Project 1

<Gluttonous Snake>

<Version>

Name: Xiaojun He

Class: CSC-17A 48096

Date: 10/28/2015

Contents

1. Introduction…………………………………………………....3
2. Summary…………………………………………………………4
3. Problems during coding…………………………………..4
4. System Libraries………………………………………………5
5. Variable List…………………………………………………….5
6. Function List……………………………………………………6
7. Flowchart………………………………………………………..7
8. Code………………………………………………………………11
9. Introduction(From wiki):

The Snake design dates back to the [arcade game](https://en.wikipedia.org/wiki/Arcade_game) [Blockade](https://en.wikipedia.org/wiki/Blockade_%28video_game%29),[[3]](https://en.wikipedia.org/wiki/Snake_%28video_game%29#cite_note-Goggin-3)[[4]](https://en.wikipedia.org/wiki/Snake_%28video_game%29#cite_note-DeMaria-4) developed and published by [Gremlin](https://en.wikipedia.org/wiki/Gremlin_Industries) in [1976](https://en.wikipedia.org/wiki/1976_in_video_gaming).[[5]](https://en.wikipedia.org/wiki/Snake_%28video_game%29#cite_note-5) It was cloned as [Bigfoot Bonkers](https://en.wikipedia.org/wiki/Bigfoot_Bonkers) the same year. In 1977, [Atari](https://en.wikipedia.org/wiki/Atari,_Inc.) released two Blockade-inspired titles: the arcade game [Dominos](https://en.wikipedia.org/wiki/Dominos_%28video_game%29) and Atari 2600 game [Surround](https://en.wikipedia.org/wiki/Surround_%28video_game%29).[[6]](https://en.wikipedia.org/wiki/Snake_%28video_game%29#cite_note-KLOV-6) Surround was one of the nine [Atari 2600 (VCS) launch titles](https://en.wikipedia.org/wiki/List_of_video_game_console_launch_games#Atari_2600) in the [United States](https://en.wikipedia.org/wiki/United_States) and was also sold by [Sears](https://en.wikipedia.org/wiki/Sears) under the name Chase. That same year, a similar game was launched for the [Bally Astrocade](https://en.wikipedia.org/wiki/Bally_Astrocade) as Checkmate.[[7]](https://en.wikipedia.org/wiki/Snake_%28video_game%29#cite_note-7)

The first known [personal computer](https://en.wikipedia.org/wiki/Personal_computer) version, titled Worm, was programmed in 1978 by Peter Trefonas of the US on the [TRS-80](https://en.wikipedia.org/wiki/TRS-80),[[3]](https://en.wikipedia.org/wiki/Snake_%28video_game%29#cite_note-Goggin-3) and published by CLOAD magazine in the same year. This was followed shortly afterwards with versions from the same author for the [Commodore PET](https://en.wikipedia.org/wiki/Commodore_PET) and [Apple II](https://en.wikipedia.org/wiki/Apple_II). A microcomputer clone of the Hustle arcade game, itself a clone of Blockade, was written by Peter Trefonas in 1979 and published by CLOAD.[[8]](https://en.wikipedia.org/wiki/Snake_%28video_game%29#cite_note-autogenerated1-8) An authorized version of Hustle was published by [Milton Bradley](https://en.wikipedia.org/wiki/Milton_Bradley_Company) for the [TI-99/4A](https://en.wikipedia.org/wiki/Texas_Instruments_TI-99/4A) in 1980.[[9]](https://en.wikipedia.org/wiki/Snake_%28video_game%29#cite_note-9) In 1982's Snake for the BBC Micro, by Dave Bresnen, the snake is controlled using the left and right arrow keys relative to the direction it is heading in. The snake increases in speed as it gets longer, and there's only one life; one mistake means starting from the beginning.

[Nibbler](https://en.wikipedia.org/wiki/Nibbler_%28video_game%29) (1982) is a single-player arcade game where the snake fits tightly into a maze, and the gameplay is faster than most snake designs. Another single-player version is part of the 1982 [Tron](https://en.wikipedia.org/wiki/Tron_%28video_game%29) arcade game, themed with light cycles. It created new interest in the snake concept, and many subsequent games borrowed the light cycle theme.

