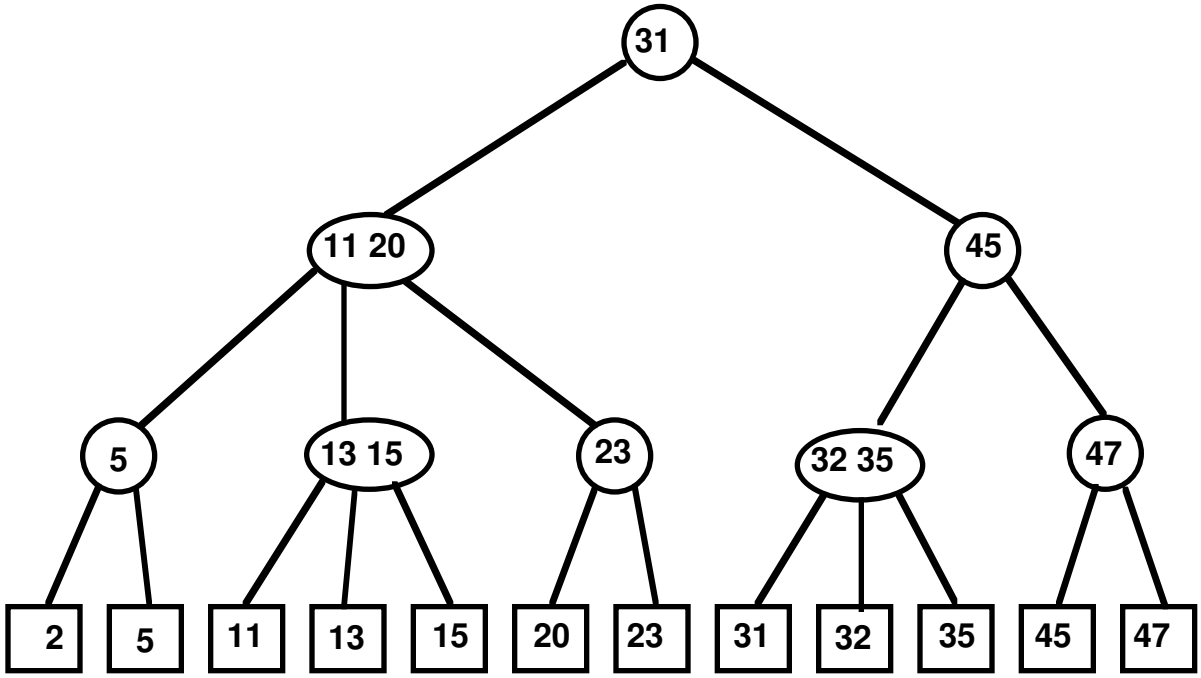
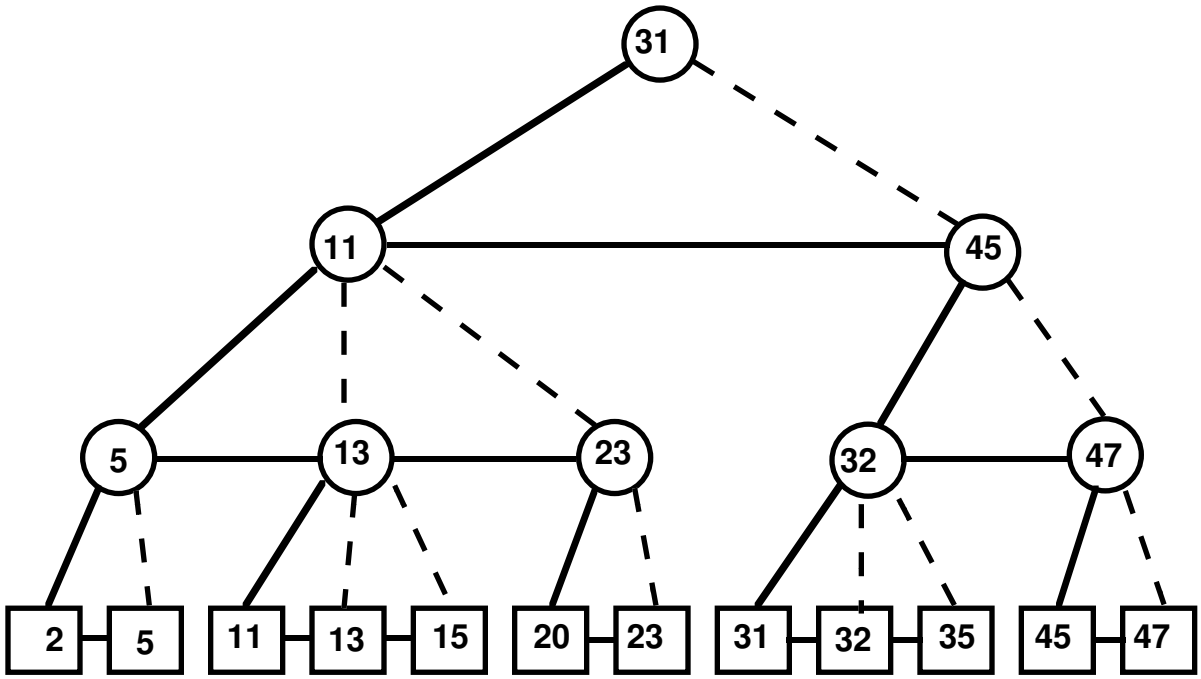
© Itai February 22, 1996

