

## Bases de la conception orientée objet Test Structurel

Toutes les réponses sont à effectuer sur la feuille d'énoncé aussi n'oubliez pas d'inscrire votre nom en haut à droite de la première feuille et en bas à gauche des autres feuilles. Rendre l'intégralité de l'énoncé avec la copie

### Exercice 1 : Un choix difficile

Sur le programme suivant, je vous propose plusieurs graphes de flot de contrôle. Un seul correspond aux consignes qui ont été précisées lors des séances de TD.

Cette fonction est issue d'un projet réalisé par vos pairs. Elle traite du placement des bateaux dans une grille 10 x 10 de bataille navale en vérifiant qu'il n'y a pas de collisions et que le bateau ne débord pas de la grille.

```
public int placerObjet(Position position, Bateau bateau, int sens) {
    int taille = bateau.getSize() - 1;
    if((sens==1 && position.getX()+taille >9) || (sens==2 && position.getX()-taille <0))
        return 0;
    else if((sens==1 && position.getY()-taille <0) || (sens==2 && position.getY()+
        taille >9))
        return 0;

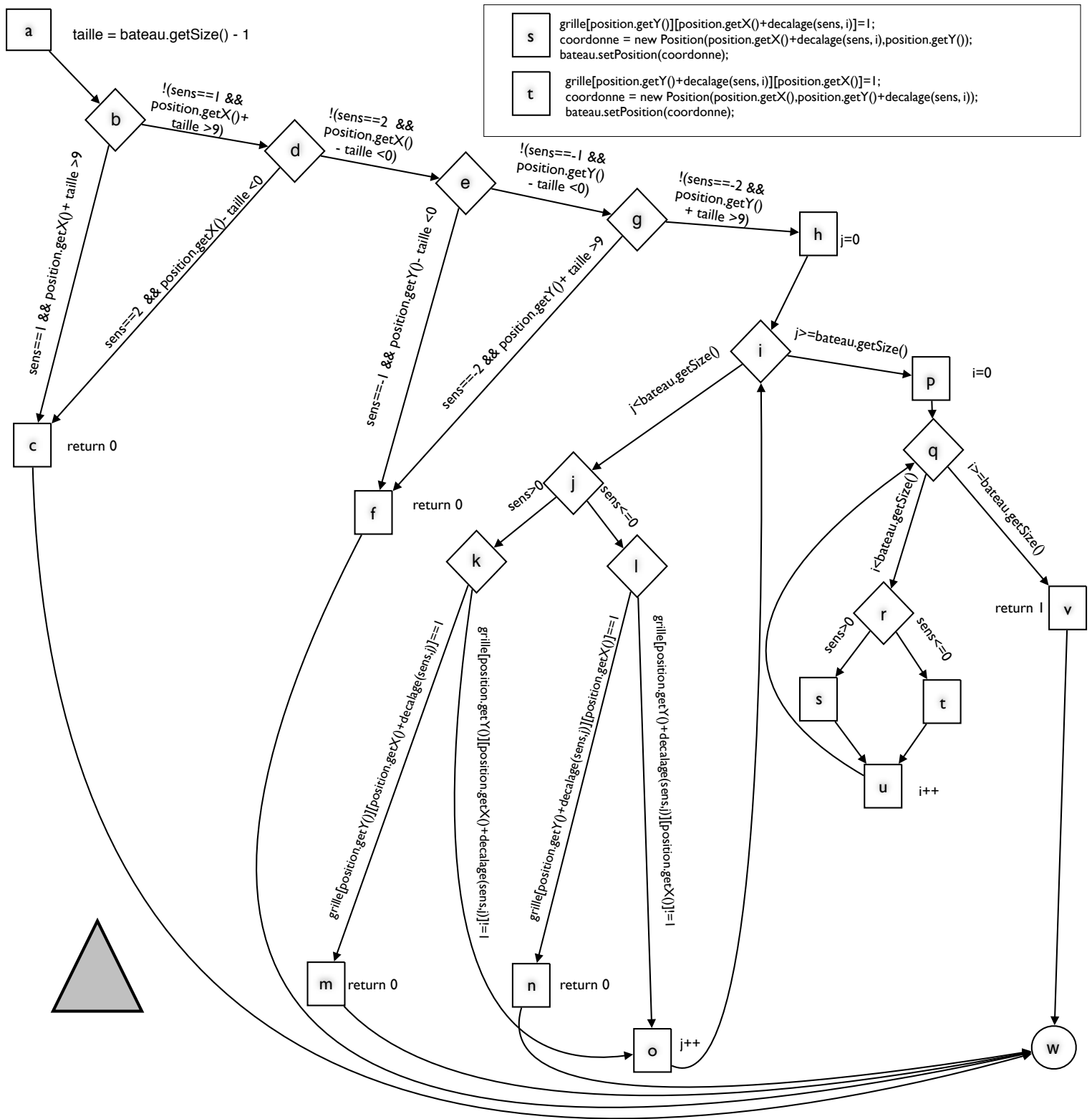
    for(int j=0; j<bateau.getSize(); j++) {
        if(sens>0) {
            if(grille[position.getY()][position.getX()+decalage(sens, j)]==1)
                return 0;
        } else {
            if(grille[position.getY()+decalage(sens, j)][position.getX()]==1)
                return 0;
        }
    }

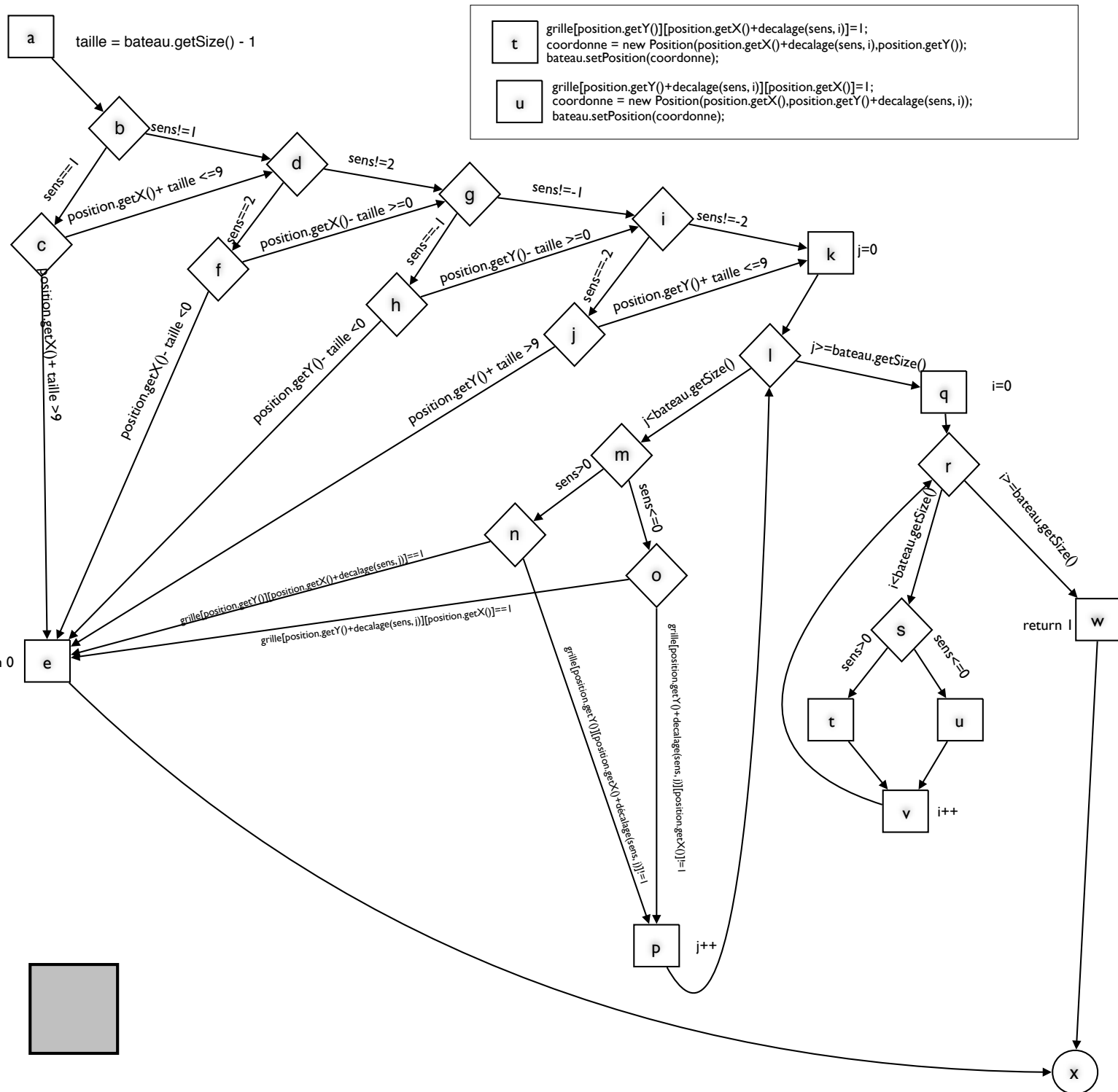
    for (int i=0; i<bateau.getSize(); i++) {
        if (sens>0) {
            grille[position.getY()][position.getX()+decalage(sens, i)]=1;
            Position coordonnee;
            coordonnee = new Position(position.getX()+decalage(sens, i), position.getY());
            bateau.setPosition(coordonnee);
        } else {
            grille[position.getY()+decalage(sens, i)][position.getX()]=1;
            Position coordonnee;
            coordonnee = new Position(position.getX(), position.getY()+decalage(sens, i));
            bateau.setPosition(coordonnee);
        }
    }
    return 1;
}
```