Starting in 1991, [Nibbles](https://en.wikipedia.org/wiki/Nibbles_%28video_game%29) was included with [MS-DOS](https://en.wikipedia.org/wiki/MS-DOS) for a period of time as a [QBasic](https://en.wikipedia.org/wiki/QBasic) sample program. In 1992 [Rattler Race](https://en.wikipedia.org/wiki/Rattler_Race) was released as part of the second [Microsoft Entertainment Pack](https://en.wikipedia.org/wiki/Microsoft_Entertainment_Pack). It adds enemy snakes to the familiar apple-eating gameplay.

[Slither.io](https://en.wikipedia.org/wiki/Slither.io) (2016) is a massively multiplayer version of Snake.

1. Summary

|  |  |
| --- | --- |
| Total Line of Code | 252 |
| Variable | 15 |
| Function | 4 |

This game contains most concepts that we have learn in the class. I use the pointer to get the x and y coordinate. I use the getch() to get the information from the key board. This function do not need me to pass enter to work. I use a lot loop to build the game work.

1. Problem during coding

The most important problem is I cannot use just the switch statement to get the player pass. Because in this game, it do not want the player to pass enter key to control the snake. So I google it, then the friend from the web tall me I should use the conio.h library. But this library is not work in NetBeans. Then I go into NetBeans website to update my NetBeans frist.

