A NOTE ON WEIGHTED BUDDY SYSTEMS FOR
DYNAMIC STORAGE ALLOCATION

HARRY T. HSU

Abstract.

In this paper we first show that the Shen and Peterson System [1] is a combination of two Fibonacci buddy systems [2, 3]. Then we present a generalized Shen and Peterson system. The generalized system has more flexibility in the generation of fixed size blocks. For the generalized system, the algorithm for the determination of the buddy of any released block is easier than that of the Shen and Peterson system.

Keywords: Weighted buddy system, dynamic storage allocation, Shen and Peterson system, Fibonacci buddy system, internal fragmentation.

CR: 3.89, 4.32, 4.39.

1. Introduction.

The buddy system [4, 5, 6] is a nice algorithm for dynamic storage allocation, but it suffers from a serious degree of internal fragmentation. The Shen and Peterson system [1] presents a modified form of the buddy system. It increases the number of fixed size blocks to almost two times as many as the buddy system, and reduces the degree of internal fragmentation to almost one-half that of the buddy system.

This paper shows that the Shen and Peterson system is actually a combination of two Fibonacci systems [2, 3] and we present a generalization of the Shen and Peterson system. The generalized system has more flexibility in the generation of the fixed size blocks.

The present paper assumes that the reader is familiar with the buddy system [4, 5, 6] and the Shen and Peterson system [1]. We have tried to keep all the symbols which are used in this paper as close as possible to those of Shen and Peterson. Unless indicated otherwise, all the symbols used in this paper have exactly the same meaning as those of Shen and Peterson.

For all the splitting diagrams which are shown in this paper, we always use an a bit and a b bit to show the parity of a block. The meaning of the a and b bits is exactly as defined by Shen and Peterson. Since the algorithms to reserve and to release a block are very simple, no details will be repeated in this paper.

Received August 3, 1976
2. Presentation of Results.

In figure 1, we show a comparison of the buddy system and the Shen and Peterson system. The Shen and Peterson system always adds one additional block size \(3 \cdot 2^{(j-1)}\) to every two adjacent block sizes \(2^j\) and \(2^{j+1}\) (where \(j \geq 1\)) in the buddy system. A careful study of figure 1 shows that Shen and Peterson use the following recurrence formula for the generation of the set of all the fixed size blocks.

\[
\begin{align*}
    a_{2t} &= a_{2t-1} + a_{2t-4} \\
    a_{2t-1} &= a_{2t-2} + a_{2t-4}
\end{align*}
\]

and \(a_0 = 2, a_1 = 3, a_2 = 4, a_3 = 6\). Therefore the Shen and Peterson system is actually a combination of two Fibonacci buddy systems [2, 3].

A generalization of the Shen and Peterson system is to use the following recurrence formula for the generation of all blocks.

\[
\begin{align*}
    a_{2t} &= a_{2t-1} + a_{2t-l_1} \\
    a_{2t-1} &= a_{2t-2} + a_{2t-l_2}
\end{align*}
\]

where \(l_1\) is an even and \(l_2\) an odd integer. It is found that only if \(l_1\) is even and \(l_2\) odd, then the system can be easily implemented. The size of the initial blocks can be any arbitrary number.

Since \(l_1\) and \(l_2\) can be any arbitrary number so long as \(l_1\) is even and \(l_2\) odd, the designer of the system has more flexibility in the generation of fixed size blocks. The generalized system will allow us to generate more fixed size blocks and have the block sizes distributed more uniformly so that the internal fragmentation will be smaller than in the Shen and Peterson system.

The generalized system can be implemented by using roughly the same scheme as the Shen and Peterson system except that now we require additional storage space to store the one-dimensional array \(\text{SIZE}(j)\), \(1 \leq j \leq \text{max}\). Each block requires a \(tag\) bit, \(a\) bit, \(b\) bit, and an oval field. The definitions of \(tag\) bit, \(a\) bit and \(b\) bit are exactly the same as in the