MonsterWorldGame111.cpp | Monster111.h | MonsterWorld111.h | VariousMonsters111.h

Week11_Homework1 | (Global Scope)

```cpp
#include "MonsterWorld111.h"
#include "VariousMonsters111.h"
#include "Human111.h"
#include <time.h>

void main()
{
    srand((unsigned int)time(NULL));
    int w = 16, h = 8;
    MonsterWorld game(w, h);

    game.add(new Zombie("허접한좀비", "§", rand() % w, rand() % h));
    game.add(new Vampire("뱀파이어짱", "★", rand() % w, rand() % h));
    game.add(new KGhost("어쩌다귀신", "♥", rand() % w, rand() % h));
    game.add(new Jiangshi("못먹어도고", "↔", rand() % w, rand() % h, true));
    game.add(new Jiangshi("못먹어세로", "↕", rand() % w, rand() % h, false));
    game.add(new Human("미래의인류", "♀", rand() % w, rand() % h));
    game.play(500, 10);
    printf("------게임 종료----------\n");
}
```

C:₩Users₩user₩source₩repos₩Week11_Homework1₩x64₩Debug₩Week11_Homework1.exe

```
[ Monster World (relaxed World) ]
■■■§■■■■♥■■■■■■■
■■■■■■■■■■■■■■■■
■■■■■■↔♀■■■■■■■■
■■■↕■■■■■■■■■■■■
■■■■■■■■■■■■■■■■
■■■■■■■■■■■■■■■■
■■■■■■■■■■■■■■■■
■■■■■■■■■■■■★■■■
■■■■■■■■■■■■■■■■

전체 이동 횟수 = 0
남은 아이템 수 = 128
        허접한좀비§:0:0
        뱀파이어짱★:0:0
        어쩌다귀신♥:0:0
        못먹어도고↔:0:0
        못먹어세로↕:0:0
        미래의인류♀:0:0
엔터를 누르세요...
```

79 % | No issues found | Ln: 1 | Ch: 1 | TABS | CRLF

MonsterWorldGame111.cpp | Monster111.h | MonsterWorld111.h | VariousMonsters111.h

Week11_Homework1 | (Global Scope)

```cpp
#pragma once
#include "Canvas111.h"
#include "Point111.h"

class Monster
{
public:
    string name, icon;
    int nItem;
    Point q, p;
    int nSleep;
    double dist;
    double total;

    void clip(int maxx, int maxy) { p(maxx, maxy); }
    void eat(int** map) {
        if (map[p.y][p.x] == 1)
        {
            map[p.y][p.x] = 0;
            nItem++;
        }
        dist += (double)(p - q);
        total += (double)(p - q);
        q = p;
        if (dist > 20)
        {
            dist = 0;
            nSleep = 10;
        }
    }
    bool isSleep() {
        if (nSleep > 0)
        {
            nSleep--;
            return true;
```

C:₩Users₩user₩source₩repos₩Week11_Homework1₩x64₩Debug₩Week11_Homework1.exe

```
[ Monster World (relaxed World) ]
■■.■.....■.....§
■■.■■■■■■■■■■■■■
■↔...■♀...■■■■■■■
.■.■■■■■■■■■■♥■■■
■■■↕■■■■■.■■...■■
■■.■■■■■■■■■.★■■■
.■.■■■■■■■■■...■■
■■■■.■■■■■■■■■■■■

전체 이동 횟수 = 22
남은 아이템 수 = 88
        허접한좀비§:11:0
        뱀파이어짱★:10:9
        어쩌다귀신♥:7:5
        못먹어도고↔:6:4
        못먹어세로↕:6:4
        미래의인류♀:0:0
```

79 % | No issues found | Ln: 1 | Ch: 1 | TABS | CRLF

MonsterWorldGame111.cpp | Monster111.h | MonsterWorld111.h | VariousMonsters111.h