1. System Libraries

<isotream>

<ctime>

<cstdlib>

<stdlib.h>

<conio.h>

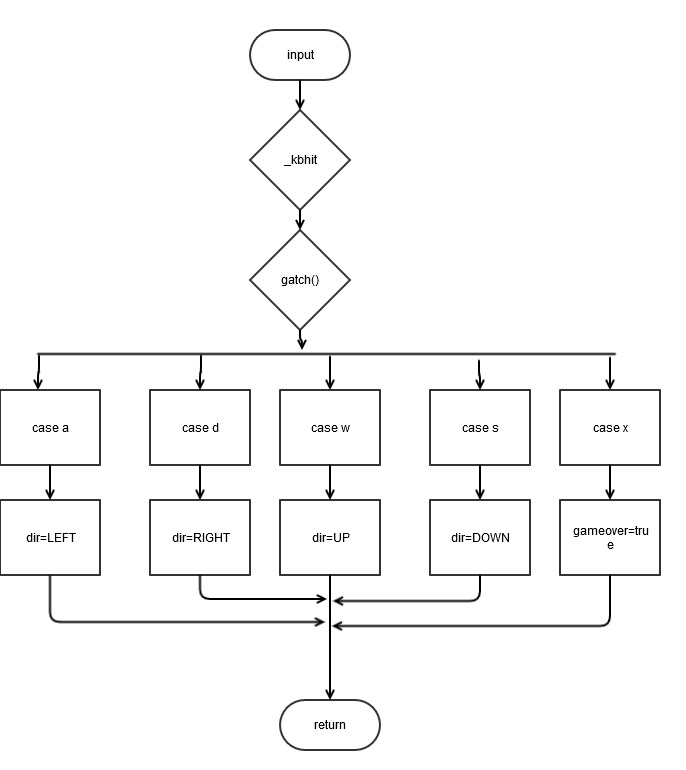
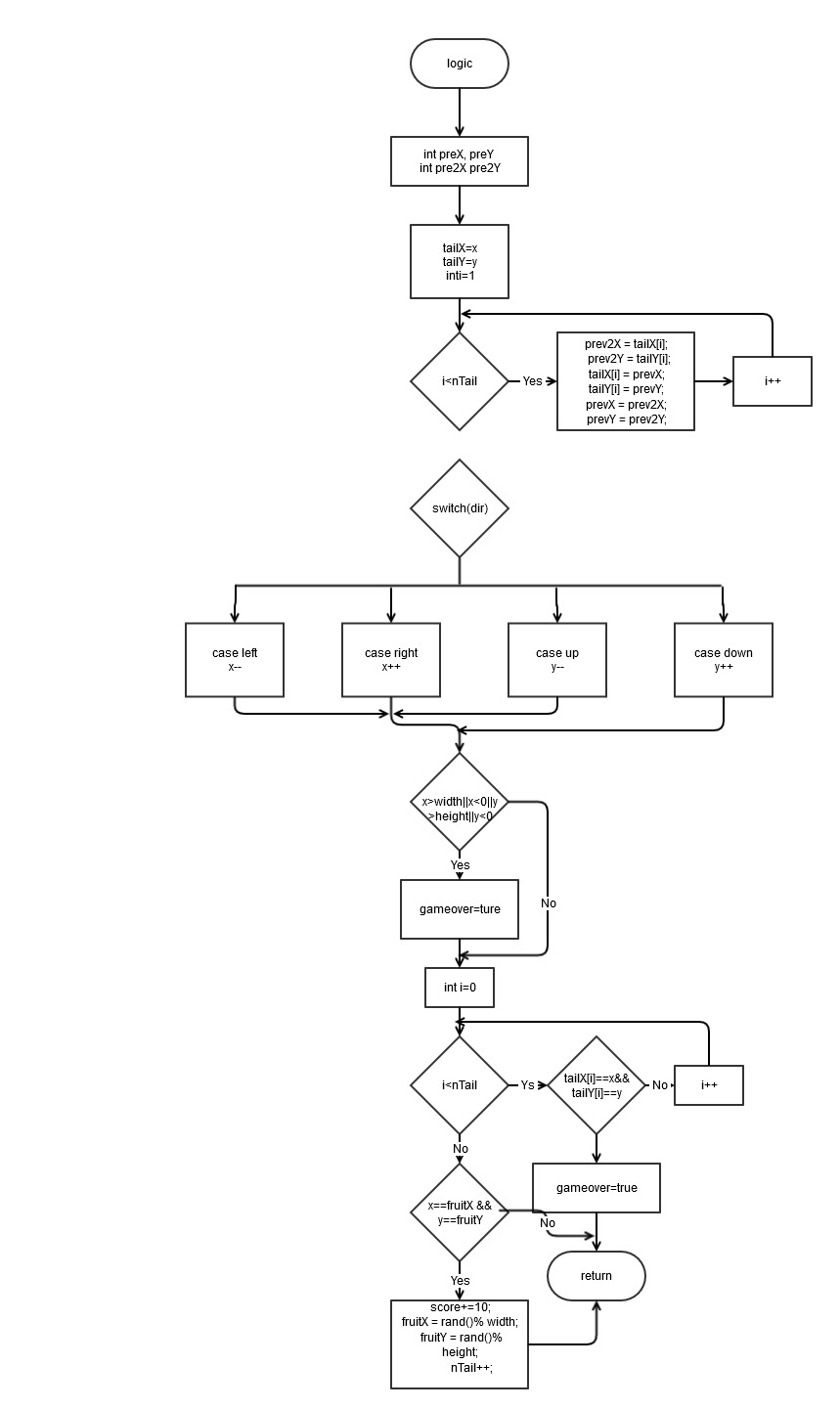
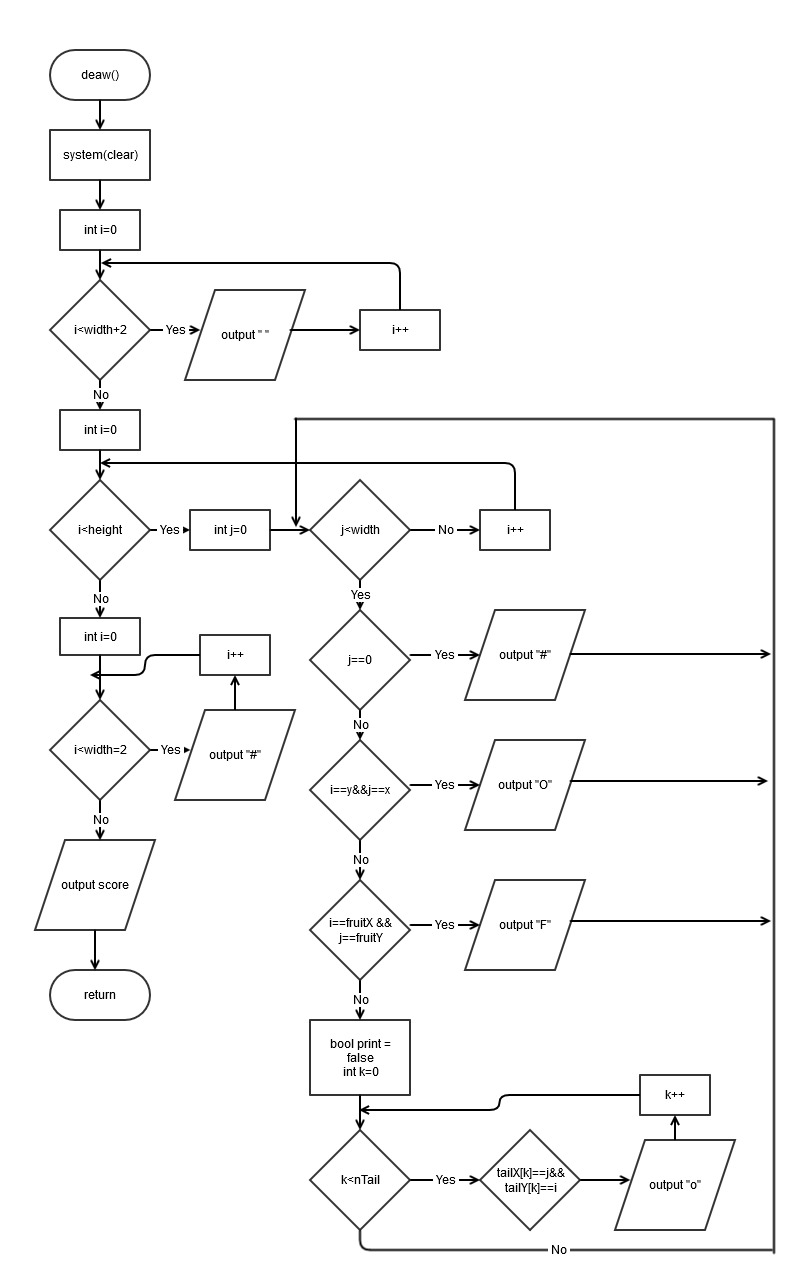
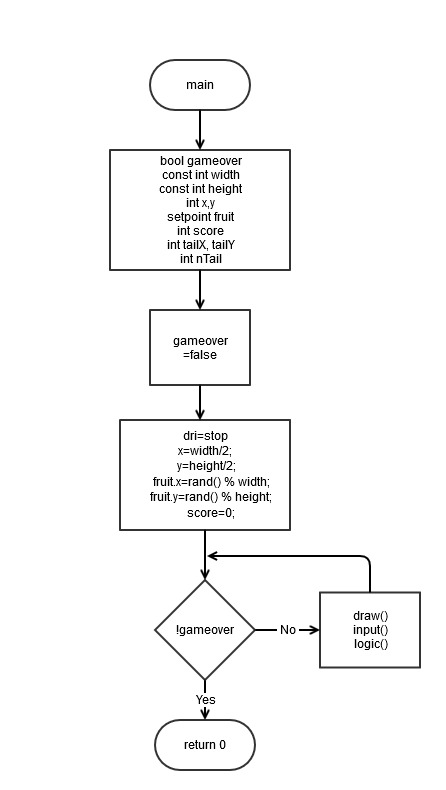
1. Variables List