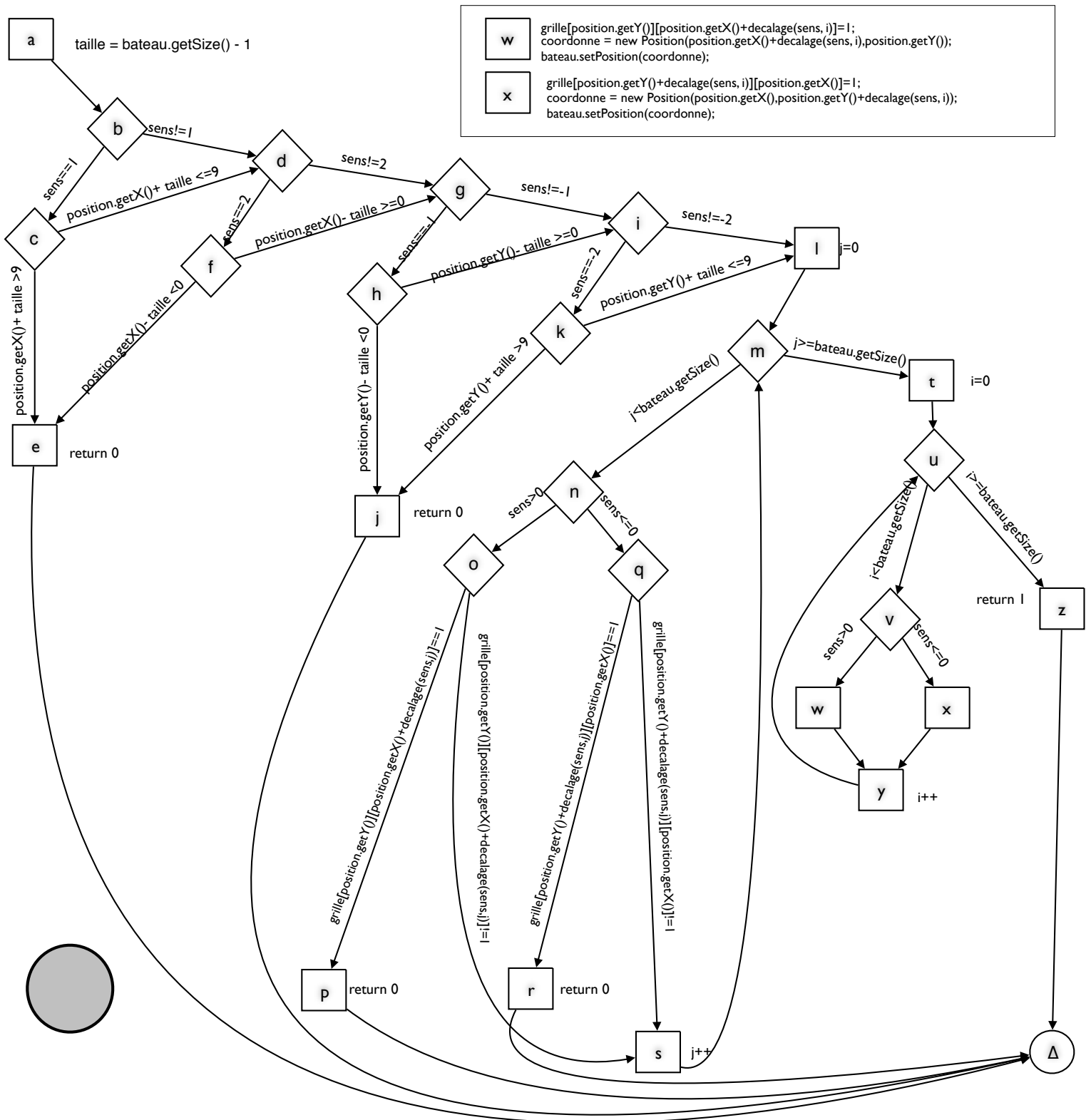
Reportez ici votre réponse à la question 1 (page 6) :

