Description of my implements of collisions

First, I noticed the size of each mobs are multiplied 2 in the function of drawMobs so I remove the \*2 in the function of drawMobs and then directly change the size of each type of mobs in header file. This change is prone for me to control the colliding and drawing consistently.

The approaches of implementing colliders mainly include:

For river colliders, I create 20 Mobs\_River in GameState to implement the river colliders. Because river is static, I create them in struct of GameState and create them only once after lunching the game.

For buildings colliders, I use the formula: (abs(this->getPosition()->x - otherBuilding->getPosition()->x) < averageOfSzie && abs(this->getPosition()->y - otherBuilding->getPosition()->y) < averageOfSzie ) to check the collision between them.

For mobs colliders, I use the formula: abs(this->getPosition()->x - otherMob->getPosition()->x) < averageOfSzie && abs(this->getPosition()->y - otherMob->getPosition()->y) < averageOfSzie） to check the collision between them.

The masses of river and buildings are both infinite, so I only need to push the “this”. std::shared\_ptr<Point> pos = std::shared\_ptr<Point>(new Point(this->getPosition()->x + (this->getPosition()->x - otherBuilding->getPosition()->x), this->getPosition()->y + (this->getPosition()->y - otherBuilding->getPosition()->y)));

this->moveTowards(pos, elapsedTime);

The collisions between each mob separate to three situations.

if (otherMob->GetMass() > this->GetMass()) {

…

}

else if (otherMob->GetMass() == this->GetMass()) {

if (this->isMoveSamedir(otherMob)) {

if (this->isBehind(otherMob)) {

this->moveTowards(pos1, elapsedTime);

}

else {

otherMob->moveTowards(pos, elapsedTime);

}

}

else {

this->moveTowards(pos1, elapsedTime);

otherMob->moveTowards(pos, elapsedTime);

}

}

I think the colliders of the mobs which have same mass still need to be improved.

else if (otherMob->GetMass() < this->GetMass()){

…

}

In the process of collisions, sometime the mobs may be pushed then do some unreasonable things. Although this phenomenon is not very obvious, there should be some improvements to avoid bugs. In present, I do not consider the flying units, if some flying units be added into the game, there will be a more complex situation.

I change the search enemy to make it more reasonable.

If there are enemies closer than current target and current target is out of attack range, “this” should first attack closer enemy. The mobs cannot find the enemies on the other side of river. The mobs first attack mobs, then attack buildings.

Many aspects of the finding enemies should be improved, for example, the left mobs only find the enemies on left route as well as the right mobs only find the enemies on right route.

I let the class of building to inherit the class of attackable to receive mobs’ damages.