

# S10 – SD

Alin Zamfiroiu

[alin.zamfiroiu@csie.ase.ro](mailto:alin.zamfiroiu@csie.ase.ro)

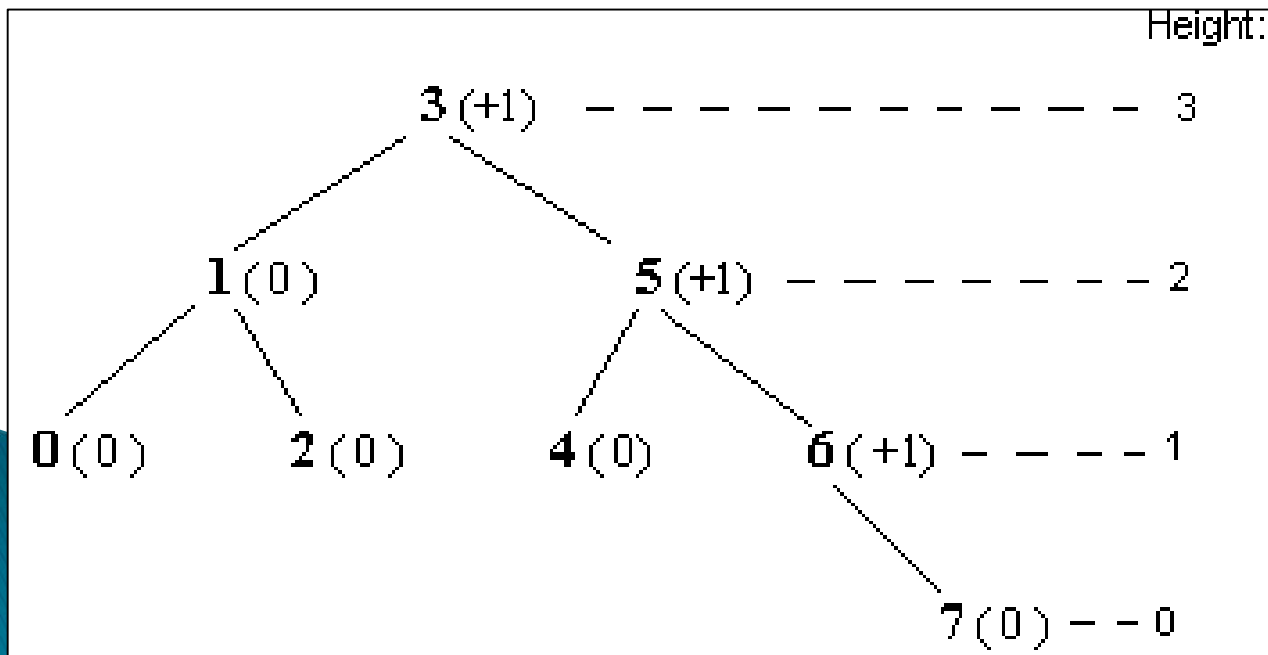
# S10 – AVL

```
struct domeniu
{
    int id;
    char* nume_domeniu;
};
```

