

Optimal BST Average Search Time Algorithm

- $p[k] = p_k$ and $q[k] = q_k$.
- $AST[i, j] = AST(i, j)$ and $w[i, j] = w(i, j)$.
- $root[i, j]$ ($1 \leq i \leq j \leq n$) holds r such that k_r is the root node of an OptBST containing keys k_i through k_j .

```

1 ComputeOptBST (p,q,n)
2 -----
3   loop i <-- 1 through n+1:
4     AST[i,i-1] <-- q[i-1]
5     w[i,i-1] <-- q[i-1]
6   endloop
7
8   loop l <-- 1 through n:
9     loop i <-- 1 through n - l + 1:
10      j <-- i + l - 1
11      AST[i,j] <-- +infinity
12      w[i,j] <-- w[i,j-1] + p[j] + q[j]
13      loop r <-- i through j:
14        e <-- AST[i,r-1] + w[i,j] + AST[r+1,j]
15        if e < AST[i,j] then
16          AST[i,j] <-- e
17          root[i,j] <-- r
18        endif
19      endloop
20    endloop
21  endloop
22
23  return AST and root

```

