**Report of Coursework**

**Assumptions for my game and narrative:**

Each creature has hit point (5 as default) and attack point (random between 0 and 20) which is used for food fight. When 2 creatures are generated to the same location one of them becomes super creature and other one dies. Their hit point is set to 9 as default and their attack point is set to 25 which is certainly higher than normal creatures’ attack point. Therefore, if a super creature fights with normal creature, super creature always survives. I didn’t write a specific algorithm for fight of 2 super creatures but if 2 super creatures fight with current algorithm, the one which was generated firstly wins, other one dies. Also super creatures are showed as yellow on the board.

Each food has hit point (2 as default). There are also super foods which are seem as green on board and generated when 2 normal foods are generated to the same location. One of those foods become super food other one is sent to out of board. I sent it to out of board so any create won’t be able to eat it. I didn’t delete it from food vector because deleting an element of a vector is kind of problematic. If we delete an element of a vector in such loop like (for i=0 to vector.size()) it causes exception. An algorithm which doesn’t cause exception for deleting and an element from vector is kind of cpu insufficent. I have done this for deleting creatures because same creatures survive during whole game, but every round foods vector is anyway cleaned so, holding just a few addional super foods in memory isn’t big deal.

Each round, a nearest food (if there is any in max eat distance) is set to a creature’s ‘neareastFood’ object. Basically, every round each creature tries to eat only one food, if there isn’t any food in max eat distance then they don’t eat anything. Creatures fight if their distances to the same food are equal to each other. Only one creature survives after fight even if fight is among multiple creatures. If one food is set as nearest food to different creatures and if their distances are different from each other, then nearest one eats that food and other one doesn’t eat anything even if it has another near food (in max eat distance but not the nearest). Because every creatures tries to eat only nearest food is my assumption.

**Explanation of code briefly:**

This is my main function.

int main()

{

int newGameOrExit;

do

{

Console console(70, 70);

//Creating an object from Console class

console.setColour(console.COLOUR::BLACK, console.COLOUR::LIGHT\_AQUA);

//It sets color of console to black and aqua

Input read;

//Creating an object from input class

read.readAll(console);

//It reads all variable from user such as numbers of foods and creatures

Board board(read.getNumberOfCreature(),read.getNumberOfFood(),console);

//Here we generate an object from board class

board.newBoard(read.getNumberOfFood(), console, read);

//Almost everything about game like generating foods, creatures, fighting, eating happens here. I will extend function below

board.localStats.printAllStats(console);

//This function prints statistics of game

read.newGame(console);

//This function asks to user either he wants to replay or exit

newGameOrExit=read.getNewGameOrExit();

//This function gets user’s decision

}while (newGameOrExit == 1); //Game restarts if user's inoput is 1

return 0;

}

First let’s have a look at variables in board class:

vector<Creature> creatures; //holds all creature objects

vector<Creature> killList; //holds the creatres that must be killled

vector<Food> foods; //holds all food objects

RandomNumber random; //object from RandomNumber class

Explanation of “newBoard()” function:

void Board::newBoard(int numberOfFoods, Console console,Input read)

{

generateCreatures(read.getNumberOfCreature());  
//Generates creatures, pushes them to the vector

setSuperCreatureif2CreaturesGeneratedtoSameLocation();

//If 2 creatures are generated to same location, one disappears other one becomes superCreature

killCreaturesInList(); //Deletes creatures from 'creatures' vector by comparing ides of creatures in killList vector

while (creatures.size() > 0) //Game runs as long as size of creatures vector is greater than zero

{

setAllNearestFoodIdAndDistanceToDefault(); //For correct eating and fighting, 'nearestFood' object that every creatures has, has default id (-1) and default distance (100)

printCreatures(console); //Prints creatures to screen that they were already generated and held in creatures vector

generateFoods(numberOfFoods, console); //Generates foods and super foods, prints them to screen and checks if a food is dropped on a creature, if yes inserts creature to killList

killCreaturesInList(); //Deletes creatures from 'creatures' vector where they are also inserted into killList by previous functions

findNearestFood(read); //Sets the neareast food to all creatures (if there is any in max eat distance)

Fight(read); //If distance of multiple creatures to the same food is equal to each other, they all fight, only one survives and eats

killCreaturesInList(); //Deletes creatures from 'creatures' vector where they are also inserted into killList by previous function

setIdToDefaultIfNotClosest(); //If same food is set to multiple creatures as nearest food, far away ones' nearestFoodId will be set to default (-1), so only closest one will be able to eat that food (the ones that their distances to same food were equal to each other already fought )

eatNearestfood(); //Finally creautres that have nearest food, eat their food and gain hitpoint

decreaseAllHitpoints(read); //Decrease hitpoint of all creatures every round by the amount the user chooses

killCreaturesInList(); //Deletes creatures from 'creatures' vector where they are also inserted into killList by previous functions