Week11_Homework1 | Monster

```cpp
            nSleep--;
            return true;
        }
        else return false;
    }
public:
    Monster(string n = "무명괴물", string i = "※", int x = 0, int y = 0)
        : name(n), icon(i), nItem(0),
        p(x, y) , q(x, y), nSleep(0), dist(0.0), total(0.0) { }
    virtual ~Monster() { cout << icon << nItem << "거리:" << total << endl; }

    void draw(Canvas& canvas) { canvas.draw(p, icon); }
    virtual void move(int** map, int maxx, int maxy) {
        if (!isSleep())
        {
            int num = rand() % 9 + 1;
            p = p + Point(num % 3 - 1, num / 3 - 1);
            clip(maxx, maxy);
            eat(map);
        }
    }
    void print() { cout << "\t" << name << icon << ":" << nItem << ":" << nSleep << endl; }
};
```

C:₩Users₩user₩source₩repos₩Week11_Homework1₩x64₩Debug₩Week11_Homework1.exe

```
[ Monster World (relaxed World) ]
■■.■.....■.....§
■■.■.■■.■.■...■■■■.
...↔...♀.....■.■■
...■■.♥■.■■.■.■■
■■.■★.■.........■■
■■.■.....■■■■■■■■■
...■.■■.........
■.↕.....■.......

전체 이동 횟수 = 216
남은 아이템 수 = 38
        허접한좀비§:11:0
        뱀파이어짱★:31:0
        어쩌다귀신♥:31:2
        못먹어도고↔:10:0
        못먹어세로↕:7:0
        미래의인류♀:0:0
```

79 % | No issues found | Ln: 16 | Ch: 23 | Col: 26 | TABS | CRLF

MonsterWorldGame111.cpp | Monster111.h | MonsterWorld111.h | VariousMonsters111.h

Week11_Homework1 > MonsterWorld

```cpp
#pragma once
#include "Canvas111.h"
#include "Monster111.h"
#include "Matrix111.h"
#include <Windows.h>
#define MAXMONS 8

class MonsterWorld
{
    Matrix world;
    int xMax, yMax, nMon, nMove;
    Monster* pMon[MAXMONS];
    Canvas canvas;

    int& Map(int x, int y) { return world.elem(x, y); }
    bool isDone() { return countItems() == 0; }
    int countItems() {
        int nItems = 0;
        for (int y = 0; y < yMax; y++)
            for (int x = 0; x < xMax; x++)
                if (Map(x, y) > 0) nItems++;
        return nItems;
    }
    void print() {
        canvas.clear(". ");
        for (int y = 0; y < yMax; y++)
            for (int x = 0; x < xMax; x++)
                if (Map(x, y) > 0) canvas.draw(x, y, "■");
        for (int i = 0; i < nMon; i++)
            pMon[i]->draw(canvas);
        canvas.print("[ Monster World (relaxed World) ]");

        cerr << " 전체 이동 횟수 = " << nMove << endl;
        cerr << " 남은 아이템 수 = " << countItems() << endl;
        for (int i = 0; i < nMon; i++) pMon[i]->print();
    }
public:
    MonsterWorld(int w, int h) : world(h, w), canvas(w, h), xMax(w), yMax(h) {
        nMon = 0;
        nMove = 0;
        for (int y = 0; y < yMax; y++)
            for (int x = 0; x < xMax; x++) Map(x, y) = 1;
    }
    ~MonsterWorld() {
        for (int i = 0; i < nMon; i++)
            delete pMon[i];
    }
    void add(Monster* m) { if (nMon < MAXMONS) pMon[nMon++] = m; }
    void play(int maxwalk, int wait) {
        print();
        cerr << " 엔터를 누르세요...";
        getchar();
        for (int i = 0; i < maxwalk; i++)
        {
            for (int k = 0; k < nMon; k++)
                pMon[k]->move(world.Data(), xMax, yMax);
            nMove++;
            print();
            if (isDone()) break;
            Sleep(wait);
        }
    }
};
```

79 % | 0 | 1 | Ln: 31 | Ch: 41 | Col: 47 | TABS | CRLF

C:₩Users₩user₩source₩repos₩Week11_Homework1₩x64₩Debug₩Week11_Homework1.exe

MonsterWorldGame111.cpp | Monster111.h | MonsterWorld111.h | VariousMonsters111.h