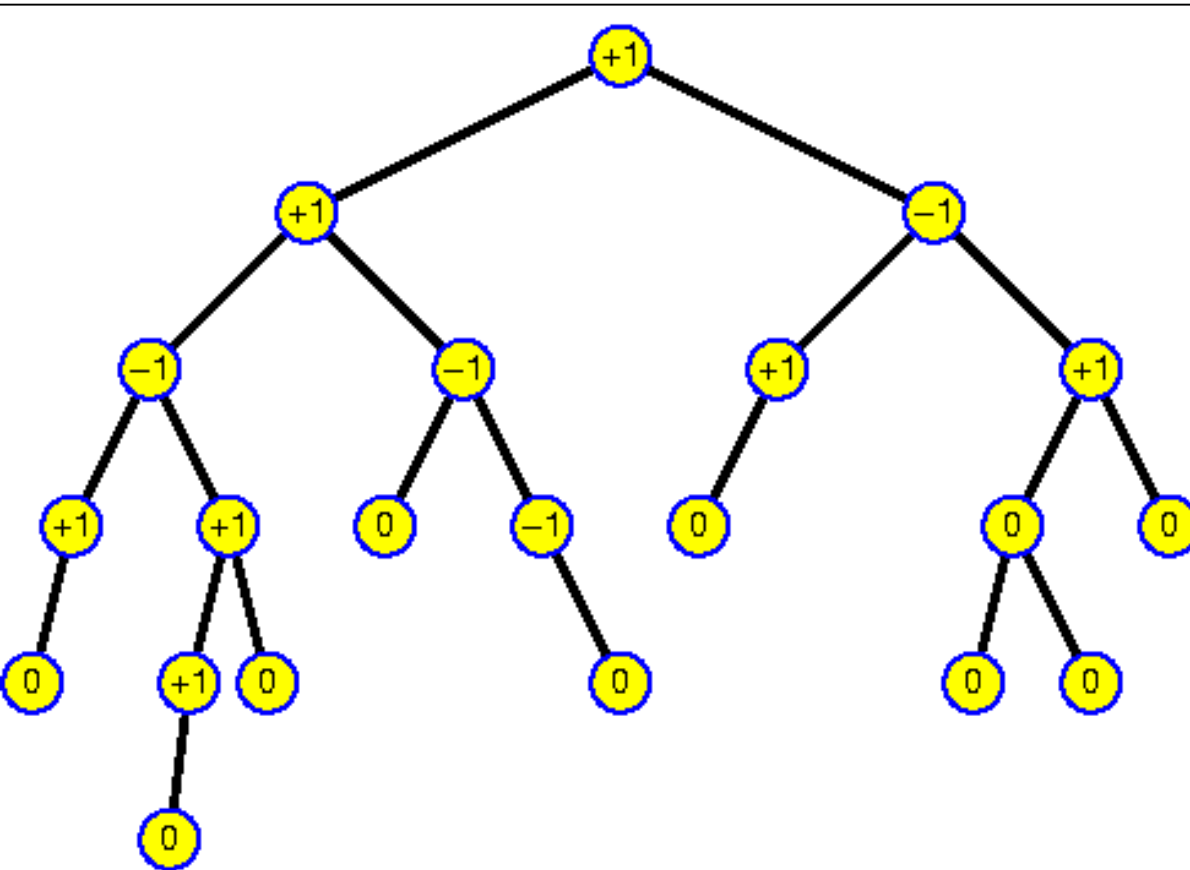
- ▶ We will use the structure of a binary search tree.
- ▶ Implement the method to insert a new node in a binary search tree.

# S10 - AVL

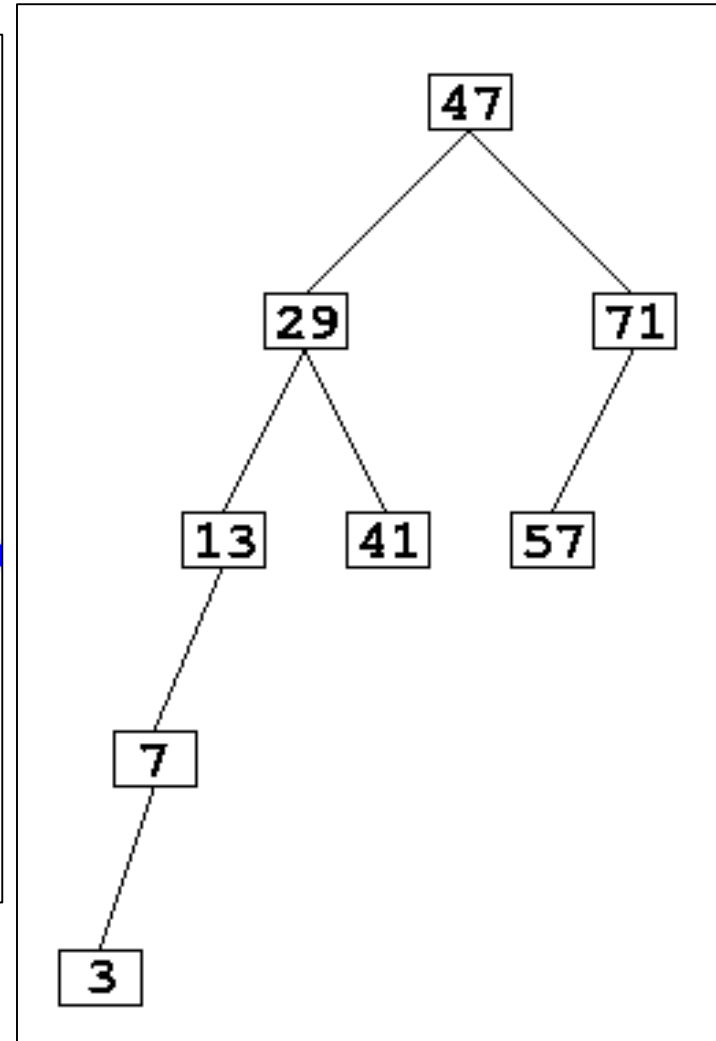
- ▶ Implement the method to determine the height of a tree.



