

X22408 ca

Preliminars: Recordeu que el recorregut en amplada d'un arbre és la llista dels nodes de l'arbre per nivells, és a dir, primer l'arrel, després els nodes a profunditat 1 d'esquerra a dreta, després els nodes a profunditat 2 d'esquerra a dreta, després els nodes de profunditat 3 d'esquerra a dreta, i així successivament.

Haureu d'implementar una funció **RECURSIVA** que rebrà un arbre binari d'enters. La funció haurà de retornar un altre arbre binari d'enters, amb exactament la mateixa estructura (conjunt de posicions) que el que s'ha rebut d'entrada, i a on cada node guardarà la posició d'aquell node en el recorregut en amplada de l'arbre.

Aquí tenim un exemple de comportament de la funció:

[illegible]

Fixeu-vos que l'enunciat d'aquest exercici ja ofereix uns fitxers que haureu d'utilitzar per a compilar: `main.cc`, `BinTree.hh`, `breadthTree.hh`. Només cal que creeu `breadthTree.cc`, posant-hi els includes que calguin i implementant la funció `breadthTree`. Només cal que pugueu `breadthTree.cc` al jutge.

Entrada

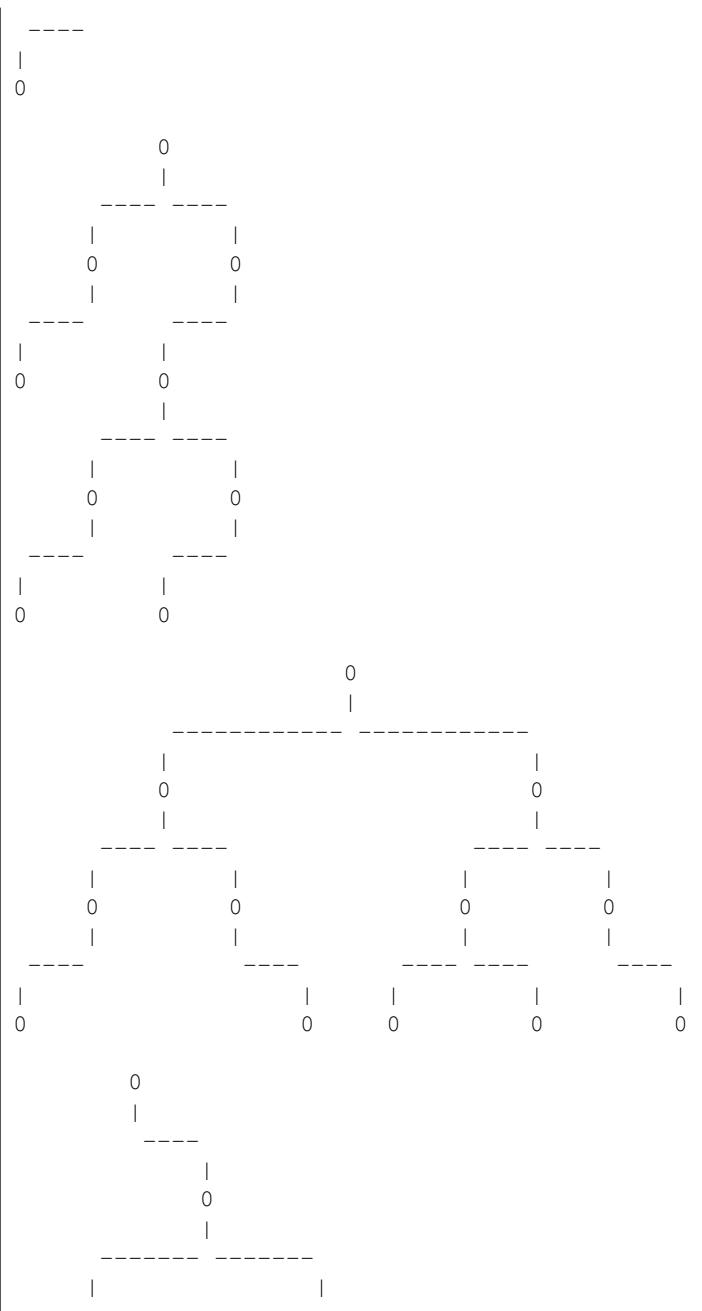
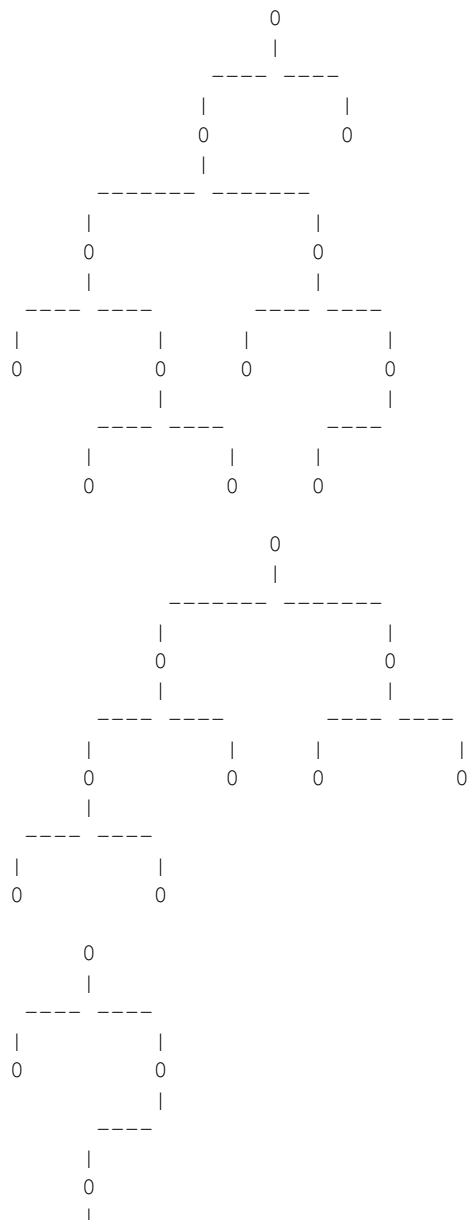
La primera línia de l'entrada descriu el format en el que es descriuen els arbres, o bé IN-LINEFORMAT o bé VISUALFORMAT. Després venen un nombre arbitrari de casos. Cada cas consisteix en una descripció d'un arbre binari d'enters (amb només el valor 0 als nodes, tot i que això és irrellevant). Fixeu-vos en que el programa que us oferim ja s'encarrega de llegir aquestes entrades. Només cal que implementeu la funció abans esmentada.