רשימת דילוגים דטרמיניסטית -
Deterministic Skip List

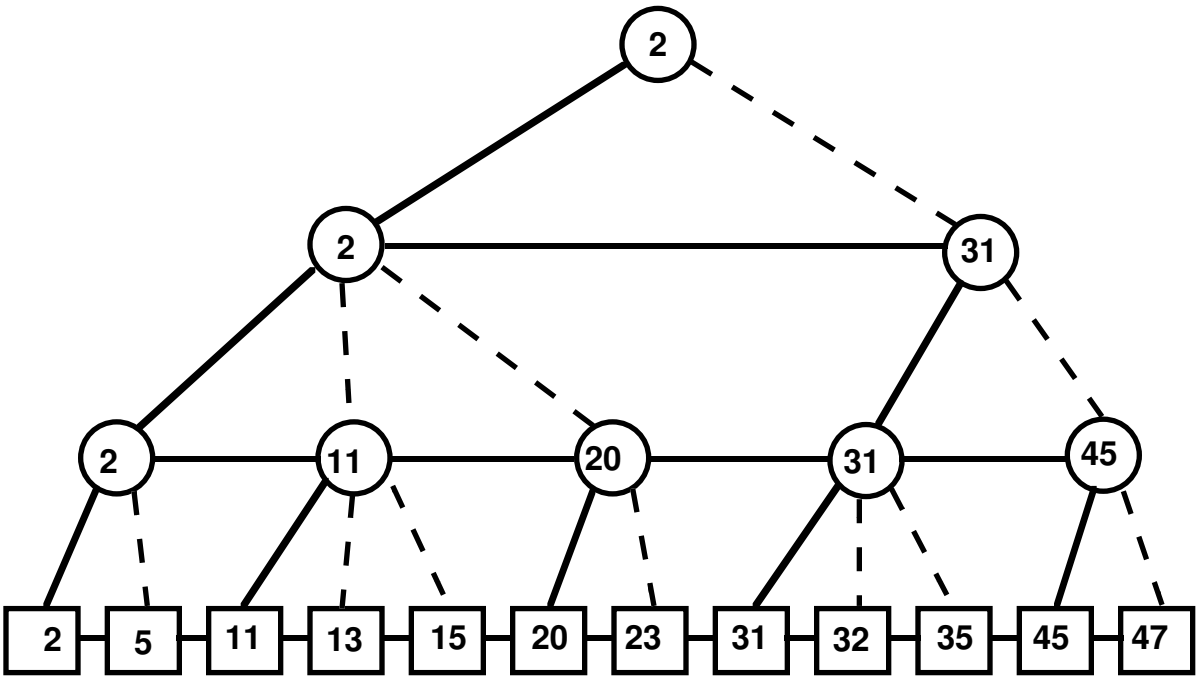
עץ 2-3



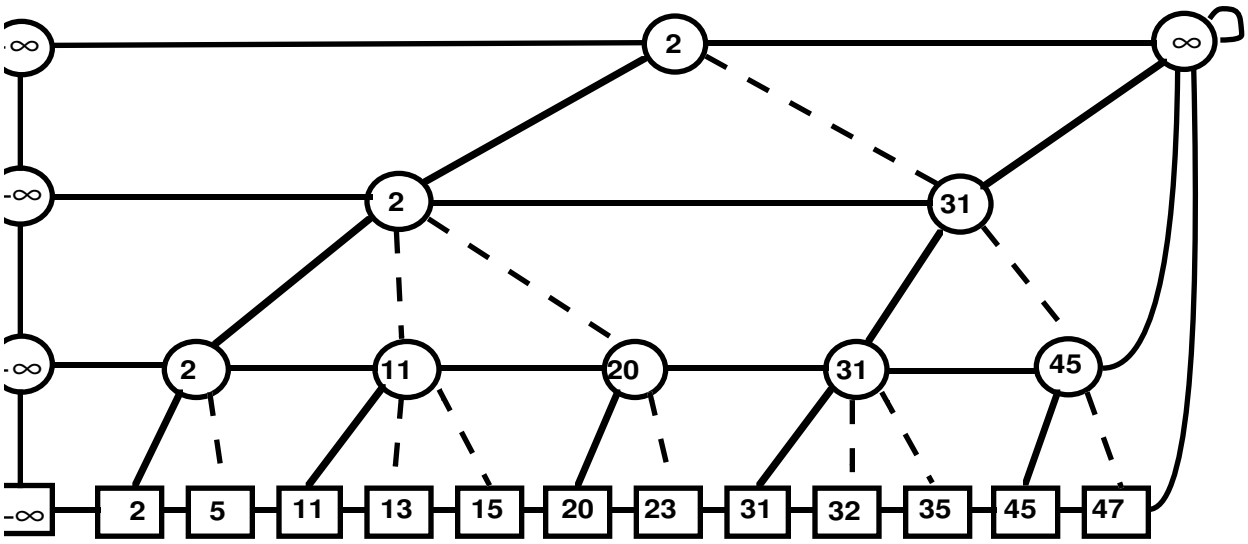
מימוש עץ 2-3 כעץ בינרי



חיבור הצמתים של כל רמה נותן skip list



הוספת איבר ראשון ואיבר סופי



תכניות

```
/* deterministic skip lists*/

typedef struct  node {
        int value;
    struct node *down, *next;
    } NODE ;
NODE *top;

NODE *get_node(int x, NODE *n, NODE *d)
{   NODE *p;
    p=(NODE *)malloc(sizeof(NODE));
    p->value=x;
    p->down=d;
    p->next=n;
    return p;
}

NODE *search(int x, NODE *T)
{   /*  returns a ptr to last node ≤ x   */
    while(1){
        for(; T->next->value <= x;
                                T=T->next)
            ;
        if(T->down == NULL)  /* leaf */
            break;
        T = T->down;
    }
    return T;
}
```

```
int insert1(int x,    NODE *T)
{
/*  returns 1 iff a new node was allocated on this level
*/
    NODE *q;

    for(; T->next->value <= x; T=T->next)
        ;
    if(!T->down){
        T->next = get_node(x,T->next,
NULL);
        return 1;
    }
    if(!insert1(x, T->down)) return 0;
    /*  check if as result of the insertion,
        there are three nodes between
        T->down and T->next->down*/
    q = T->down->next->next;
    if( q->next->value < T->next->value){
        T->next = get_node(q->value,
                           T->next, q);
        return 1;
    }
    return 0;
}

NODE *insert(int x,    NODE *T)
{
     /*  top level,
         returns first node of the top level list */
    if(insert1(x,T))
        return get_node(T->value, top, T);
    return T;
}
```