Week11_Homework1 > (Global Scope)

```cpp
        cerr << " 남은 아이템 수 = " << countItems() << endl;
        for (int i = 0; i < nMon; i++) pMon[i]->print();
    }
public:
    MonsterWorld(int w, int h) : world(h, w), canvas(w, h), xMax(w), yMax(h) {
        nMon = 0;
        nMove = 0;
        for (int y = 0; y < yMax; y++)
            for (int x = 0; x < xMax; x++) Map(x, y) = 1;
    }
    ~MonsterWorld() {
        for (int i = 0; i < nMon; i++)
            delete pMon[i];
    }
    void add(Monster* m) { if (nMon < MAXMONS) pMon[nMon++] = m; }
    void play(int maxwalk, int wait) {
        print();
        cerr << " 엔터를 누르세요...";
        getchar();
        for (int i = 0; i < maxwalk; i++)
        {
            for (int k = 0; k < nMon; k++)
                pMon[k]->move(world.Data(), xMax, yMax);
            nMove++;
            print();
            if (isDone()) break;
            Sleep(wait);
        }
    }
};
```

79 % | 0 | 1 | Ln: 7 | Ch: 1 | TABS | CRLF

C:₩Users₩user₩source₩repos₩Week11_Homework1₩x64₩Debug₩Week11_Homework1.exe

```
[ Monster World (relaxed World) ]
■. ↕■. . . . . ■. . . . . §
■♥. . ■. . . . . ■■. ■.
. . . . ↔♀. . . . . . . .
. . . ■. . . . . ■. . . ■■
. . . . . . . . . . . . ■.
. . . . . . . . . . . . . ■
. . . . . . . . . . . . .
. . . . ★. . . . . . . . .

 전체 이동 횟수 = 468
 남은 아이템 수 = 14
        허접한좀비§:11:0
        뱀파이어짱★:41:2
        어쩌다귀신♥:42:8
        못먹어도고↔:13:1
        못먹어세로↕:7:0
        미래의인류♀:0:0
```

MonsterWorldGame111.cpp | Monster111.h | MonsterWorld111.h | VariousMonsters111.h

Week11_Homework1 > Jiangshi

```cpp
#pragma once
#include "Monster111.h"

class Zombie : public Monster {
public:
    Zombie(string n = "허접좀비", string i = "§", int x = 0, int y = 0)
        : Monster(n, i, x, y) {}
    ~Zombie() { cout << " Zombie"; }
};

class Vampire : public Monster {
public:
    Vampire(string n = "뱀파이어", string i = "★", int x = 0, int y = 0)
        : Monster(n, i, x, y) {}
    ~Vampire() { cout << " Vampire"; }

    void move(int** map, int maxx, int maxy) {
        if (!isSleep()) {
            int dir = rand() % 4;
            if (dir == 0) p[0]--;
            else if (dir == 1) p[0]++;
            else if (dir == 2) p[1]--;
            else p[1]++;
            clip(maxx, maxy);
            eat(map);
        }
    }
};

class KGhost : public Monster {
public:
    KGhost(string n = "처녀귀신", string i = "♥", int x = 0, int y = 0)
        : Monster(n, i, x, y) {}
    ~KGhost() { cout << " KGhost"; }
```

79 % | No issues found | Ln: 53 | Ch: 8 | Col: 14 | TABS | CRLF

Microsoft Visual Studio Debug Console

```
[ Monster World (relaxed World) ]
■. . ■. . . . . ■. . . . . §
■♥. . ■. . . . . ■■. ■.
. . ↔. . . ♀. . . . . . . .
. . ↕■. . . . . ■. . . ■■
. . . . . . . . . . . . ■.
. . . . . . . . . . . . . ■
. . . . . . . . . . . . .
. ★. . . . . . . . . . . .

 전체 이동 횟수 = 500
 남은 아이템 수 = 14
        허접한좀비§:11:0
        뱀파이어짱★:41:2
        어쩌다귀신♥:42:5
        못먹어도고↔:13:0
        못먹어세로↕:7:0
        미래의인류♀:0:0
------게임 종료-----------
 Zombie§11거리:12
 Vampire★41거리:315
 KGhost♥42거리:866.79
 Jangshi↔13거리:413
 Jangshi↕7거리:392
[Human ]♀0거리:0

C:₩Users₩user₩source₩repos₩Week11_Homework1₩x64₩Debug₩Week11_Homework1.exe (p
Press any key to close this window . . .
```