|  |  |  |  |
| --- | --- | --- | --- |
| Type | Name | Description | Line |
| int | X | Point x | 20 |
|  | y | Point y | 20 |
| bool | gameover | Game over | 46 |
| Const int | width | width | 47 |
|  | height | high | 48 |
| int | x | number | 49 |
|  | y | number | 49 |
|  | Fruit x | Point fruit | 50 |
|  | Fruit y | Point fruit | 50 |
|  | score | Score | 51 |
|  | tailX | Point tail | 52 |
|  | tailY | Point tail | 52 |
|  | nTail | Tail long | 53 |

1. Function List

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| type | name | argument | function | Location |
| Void | Draw | int width,int height,int x,int y, int fruitX,int fruitY,int nTail, int tailX,int tailY,int score | Draw the map, snake and fruit | 79 |
| Void | Input | bool gameover | input thing in the game | 140 |
| void | logic | int width,int height,int x,int y,  int fruitX,int fruitY,int nTail, int tailX,int tailY,int score,bool gameover |  |  |

1. Frowchart



1. Code

/\*

\* File: main.cpp

\* Author: Jimmy

\* Created on October 24, 2016, 4:13 PM

\* project 1

\*/

//library

#include <iostream>

#include <ctime>

#include <cstdlib>

#include <stdlib.h>

#include <conio.h>

using namespace std;

//set the point x and y

struct setpoint{

int x, y;

};

//set the enum to set the direction

enum eDirecton{STOP=0,LEFT,RIGHT,UP,DOWN};

eDirecton dir;

//the function to draw

//include the map , snake and the fruit

void draw(int width,int height,int x,int y,

int fruitX,int fruitY,int nTail,

int tailX,int tailY,int score);

