

Algorithme 3 : Exécution d'un cycle de prédateur

```
Entrées : grille, ligne, colonne, animal
1 début
2   Incrémenter l'âge de l'animal par NB_JRS_PUBERTE_PRED;
3   si énergie de l'animal < MIN_SANTE_PRED ou âge de l'animal > MAX_AGE_PRED alors
4       Mettre la case (ligne, colonne) de la grille à vide;
5       Décrémenter le compteur de prédateurs de la grille;
6   retourner
7 fin
8 Chercher un voisin autour de la case (ligne, colonne) qui contient une proie;
9 si il existe un voisin avec une proie alors
10     Mettre la case du voisin à l'état de prédateur;
11     Mettre la case (ligne, colonne) de la grille à vide;
12     Rendre l'animal non disponible pour déplacement;
13     Ajouter AJOUT_ENERGIE à l'énergie de l'animal;
14     Mettre à jour l'animal dans la case du voisin;
15     Décrémenter le compteur de proies de la grille;
16     if âge de l'animal  $\geq$  NB_JRS_PUBERTE_PRED et jours de gestation de l'animal  $\geq$   

       NB_JRS_GESTATION_PRED then
17         Chercher un voisin autour du nouveau emplacement de l'animal qui est vide;
18         if il existe un voisin vide then
19             Mettre le voisin vide à l'état de prédateur;
20             Initialiser le bébé prédateur avec des valeurs par défaut et le placer dans la case vide;
21             Incrémenter le compteur de prédateurs de la grille;
22         end
23         Réinitialiser les jours de gestation de l'animal à 0;
24         Mettre à jour l'animal dans la grille;
25     end
26 sinon
27     Décrémenter l'énergie de l'animal de 1;
28     Déplacer l'animal vers une case voisine disponible ou le laisser sur place si aucun  

       déplacement n'est possible;
29 fin
30 fin
```