

BPO

TP 4 - Le pingouin et les poissons

Un pingouin sur l'océan

1. Complétez l'application avec la classe **Pingouin**. Un pingouin est défini par sa position et sa taille (un pingouin est carré), il peut être déplacé sur les abscisses ou sur les ordonnées. L'océan contient un pingouin, placé en bas à droite. À noter que la classe **ArcticlImage** gère deux nouvelles couleurs (codées 2 et 3) pour colorier le pingouin.
2. Dessinez le diagramme de classe simplifié (sans fonction) de votre application.
3. Écrivez la classe **Jeu** qui permet de jouer, avec un constructeur sans paramètre et une fonction **jouer** sans paramètre. L'exécution de cette fonction affiche sur la sortie standard un menu proposant à l'utilisateur les options **z q s d** ou **Z Q S D**. Ces options permettent de déplacer le pingouin dans l'océan dans l'une des quatre directions. Tout autre choix d'option permet d'arrêter de jouer.

À chaque déplacement, l'affichage doit être rafraîchi par un appel à la fonction **setColors** sur une **ArcticlImage**.

4. Dans la fonction **jouer**, ajoutez des contraintes sur le déplacement du pingouin : il ne peut pas sortir des limites de l'océan.

Le pingouin mange des poissons

5. Sur le modèle de la classe **Pingouin**, complétez l'application avec la classe **Poisson**. Un poisson est défini par sa position, sa hauteur et sa largeur, et son type de déplacement. Il ne peut se déplacer que sur les abscisses ou bien que sur les ordonnées. Dès qu'il touche un bord de l'océan, il réapparaît de l'autre côté. L'océan contient plusieurs poissons. La classe **ArcticlImage** gère deux couleurs de poissons (codées 4 et 5) pour colorier les poissons. Reprendre le nouveau code de **ArcticlImage** sur arche.
6. Compléter le jeu de sorte qu'à chaque tour de jeu, les poissons se déplacent conformément à leur sens de déplacement.
7. Le pingouin mange les poissons qu'il atteint. Un poisson mangé disparaît ...

Le jeu se complique

8. Continuez la programmation d'un jeu avec de nouvelles règles :
 - un pingouin qui monte sur un iceberg le casse,
 - les icebergs fondent régulièrement,
 - au bout d'un certain nombre de déplacements sans manger, le pingouin fatigue ... et change de couleur,
 - au bout d'un certain nombre d'allers et retours, un poisson meurt de fatigue.
 - *etc.*
9. À vous d'imaginer la suite, en ajoutant des règles ou/et d'autres éléments de jeu. Il reste d'autres couleurs non utilisées dans **ArcticImage** (de 6 à 7).

Vous avez quelques jours pour terminer le développement de ce jeu et déposer sur Arche les fichiers sources du package **glaces**, incluant le diagramme de classes (un diagramme dessiné à la main et pris en photo convient).

Pour ceux qui ont réalisé une jolie application, pourquoi ne pas faire une capture vidéo d'une partie ? Vous pouvez ensuite déposer cette vidéo sur Arche et elle sera mise en ligne sur le site web de la licence informatique. Cette vidéo ne vous rapporte aucun point... juste le plaisir de finaliser et de montrer votre travail.