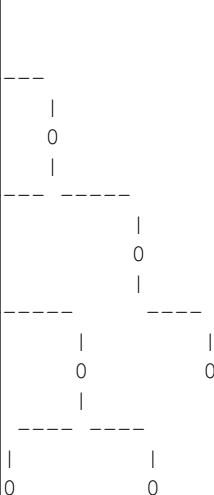
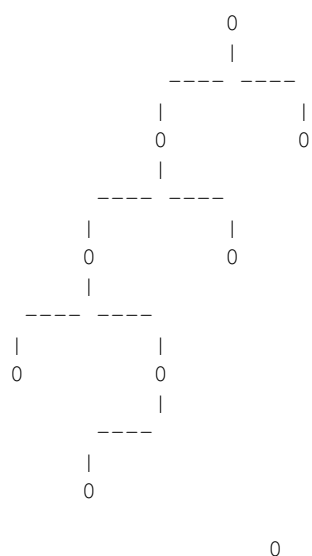
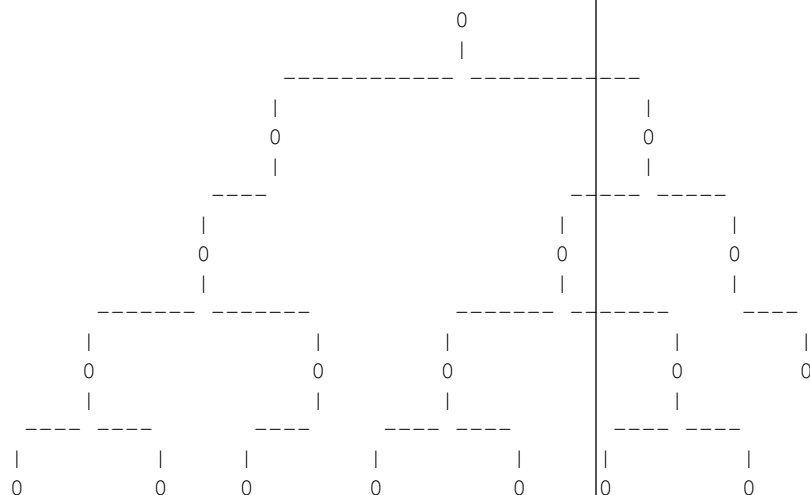
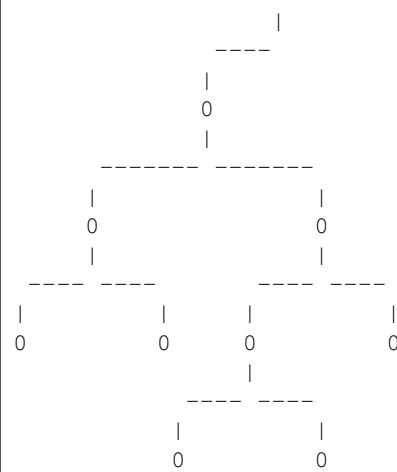
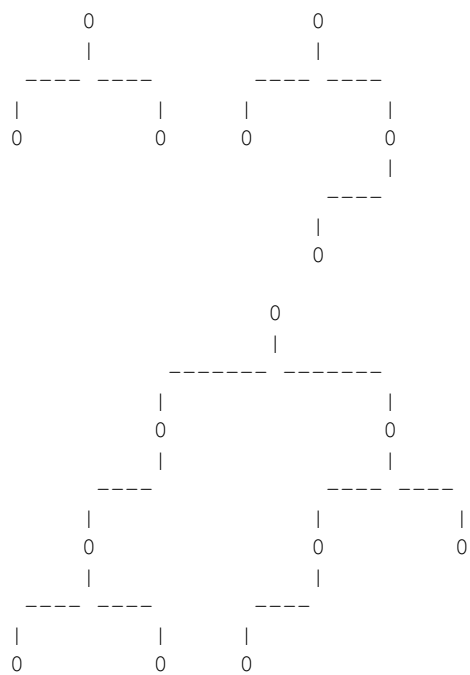
Sortida

Per a cada cas, cal escriure l'arbre binari resultant de cridar a la funció abans esmentada amb l'arbre d'entrada. Fixeu-vos en que el programa que us oferim ja s'encarrega d'escriure aquesta sortida. Només cal que implementeu la funció abans esmentada.

Exemple d'entrada 1

VISUALFORMAT





0

Exemple de sortida 1

