**2 METHODE GLOUTONNE**

Non elle n’est pas optimale étant donné qu’il suffit que le premier pavé ait une surface petite pour que l’empilement soit bloqué. Par exemple avec un pavé de longueur : 5 et de profondeur : 2. Elle ne pourra qu’empiler une seule boite par exemple [46, 4, 1]. Pour que notre algorithme glouton face bien son travail il faut y ajouter des sous problème, mais dans c’est cas on obtient un algorithme Bottom Up.

**3 RECURRENCE ET RECURSIVITE NAÏVE**

Soit h(i) la valeur de la hauteur de la ième boite empilé, l(i) la longueur et p(i) la profondeur. Si H(i) est la hauteur maximale de l’empilement alors on peut établir la formule de récurrence suivante : H[i] = h(i)+H[i-1] if l(i) > l(i-1) and p(i) > p(i-1)

H(0) = 6 (on le trouve grâce à notre ancien algorithme) .

La complexité de notre algorithme est T(n) = 3n-1, on répète n foi une comparaison, une affectation et une addition (sauf quand i == 1 d’où le -1).

**4.2 APPROCHE BOTTOM UP**

La complexité de notre algorithme est :

Trois affectations -> +3

Une boucle -> 1\*n

Une comparaison -> (1+1)\*n

Une affectation -> (1+2)\*n

Une autre boucle -> +1\*n²

Une comparaison -> (1+1)\*n²

Une boucle de rangé 3-> \*3

Et encore une boucle de rangé 3 -> \*3

Trois comparaisons -> (3+1)\*n^4

Six opérations -> (6+4)\*n^4

Quartes affectations -> (4+10)\*n^4

Une et demi comparaisons -> (3/2+2)\*n

Sept affectations -> (7+7/2)\*n

Deux opérations -> (2 + 21/2)\*n

Et enfin une affectation 1+3

Ce qui nous donne : T(n) (25/2)n + 42n² + 4