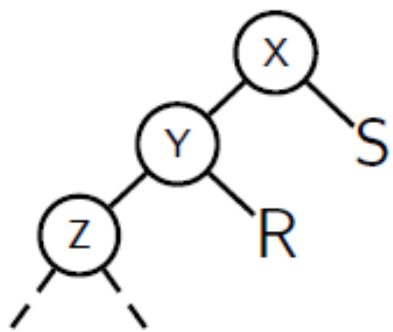
# S10 - AVL



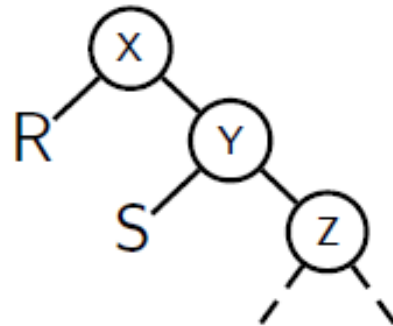
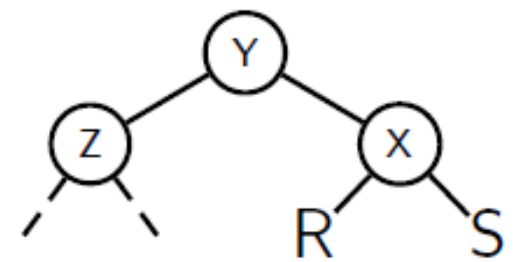
<http://users.informatik.uni-halle.de/>



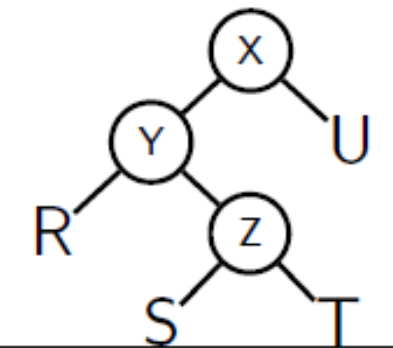
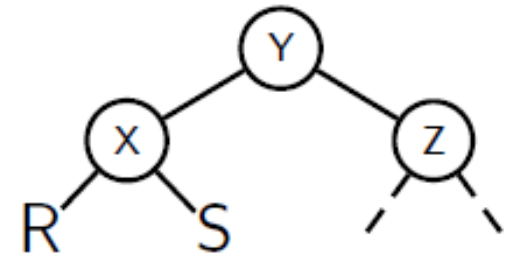
<http://pages.cs.wisc.edu/>



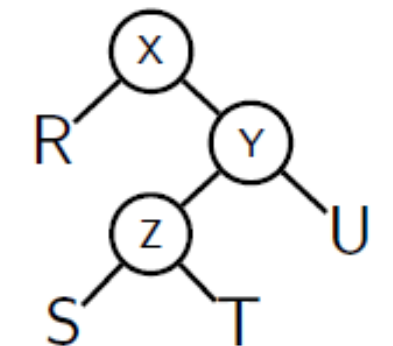
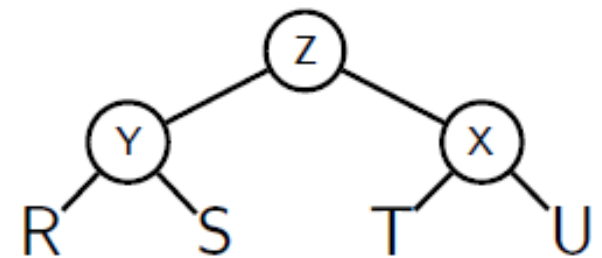
right-rotate(x)  $\rightarrow$



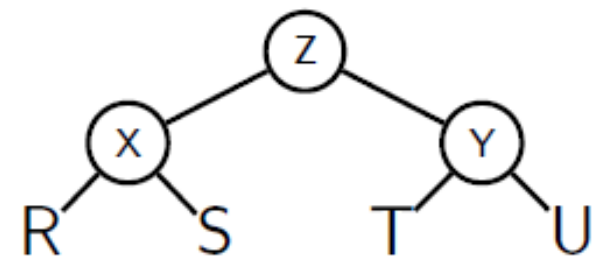
left-rotate(x)  $\rightarrow$



right-double-rotate(x)  $\rightarrow$



left-double-rotate(x)  $\rightarrow$



# S10 - AVL

case 2:

```
    { //avem arbore dezechilibrat in partea stanga
      nod_arbore * aux;
      aux=rad->st;
      if(factor_balansare(aux)==1)
      {
          //este cazul simplu in care avem doar rotatie dreapta
          rad->st=aux->dr;
          aux->dr=rad;
          rad=aux;
      }
    }
```

# S10 - AVL

```
else
{
    //este cazul complex in care avem dubla rotatie(stanga-dreapta)
    //stanga
    nod_arbore* aux2=aux->dr;
    aux->dr=aux2->st;
    aux2->st=aux;
    //dreapta
    rad->st=aux2->dr;
    aux2->dr=rad;
    rad=aux2;
}
}
```