A NOTE ON THE STACK SIZE
OF REGULARLY DISTRIBUTED BINARY TREES

R. KEMP

Abstract.
Assume that in one unit of time a node is stored in the stack or is removed from the top of the stack during postorder-traversing of a binary tree. If all binary trees are equally likely the average stack size after \( t \) units of time and the variance is computed as a function of the proportion \( \varrho = \frac{t}{n} \).

1. Introduction.
Let \( T(n) \), \( n \in \mathbb{N} \), be the set of all extended binary trees ([6]) with \( n \) leaves and \( T \in T(n) \). The stack size \( S(T) \) is recursively defined by

\[
S(T) := \begin{cases} 
1 & \text{IF } |T| = 1 \\
\text{IF } S(T_1) > S(T_2) & \text{THEN } S(T_1) \\
\text{ELSE } S(T_2) + 1 & \text{ELSE } S(T_2) + 1;
\end{cases}
\]

where \( |T| \) is the number of nodes of the tree \( T \) and \( T_1 \) (\( T_2 \)) is the left (right) subtree of \( T \). \( S(T) \) is the maximum number of nodes stored in the stack during postorder-traversing of \( T \in T(n) \). In [4] it is implicitly shown that the average stack size of a binary tree \( T \in T(n) \) is asymptotically given by \( \sqrt{\pi n} - \frac{1}{2} + O(\ln n/\sqrt{n}) \) assuming that all \( n \)-node trees are equally likely. The variance is computed in [5] and is asymptotically given by

\[
(\pi/3 - 1)\pi n + \frac{1}{12} - \frac{1}{8}\pi^2 + \frac{1}{12}\pi + O(\ln n/\sqrt{n^3}) \text{ for all } \epsilon > 0.
\]

In this paper we consider an analogous problem. Evaluating a binary tree \( T \in T(n) \) in postorder we assume that in one unit of time a node is stored in the stack or is removed from the top of the stack. Considering all trees \( T \in T(n) \) equally likely we shall compute the average number of nodes \( R_1(n, t) \) stored in the stack after \( t \) units of time as a function of the proportion \( \varrho = \frac{t}{n} \). Moreover, we give an asymptotic equivalent for the \( s \)th moment \( R_s(n, t) \) with respect to the origin, and for the variance.

2. The average stack size after \( t \) units of time.
Obviously, each path from \( (t, k) = (1, 1) \) to \( (t, k) = (2n-1, 1) \) in Figure 1 corresponds to the evaluation of a binary tree \( T \in T(n) \) in postorder ([6; p. 316]); for example, the marked path in Figure 1 corresponds to the following tree \( T \in T(6) \).

Part of this paper was presented at the 6th Colloquium on Automata, Languages and Programming, ICALP' 79, July 1979.

Received October 5, 1979.
If we reach the point \((i,j)\) then we have exactly \(j\) nodes in the stack after \(i\) units of time.

Now, let \(H(n,k,t)\) be the number of binary trees \(T \in T(n)\) having exactly \(k\) nodes in the stack after \(t\) units of time. An inspection of Figure 1 shows that this number is the product of

(a) the number of paths from \((1,1)\) to \((t,k)\), which is

\[
\frac{k}{t} \left( \frac{t}{(t+k)/2} \right),
\]

and

(b) the number of paths from \((t,k)\) to \((2n-1,1)\), which is

\[
\frac{k}{2n-t} \left( \frac{2n-t}{n-(t+k)/2} \right).
\]

These enumeration results of the number of paths are well-known ([3]). Hence

\[
H(n,k,t) = \frac{k^2}{t(2n-t)} \left( \frac{t}{(t+k)/2} \right) \left( \frac{2n-t}{n-(t+k)/2} \right).
\]

Obviously, we have the conditions \(k \leq t \leq 2n-1\) and \((k+t) \equiv 0 \mod 2\). Now, let \(|T(n)| = t(n)\). It is well-known ([6]) that

\[
t(n) = \frac{1}{n} \left( \frac{2n-2}{n-1} \right).
\]

Considering all binary trees \(T \in T(n)\) equally likely the quotient \(p(n,k)\)