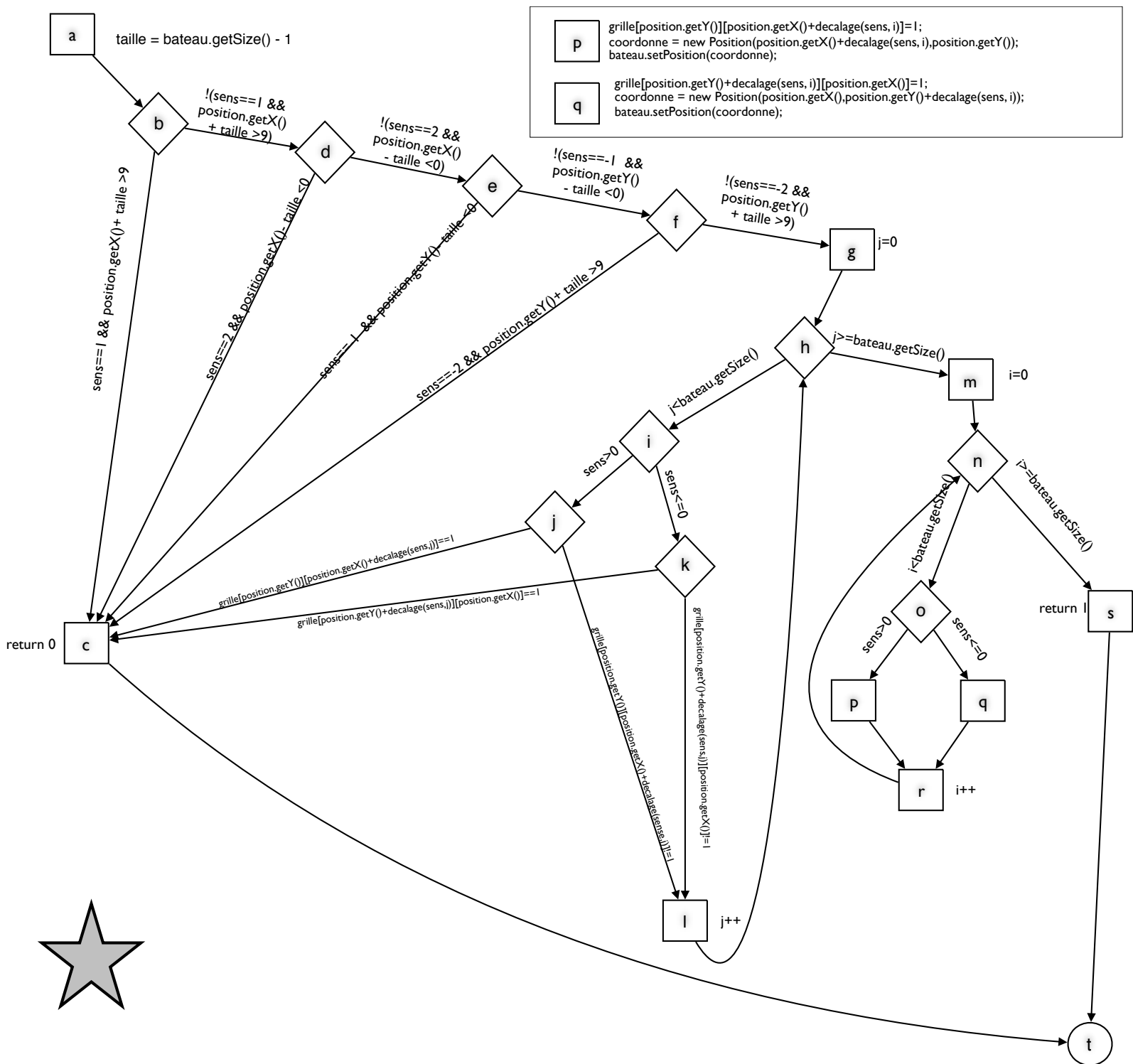


(encerclez le motif)









---

Chaque graphe est repéré par une figure géométrique utilisée comme identifiant du graphe.

