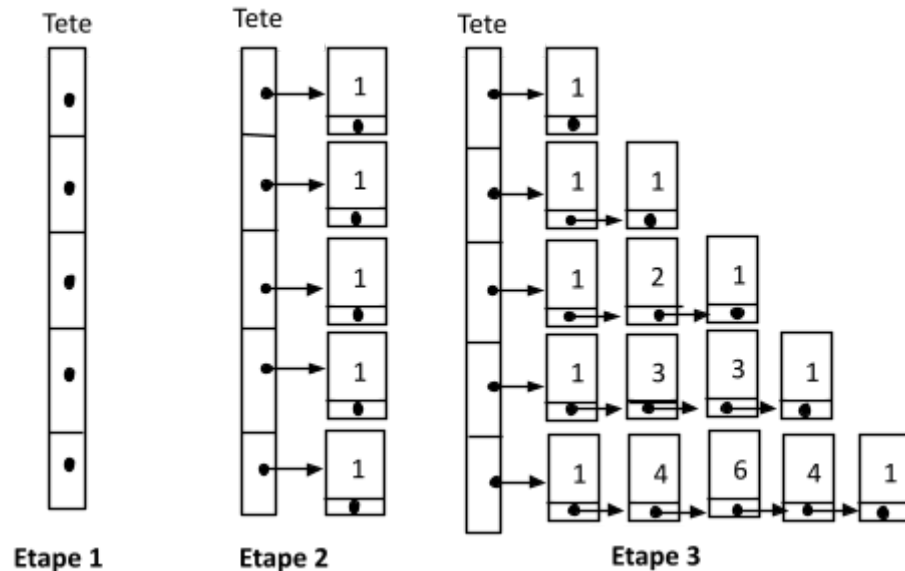


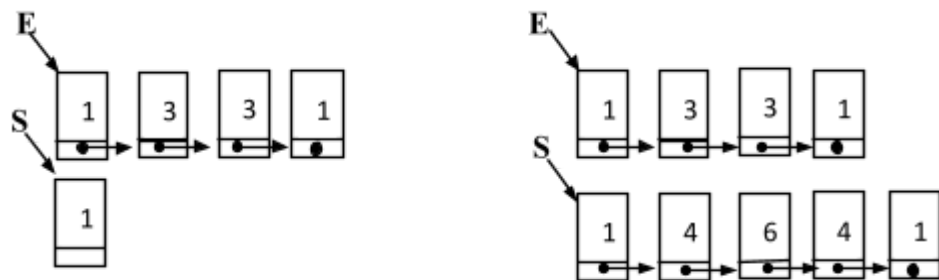
Exercice 1 :

On veut construire le triangle de Pascal pour un nombre de lignes donné en utilisant un tableau Tete et plusieurs listes chaînées. Les étapes à suivre sont données comme suit :



1. Initialisez le tableau Tete[100] et donnez son type (voir Etape 1).
2. Créez la première colonne du triangle avec la valeur 1. Mettez l'adresse de cet élément dans le tableau Tete (voir Etape 2). Le prototype de cette fonction est le suivant :
3. Écrire une procédure qui, à partir de la ligne n du triangle, calcule les valeurs de la ligne $n+1$ (voir la figure ci-dessous). Le prototype de cette procédure est le suivant :

Procédure Const_ligne_svt(?? E, ?? S)



· Avant l'appel de la fonction

4. Écrire une Action qui affiche les éléments d'une ligne du triangle.
?? Affiche_ligne (??)
5. On souhaite imprimer, parmi les 100 premières lignes du triangle, toutes celles qui ne contiennent que des valeurs strictement inférieures à un nombre strictement positif donné nb ($nb \geq 1$). On vous demande d'écrire une Action **F()** qui utilise les actions **Affiche_ligne()** et **Const_ligne_svt()** pour calculer et afficher une à une les lignes du triangle jusqu'à la première ligne contenant une valeur supérieure ou égale au nombre nb

donné ($nb \geq 1$). Cette ligne ne sera pas affichée. Si on atteint la ligne 100 sans avoir trouvé, on s'arrête.

6. Ecrire une action **Liberer ()** qui libère l'espace mémoire alloué au triangle dans le but de retrouver la configuration donnée par l'étape 2.
7. Écrire le programme C qui construit et affiche le triangle pour un nombre de lignes donné ($nb_ligne \leq 100$) en utilisant les deux méthodes :
 - `Const_ligne_svt()`, `Affiche_ligne()`
 - `Liberer ()`, `F ()`

Exercice 2 :

