6. There are $n-1$ internal vertices in a complete binary tree with $n$ leaves.

7. The binary search tree requires $n$ nodes. Each node contains one integer value as key and two pointers to its left and right child respectively.

$$size = n \times (\text{sizeof(key)} + 2 \times \text{sizeof(pointer)}) = n \times (4 + 2 \times 4) = 12n \text{ bytes}$$

8. 

9. Type lexeme is record

   start: …
   stop: …
   typ: …
   value: integer; -- value

   end record

procedure scan is
begin

   …
   elseif line(pos) in ‘0’..’9’ then
      next.typ = number;
      next.value = 0;
   loop
next.value = next.value*10 + line(pos) – ‘0’;
pos:=pos+1;
exit when line(pos) not in ‘0’..’9’;
elseif line(pos) = ‘#’ then
    next.typ := number;
    next.value = 0;
end loop

    pos:=pos+1;
    if linebuf(pos) in ‘0’..’9’ then
        next.value = next.value*16 + line(pos) – ‘0’;
    elseif line(pos) in ‘A’..’F’ then
        next.value = next.value*16 + line(pos) – ‘A’+10;
    exit when linebuf(pos) not in ‘0’..’9’
    and when linebuf[pos] not in ‘A’..’F’
endloop

…
end scan;

(note: the bold italic are added codes)