Classes:

Player:

* Level
* id
* c’tor
* copy c’tor
* d’tor
* set level
* get level
* ==
* <=

Group:

* AVLTree<Players>
* Size
* Group id
* Add player
* Remove player
* CombGroup(Group)
* Operator == , operator <=
* Return Highest Player

Node:

* Node\* Father
* Node\* LSon
* Node\* RSon
* T\* Value
* Int height
* Node\* root;// for player, he knows in what group he is .

AvlTree:

* Node\* Root
* Insert(T value);
* Find(T value);
* Remove(T value);
* LL,LR,RL,RR, RollChecker(Node\* father)
* Tour(preorder,**inorder**,postorder)
* CombineTrees.(friend non-member method).
  + יצירת שני מערכים בגדלים מתאים, סיור "לפי סדר", מיזוג שני המערכים למערך אחד גדול, יצירת עץ ריק בגודל שני המערכים ביחד, מעבר על המערך הממוזג והעץ במקביל והכנסה "לפי סדר"
* D’tor,C’tor,Cp’tor(?)
* = =delete