**Question 1 :** Parmi les quatre graphes proposés, lequel est conforme à la fonction `placerObjet` et aux consignes de construction des graphes de flot de contrôle vues en TD

- ☐ Triangle
- ☐ Carré
- ☐ Rond
- ☐ Etoile

**Question 2 :** Sur le graphe que vous avez choisi à la question précédente, établissez l'expression des chemins du graphe (pour rappel : expression régulière permettant de représenter tous les chemins du graphe)

**Question 3 :** Calculez le nombre de chemins présents dans le graphe que vous avez choisi

## Exercice 2 : Couverture

**Question 4 :** Calculez le nombre de cas de tests nécessaires pour assurer un niveau de couverture de tous chemins indépendants

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27

Entourez le nombre choisi

**Question 5 :** Sur le graphe que vous avez choisi à la question 1, vous souhaitez obtenir le niveau de couverture de tous les arcs. Un certain nombre de données de test vous sont fournies, mais peut être que certaines sont inutiles et que d'autres sont manquantes.

Une donnée de test est constituée de l'ensemble des valeurs passées en paramètre de la fonction, c'est à dire :

- une **position** donnée sous la forme d'un doublet  $\langle x, y \rangle$  représentant les coordonnées de la position (les valeurs  $x$  et  $y$  sont récupérées respectivement par les méthodes `getX()` et `getY()`)
- un **bateau**, qui sera représenté par `\_3\_` pour un bateau de longueur 3. La taille du bateau est récupérée par `getSize()`
- un **sens** dont la valeur est un entier
  - 1 si le bateau est orienté vers l'Est ;
  - 2 s'il est orienté vers l'Ouest ;
  - -1 s'il est orienté vers le Nord ;
  - -2 s'il est orienté vers le Sud.
- l'état de la **grille**, qui sera représenté par la liste des bateaux. Les bateaux présents seront mentionnés sous la forme d'un triplet ( position, bateau, sens ) tels que décrit ci-dessus.
  - Ainsi, `[]` désignera une grille vide et
  - `[ (< 1,2 >, \_2\_/, 1), (< 5,6 >, \_4\_/, -1) ]` une grille dans laquelle ont été placé 2 bateaux, le premier de longueur 2 en coordonnées 1,2 dans l'orientation 1 et le deuxième de longueur 4 en coordonnées 5,6 et dans l'orientation -1.

Les données de test fournies sont :

DT1  $\langle 2, 2 \rangle$ , `\_3\_`/, -1, []  
DT2  $\langle 2, 3 \rangle$ , `\_3\_`/, 1, []  
DT3  $\langle 4, 5 \rangle$ , `\_4\_`/, 2, []  
DT4  $\langle 6, 3 \rangle$ , `\_5\_`/, 1, []  
DT5  $\langle 7, 7 \rangle$ , `\_4\_`/, -2, []  
DT6  $\langle 5, 2 \rangle$ , `\_4\_`/, -1, []  
DT7  $\langle 2, 3 \rangle$ , `\_4\_`/, 2, []  
DT8  $\langle 3, 6 \rangle$ , `\_4\_`/, -1, [ $\langle 3, 3 \rangle$ , `\_2\_`/, 1]  
DT9  $\langle 7, 3 \rangle$ , `\_4\_`/, 2, [ $\langle 3, 3 \rangle$ , `\_2\_`/, 1]

Pour chacune de ces données de test, établissez le chemin couvert par la donnée de test ( $C_{DT}$ ) et l'oracle du test.

- $C_{DT1}$  :
- $C_{DT2}$  :
- $C_{DT3}$  :
- $C_{DT4}$  :
- $C_{DT5}$  :

- $C_{DT6}$  :
- $C_{DT7}$  :
- $C_{DT8}$  :
- $C_{DT9}$  :

Ces données de tests permettent t'elles d'atteindre (rayez la mention inutile ou entourez la mention retenue) :

1. le niveau de couverture tous les noeuds : oui - non
2. le niveau de couverture tous les arcs : oui - non
3. le niveau de couverture tous chemins indépendants : oui - non

**Question 6 :** Si les données de tests ne permettent pas de couvrir tous les noeuds, proposez des cas de test complémentaires permettant d'atteindre le niveau de couverture tous noeuds.

**Question 7 :** Si les données de tests ne permettent pas de couvrir tous les arcs, proposez des cas de test complémentaires permettant d'atteindre le niveau de couverture tous arcs.

**Question 8 :** Est-il pertinent de répondre séparément aux 2 questions précédentes ?

- ☐ vrai  
☐ faux