**public** **class** Author {

**private** String name;

**private** **int** age;

**public** Author(String name, **int** age) {

**this**.name = name;

**this**.age = age ;

}

**public** **int** getAge() {

**return** age;

}

}

**public** **class** Paper {

**private** String title;

**private** **int** nbWords;

**private** Author arAuth[];

**private** **int** nba;

**public** Paper(String title, **int** nbWords) {

**this**.title = title;

**this**.nbWords = nbWords;

arAuth = **new** Author[5];

nba = 0;

}

**public** Paper(Paper P) {

**this**.title = P.title;

**this**.nbWords = P.nbWords;

arAuth = **new** Author[P.arAuth.length];

**for**(**int** i=0; i < nba; i++)

arAuth[i] = P.arAuth[i];

nba = P.nba;

}

**public** **boolean** addAuthor(Author a) {

**if** (nba < arAuth.length) {

arAuth [nba] = a; //aggregation

nba ++;

**return** **true**;

}

**else**

**return** **false**;

}

**public** Author findYoungestAuthor() {

Author res = arAuth[0];

**for** (**int** i = 1; i < nba; i++) {

**if** (arAuth[i].getAge() < res.getAge())

res = arAuth[i];

}

**return** res;

}

**public** **int** getNbWords() {

**return** nbWords;

}

**public** **int** countAuthors(**int** a){

**int** count =0;

**for** (**int** i=0; i < nba; i++) {

**if** (arAuth[i].getAge() >= a)

count++;

}

**return** count;

}

}

**public** **class** Conference{

**private** String name;

**private** String location;

**private** Paper arPap[];

**private** **int** nbp;

**public** Conference(String name, String location, **int** size){

**this**.name = name;

**this**.location = location;

arPap = **new** Paper[size];

nbp = 0;

}

**public** **boolean** addPaper(Paper p) {

**if** (nbp < arPap.length) {

arPap[nbp] = **new** Paper(p); //composition

nbp++;

**return** **true**;

}

**else**

**return** **false**;

}

**public** **void** splitPaper(**int** n, Paper[]longPapers, Paper[]shortPapers)

{

**int** j=0, k=0;

**for** (**int** i = 0; i < nbp; i++){

**if** (arPap[i].getNbWords()> n) {

longPapers[j] = arPap[j];

j++;

}

**else**

{

shortPapers[k] = arPap[k];

k++;

}

}

}

**public** **int** countSeniorAuthors(){

**int** count =0;

**for** (**int** i = 0; i < nbp; i++){

count += arPap[i].countAuthors(50);

}

**return** count;

}

}