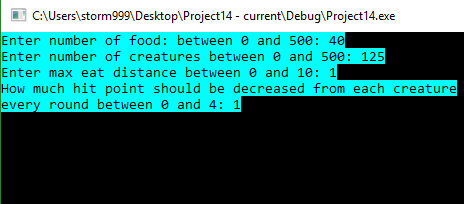
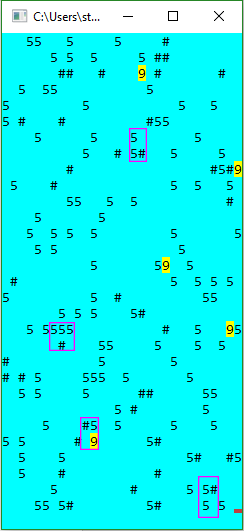
localStats.incRoundNumber(); //Increases round number of statistics by 1

console.clear(); //Cleans console for next round

}

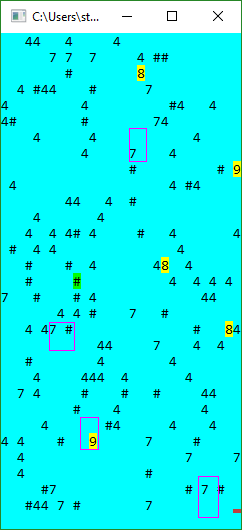
}

**Screenshots**



In first screen we enter inputs.

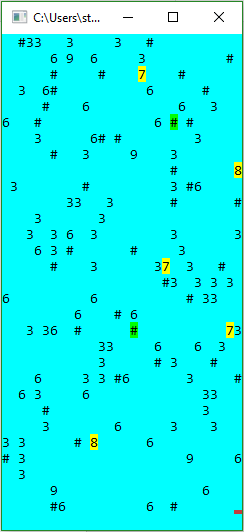
Screenshot in right side is first round of the game. Yellow ones represent super creatures. There is no super food in this round. The creatures that I have taken in purple box are going to fight. Only one creature in each box is going to survive. In one of boxes there is a super creature, it is absolutely going kill other.



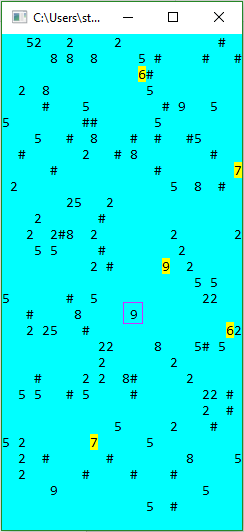
Here we can see that, the creatures in purple boxes fought and only one of them survived.

And hit points of the normal creatures that have eaten food increased to 7 and the super creatures stayed at 9. Because max possible hit point is 9 as it is written in coursework sheet.

In this round we also have one super food. But any creature isn’t going to eat it because in the beginning I have entered 1 for max eat distance.



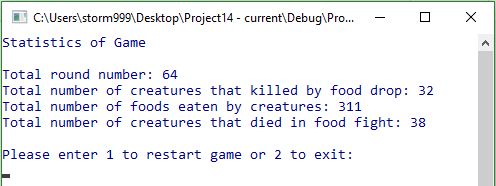
Here we have 2 super foods close to creatures and also one new fight case as well. I want to show that the creatures’ hit point that will eat super food will be 9. Because hit point of super food is 5 while hit point of normal food is 2.



Here we can see that hit point of creatures that ate super food became 9.

Also in bottom of board, another food fight happened.

So as you saw in the screenshots my game is working very well. As I said in assumptions:

* Creatures fight if their distance to same food is equal to each other.
* Super creatures always win fighting over normal creatures.
* Super food increases hit point of creature by 5 while normal foods increase by 2.
* Hit point of every creature is decreased by one every round.

Lastly in this screenshot, you can see statistics of game.

**References**

Everything about C++  
http://ninova.itu.edu.tr/tr/dersler/bilgisayar-bilisim-fakultesi/21/blg-252e/ekkaynaklar?g397

How to hold objects in vector  
https://www.youtube.com/watch?v=iPlW5tSUOUM

General properties of vectors and its functions  
http://www.cplusplus.com/reference/vector/vector/

General information about vectors  
https://msdn.microsoft.com/en-us/library/9xd04bzs.aspx

How to delete element/s from vector  
http://stackoverflow.com/questions/875103/how-to-erase-element-from-stdvector-by-index

Multidimensional vectors  
http://www.cplusplus.com/forum/general/833/