CS 600 Advanced Algorithms

Question about C-1.30

C-1.30 Consider an implementation of the extendable table, but instead of copying the elements of the table into an array of double the size (that is from N to $2N$) when its capacity is reached, we copy the elements into an array with $\lceil\sqrt{N}\rceil$ additional cells, going from capacity N to $N + \lceil\sqrt{N}\rceil$. Show that performing a sequence of n add operations (that is, insertions at the end) runs in $\Theta(n^{3/2})$ time in this case.

We can use the accounting method for amortization to solve this. We view the computer as a coin operated machine that requires a payment of 1 cyber-dollar for a constant amount of computing time. We must have enough cyber-dollars to pay for an operation's running time. One way to ensure this is to overcharge "cheap" operations and undercharge "expensive" operations so as to make enough profit from the "cheap" operations in order to compensate for the "expensive" ones.

Let's start by assuming that to grow the table from some size k to size $k + \lceil\sqrt{k}\rceil$, we require $\sqrt{k}$ cyber-dollars. This accounts for the time spent copying the elements. Further, let's assume that an add operation runs in constant time (except when the table is being extended) and requires 1 cyber-dollar.

The idea is to charge the add operations more than 1 cyber-dollar so as to make enough profit to compensate for the "expensive" operation of extending the table (which requires $\sqrt{k}$ cyber dollars). How much should we charge?

One way is to ensure that the elements inserted after making an extension are overcharged so that they can cover up for the cost of rerforming that extension. So if we had k elements and we extended our table to $k + \lceil\sqrt{k}\rceil$, we would have to make a profit from all the elements that are inserted in locations $k + 1$ to $k + \lceil\sqrt{k}\rceil$ (technically, this profit must be made on the add operations that come after an extension, but you get the idea).

To do this, we must make a profit of $\frac{k+\sqrt{k}}{\sqrt{k}}$ from each operation that takes place after an extension ($k + \sqrt{k}$ is simply the size of the table, and $\sqrt{k}$ is the number of operations that must be profited from). This simplifies to a profit of $1 + \sqrt{k}$ cyber dollars from each add operation. Now, each add operation costs 1 cyber-dollar, so in order to make this profit, each add operation is charged a total of $2 + \sqrt{k}$ cyber-dollars. For n such operations, we charge,

$$\sum_{k=1}^{k=n} 2 + \sqrt{k} = 2n \sum_{k=1}^{k=n} \sqrt{k}$$

We need to establish upper and lower bounds for this sum in order to show the complexity. We can use integration for this (don't worry about the other method shown in the previous version of this solution).

A lower bound is given by,

$$\int_0^{n-1} \sqrt{k}\,\mathrm{d}k = \frac{2}{3}(n-1)^{3/2}$$

and an upper bound is given by,

$$\int_1^{n} \sqrt{k}\,\mathrm{d}k = \frac{2}{3}(n^{3/2}-1)$$

Clearly, this is $\Theta(n^{3/2})$.