//input thing in the game

//it use w a s d to control the snake

//w is up, s is down, a is left, d is right

void input(bool);

//the function for show the snake hand and body

//it also calculate the score

void logic(int width,int height,int x,int y,

int fruitX,int fruitY,int nTail,

int tailX,int tailY,int score,bool gameover);

int main(int argc, char\*\* argv) {

bool gameover; //check game over

const int width=20; //set the width is 20

const int height=20; //set the height is 20

int x, y;

setpoint fruit; //the fruit point

int score; //score in the game

int tailX[100],tailY[100]; //the tail for the snake

int nTail; //number of tail

gameover=false;

dir = STOP;

x=width/2;

y=height/2;

//random set the fruit

fruit.x=rand() % width;

fruit.y=rand() % height;

score=0;

//play the game

while(!gameover){

draw(width,height,x,y,fruit.x,

fruit.y,nTail,tailX,tailY,score);

input();

logic(width,height,x,y,fruit.x,

fruit.y,nTail,tailX,tailY,score,gameover);

}

return 0;

}

//the function to draw

//include the map , snake and the fruit

void draw(int width,int height,int x,int y,

int fruitX,int fruitY,int nTail,

int tailX,int tailY,int score){

system("clear");//clear

//set the width wall

for(int i=0;i<width+2;i++)

cout<<"#";

cout<<endl;

//set the height wall

for(int i=0;i<height;i++){

for(int j=0;j<width;j++){

//set the wall for left

if(j==0)

cout<<"#";

//set the snake hand

if(i==y && j==x)

cout<<"O";

//show the fruit

else if(i==fruitX && j==fruitY )

cout<<"F";

//show the snake tail

//the tail will be follow the snake hand to move

else{

bool print = false;

for(int k=0; k<nTail;k++){

if(tailX[k]==j&&tailY[k]==i){

cout<<"o";

print = true;

}

}

//after the snake tail move

//reset the tail to space

if(!print)

cout<<" ";

}

//set the wall for right

if(j==width-1)

cout<<"#";

}

cout<<endl;

}

for(int i=0;i<width+2;i++)

cout<<"#";

cout<<endl;

//show the score

cout<<"Score :"<<score<<endl;

}

//input thing in the game

//it use w a s d to control the snake

//w is up, s is down, a is left, d is right

void input(bool gameover){

//get the button from key board

//it don't want pass the enter

if(\_kbhit()){

//the key in key board

//a mean turn left

//d mean turn right

//w mean go up

//s mean go down

switch(\_getch()){

case 'a':

dir = LEFT;

break;

case 'd':

dir = RIGHT;

break;

case 'w':

dir = UP;

break;

case 's':

dir = DOWN;

break;

case 'x':

gameover = true;

break;

}

}

}

//the function for show the snake hand and body

//it also calculate the score

void logic(int width,int height,int x,int y,

int fruitX,int fruitY,int nTail,

int tailX,int tailY,int score,bool gameover){

//mark down the point for the snake tail

//and mark the number of tail

//that mean how long is the tail

int prevX = tailX[0];

int prevY = tailY[0];

int prev2X, prev2Y;

tailX[0] = x;

tailY[0] = y;

//get the number for the tail

//and mark all the point for the tail

for(int i=1;i<nTail;i++){

prev2X = tailX[i];

prev2Y = tailY[i];

tailX[i] = prevX;

tailY[i] = prevY;

prevX = prev2X;

prevY = prev2Y;

}

//get the button from the key board

//x is for left and right

//if x -1, that mean turn left

//if x +1, that mean turn right

//y is fro up and down

//if y-1. that mean up

//if y+1, that mean down

switch(dir){

case LEFT:

x--;

break;

case RIGHT:

x++;

break;

case UP:

y--;

break;

case DOWN:

y++;

break;

default:

break;

}

//if the snake hand in to the wall

//the game is over

if(x>width||x<0||y>height||y<0)

gameover = true;

//if the snake hand get in to snake tail

//the game is over

for(int i=0;i<nTail;i++)

if(tailX[i]==x&&tailY[i]==y)

gameover = true;

//if the snake hand get in to the fruit

//that mean the snake eat the fruit

//the fruit will reset

//and the tail of snake will +1

if(x==fruitX && y==fruitY){

score+=10;

fruitX = rand()% width;

fruitY = rand()% height;

nTail++;

}

}