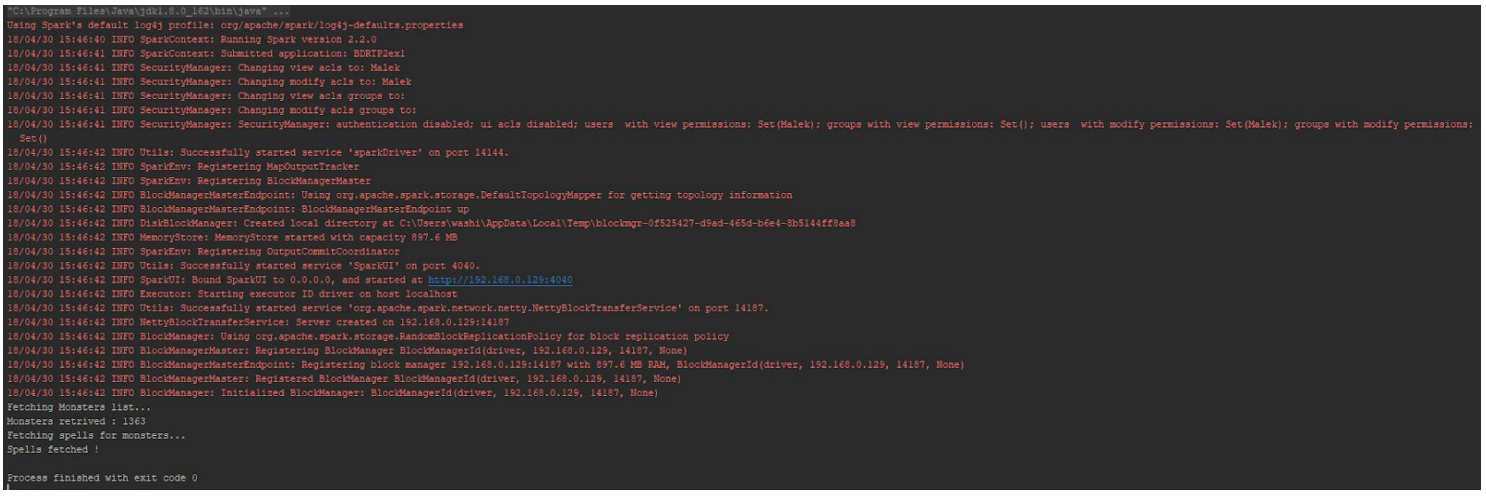
Rapport TP2

- Paul Seichais
- Malek El Ouerghi
- Pierre Lochouarn

Exercice 1:



Exécution du crawler pour retrouver les monstre par sort.
Pour cela nous avons réaliser un programme en scala ou nous envoyons des requêtes sur un site et parsons ensuite le résultat.

Resultat : https://gist.github.com/washiqeri/7fdd1a85b66f0cb7eef0d2b2511d6675

Exercice 2 :

***Question ouverte** : Comment gérer efficacement un système de distance 3D avec un graphe d'agents distribué? Avoir un système de distance 3D permet d'avoir accès aux attaques de zone, le vol, les arcs et flèches etc.*
On met la distance sur le sommet? sur l'arête? problèmes de collision etc.

Nous avons choisi de stocker les positions (x,y) de chaque sommet du graphe. La distance en elle-même n'est stockée nulle part, elle est calculée (distance euclidienne) sur le moment si on en a besoin. Ceci nous permet de modifier facilement les positions de chacun sans toucher à la structure du graphe et sans toucher aux arêtes. Pour gérer facilement les collisions, on peut donner une hitbox sphérique a chaque sommet et ainsi pour vérifier la collision entre deux sommet, il suffit de regarder si la distance entre les deux sommets est inférieure à la somme des rayons des deux sphères de collisions.

Combat 1 :

Nous avons donc pour ce combat créé un graphique ou tous les monstres sont relié au solar.
Nous avons donc le Solar, 10 Worgriders, 4 Barbarian Orcs et 1 warlords

Après déroulement du combat voici le résultat final :

```
----------------------------------FIN--------------------
ETAT APRES LE COMBAT
SURVIVANTS
Solar 1 HP: 363
---------------------------------- FIN COMBAT 1--------------------
```

Pour arriver à ce résultat voici notre workflow :

- Chaque noeud représente un monstre et chaque arête représente un type de relation entre les différents noeuds (dans ce combat il n'y a que des arête de type ENEMY)
- Chaque monstre dispose d'une méthode action ainsi que d'une position représenté selon des axes X, Y
- Egalement il dispose chacun d'une méthode damage qui renvoie des dégâts en fonctions de la classe du monstre
- Nous lancer notre méthode d'exécution qui :
    - Créer une liste d'action possible pour chaque monstre en fonction des différentes arêtes qui lui sont lié ainsi que la distances entre chaque monstre.
    - Puis un joins vertice qui choisit l'action que le noeuds effectuera et sur quel cible cela sera appliquée
    - Ensuite nous relancer un aggregate pour envoyer les messages d'attaque au noeuds ciblé ou, dans le cas d'un déplacement ceux ci sont renvoyé sur le noeuds actuel
    - Enfin le dernier joins vertice applique les actions qui on été envoyé précédemment

```
----------------------------------COMBAT 1--------------------
----------------------------------DEBUT DU TOUR--------------------
Tour 1
Warlord16 avance vers Solar1
Worgrider4 avance vers Solar1
BarbarianOrc12 avance vers Solar1
Worgrider11 avance vers Solar1
Worgrider3 se fait attaquer a RANGED de Solar pour 21 points de degats
Worgrider3 avance vers Solar1
----------------------------------MONSTER--------------------
Worgrider 3 est mort
```

Pour choisir les actions tous les monstre choisissent les attaques en priorité, par exemple si le solar à le choix entre un mouvement ou une attaque à distance ils choisira d'attaquer à distance, cependant en cas de possibilité d'attaquer au corps à corps, ils préfèreront l'attaque corps à corps.

Combat 2 :

Le déroulement du combat 2 est exactement le même que le combat 1, sachant que le dragon est pris en compte, celui-ci adopte une stratégie assez basique, il tente de tuer le solar dès le début du combat en traversant le champs de bataille grâce à l'alterSelf, et si cela échoue il s'envole et bombarde les anges de façon aléatoire.

Pour ce programme nous avons nos ordinateurs en cluster afin d'exécuter plus rapidement.

Voici le résultat final :

```
----------------------------------MONSTER--------------------
----------------------------------FIN--------------------
ETAT APRES LE COMBAT
SURVIVANTS
Astral 8 HP: 172
Solar 1 HP: 363
Astral 9 HP: 172
Astral 10 HP: 172
Planetar 2 HP: 229
Planetar 3 HP: 229
MovanicDeva 4 HP: 117
MovanicDeva 5 HP: 126
Astral 6 HP: 172
Astral 7 HP: 172
--------------------------------- FIN COMBAT 2--------------------
```

Exécution en cluster (2 PCs) : https://youtu.be/XLjVecyszXQ
En cluster, on n'a pas les print qui détaillent l'avancement du combat (attaque, bouge, soigne, tire à distance, etc.) vu que les print se font sur chaque worker.