Algorithm 3 predecessor(T, k)

```
1: x=T.root
2: y=nil
3: i=1
4: j=1
5: while true do
6:     i=1
7:     while i<=x.n and k>x.key_i do
8:         i+=1
9:     end while
10:    if i!=1 then
11:        y=x
12:        j=i
13:    end if
14:    if i<=x.n and k==x.key_i then
15:        break
16:    end if
17:    x=x.c_i
18: end while
19: if x.leaf then
20:    if i==1 then
21:        if j==1 then
22:            return NO PREDECESSOR
23:        else
24:            return y.key_{j-1}
25:        end if
26:    else
27:        return x.key_{i-1}
28:    end if
29: else
30:    return maximum(x.c_n)
31: end if
```