n is given below.

```
#include<stdio.h>

int worst(int n)
{  if (n == 1)
        return 0;
   else if (n > 1)
        return worst(n/2) + worst((n+1)/2) + n - 1;
}
main()
{  int k,N,W;
   printf("Enter a positive integer N:\n");
   scanf("%d",&N);
   for (k = 1; k <= N; k++)
         printf("W(%2d) = %4d,\n",k,worst(k));
   return 0;
}
```

If we assume that $n = 2^k$, then the recurrence relation becomes

$$W(2^k) = 2W(2^{k-1}) + 2^k - 1,$$

and if we let $b_k = W(2^k)$ for $k \geq 1$, then we have

$$\begin{aligned} &b_k = 2b_{k-1} + 2^k - 1 \qquad \text{for} \qquad k \geq 1 \\ &b_0 = 0. \end{aligned}$$

We can solve this equation by iteration to get

$$\begin{aligned} b_k &= 2b_{k-1} + 2^k - 1 \\ &= 2[2b_{k-2} + 2^{k-1} - 1] + 2^k - 1 \\ &= 2^2 b_{k-2} + 2 \cdot 2^k - 1 - 2 \\ &= 2^2[2b_{k-3} + 2^{k-2} - 1] + 2 \cdot 2^k - 1 - 2 \\ &= 2^3 b_{k-3} + 3 \cdot 2^k - 1 - 2 - 2^2 \\ &\vdots \\ &= 2^k b_0 + k \cdot 2^k - 1 - 2 - 2^2 - \cdots - 2^{k-1} \\ &= k \cdot 2^k - 2^k + 1. \end{aligned}$$

Thus, $W(2^k) = k \cdot 2^k - 2^k + 1$ for $k \geq 0$, and as a it appears that

$$W(n) = n \cdot m - 2^m + 1$$

for $n \geq 1$, where $m = \lceil \log_2 n \rceil$.

In order to find the exact solution $W(n)$, we note first that the solution to the discrete initial value problem is unique. To see this, suppose that $V(n)$ is also a solution to this problem, then the function

$$U(n) = V(n) - W(n)$$

satisfies the discrete initial value problem

$$\begin{aligned} &U(n) = U\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + U\left(\left\lceil \frac{n}{2} \right\rceil\right) \qquad n > 1 \\ &U(1) = 0. \end{aligned}$$

Since $\left\lfloor \frac{2}{2} \right\rfloor = 1$ and $\left\lceil \frac{2}{2} \right\rceil = 1$, then $U(2) = 2 \cdot U(1) = 0$.

Similarly, $\left\lfloor \frac{3}{2} \right\rfloor = 1$ and $\left\lceil \frac{3}{2} \right\rceil = 2$, so that $U(3) = U(1) + U(2) = 0 + 0 = 0$.

Now a simple induction argument shows that $U(n) = 0$ for all $n \geq 1$, and the solution to this discrete initial value problem is unique.

Next we show that if $V(n)$ is given by

$$V(n) = \sum_{k=1}^{n} \lceil \log_2 k \rceil$$

for $n \geq 1$, then $V(n) = n \cdot m - 2^m + 1$, where $m = \lceil \log_2 n \rceil$.

Since $m = \lceil \log_2 n \rceil$, then

$$\log_2 n \leq m < \log_2 n + 1,$$

that is,

$$n \leq 2^m < 2 \cdot n,$$

and therefore, for $n \leq k \leq 2^m$, we have

$$\log_2 n \leq \log_2 k \leq m.$$

This implies that $m = \lceil \log_2 n \rceil \leq \lceil \log_2 k \rceil \leq m$, that is,

$$m = \lceil \log_2 k \rceil$$

for all k with $n \leq k \leq 2^m$. Therefore,

$$V(n) + (2^m - n) \cdot m = \sum_{k=1}^{n} \lceil \log_2 k \rceil + \sum_{k=n+1}^{2^m} \lceil \log_2 k \rceil = \sum_{k=1}^{2^m} \lceil \log_2 k \rceil.$$

From this, we have

$$V(n) + (2^m - n) \cdot m = \sum_{j=1}^{m} \sum_{k=2^{j-1}+1}^{2^j} \lceil \log_2 k \rceil = \sum_{j=1}^{m} j \left(2^j - 2^{j-1}\right) = \sum_{j=1}^{m} j \cdot 2^{j-1} = (m-1) \cdot 2^m + 1,$$

and thus,

$$V(n) = n \cdot m - 2^m + 1, \qquad n \geq 1$$

where $m = \lceil \log_2 n \rceil$.

Finally, we show that $V(n)$ and $W(n)$ satisfy the same discrete initial value problem. First note that

$$\left\lceil \frac{n}{2} \right\rceil = \left\lfloor \frac{n+1}{2} \right\rfloor$$

and that

$$\lceil \log_2 2j \rceil = \lceil \log_2 j \rceil + 1$$

while

$$\lceil \log_2 (2j-1) \rceil = \begin{cases} \lceil \log_2 j \rceil + 1 & \text{if} \quad j > 1, \\ \lceil \log_2 j \rceil & \text{if} \quad j = 1. \end{cases}$$

Therefore,

$$\begin{aligned} V(n) &= \sum_{1 \le j \le \frac{n}{2}} \lceil \log_2 2j \rceil + \sum_{1 \le j \le \frac{n+1}{2}} \lceil \log_2 (2j-1) \rceil \\ &= \sum_{1 \le j \le \frac{n}{2}} (\lceil \log_2 j \rceil + 1) + \sum_{2 \le j \le \frac{n+1}{2}} (\lceil \log_2 j \rceil + 1) \end{aligned}$$

that is,

$$V(n) = V\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + \left\lfloor \frac{n}{2} \right\rfloor + V\left(\left\lfloor \frac{n+1}{2} \right\rfloor\right) + \left\lfloor \frac{n+1}{2} \right\rfloor - 1,$$

that is,

$$V(n) = V\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + V\left(\left\lceil \frac{n}{2} \right\rceil\right) + \left\lfloor \frac{n}{2} \right\rfloor + \left\lceil \frac{n}{2} \right\rceil - 1 = V\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + V\left(\left\lceil \frac{n}{2} \right\rceil\right) + n - 1.$$

Clearly, $V(1) = \lceil \log_2 1 \rceil = 0$, so that $V(n) = W(n)$ for all $n \geq 1$, and therefore, my lucky guess was correct, that is,

$$W(n) = n \cdot m - 2^m + 1,$$

where $m = \lceil \log_2 n \rceil$.