MonsterWorldGame111.cpp | Monster111.h | MonsterWorld111.h | VariousMonsters111.h

Week11_Homework1 | (Global Scope)

```
    ~KGhost() { cout << " KGhost"; }

    void move(int** map, int maxx, int maxy) {
        if (!isSleep()) {
            p = Point(rand() % maxx, rand() % maxy);
            clip(maxx, maxy);
            eat(map);
        }
    }
};

class Jiangshi : public Monster {
    bool bHori;
public:
    Jiangshi(string n = "대륙강시", string i = "↔", int x = 0, int y = 0, bool bH = true)
        : Monster(n, i, x, y), bHori(bH) {}
    ~Jiangshi() { cout << " Jangshi"; }

    void move(int** map, int maxx, int maxy) {
        if (!isSleep()) {
            int dir = rand() % 2;
            int jump = rand() % 2 + 1;
            if (bHori) p[0] += ((dir == 0) ? -jump : jump);
            else p[1] += ((dir == 0) ? -jump : jump);
            clip(maxx, maxy);
            eat(map);
        }
    }
};
```

Microsoft Visual Studio Debug Console

```
[ Monster World (relaxed World) ]
■. . ■. . . . . ■. . . . . §
■♥. . ■. . . . . ■■. ■.
. . ↔. . . ♀. . . . . . . .
. . ↕■. . . . . ■. . . ■■
. . . . . ■.
. . . . . ■
. . . . . .
. ★. . . . . .

전체 이동 횟수 = 500
남은 아이템 수 = 14
        허접한좀비§:11:0
        뱀파이어짱★:41:2
        어쩌다귀신♥:42:5
        못먹어도고↔:13:0
        못먹어세로↕:7:0
        미래의인류♀:0:0
------게임 종료----------
Zombie§11거리:12
Vampire★41거리:315
KGhost♥42거리:866.79
Jangshi↔13거리:413
Jangshi↕7거리:392
[Human ]♀0거리:0

C:₩Users₩user₩source₩repos₩Week11_Homework1₩x64₩Debug₩Week11_Homework1.exe (p
Press any key to close this window . . .
```

79 % | No issues found | Ln: 3 | Ch: 1 | TABS | CRLF

MonsterWorldGame111.cpp | Monster111.h | MonsterWorld111.h | VariousMonsters111.h | Human111.h | Canvas111.h | Point111.h | Matrix111.h

Week11_Homework1 | Human

```
#pragma once
#include "Monster111.h"
#include <conio.h>
enum Direction { Left = 75, Right = 77, Up = 72, Down = 80 };

class Human : public Monster {
public:
    Human(string n = "미래인류", string i = "♀", int px = 0, int py = 0)
        : Monster(n, i, px, py) {}
    ~Human() { cout << "[Human ]"; }

    int getDirKey() { return _getche() == 224 ? _getche() : 0; }

    void move(int** map, int maxx, int maxy) {
        if (_kbhit())
        {
            char ch = getDirKey();
            if (ch == Left) p[0]--;
            else if (ch == Right) p[0]++;
            else if (ch == Up) p[1]--;
            else if (ch == Down) p[1]++;
            else return;
            clip(maxx, maxy);
            eat(map);
        }
    }
};
```

79 % | No issues found | Ln: 27 | Ch: 3 | TABS | CRLF

MonsterWorldGame111.cpp | Monster111.h | MonsterWorld111.h | VariousMonsters111.h | Human111.h | Canvas111.h* | Point111.h | Matrix111.h

Week11_Homework1 | Canvas

```
#pragma once
#include <iostream>
#include <string>
#include "Point111.h"
#define MAXLINES 100
using namespace std;

class Canvas
{
    string line[MAXLINES];
    int xMax, yMax;
public:
    Canvas(int nx = 10, int ny = 10) : xMax(nx), yMax(ny) {
        for (int y = 0; y < yMax; y++)
            line[y] = string(xMax * 2, ' ');
    }
    void draw(int x, int y, string val) {
        if (x >= 0 && y >= 0 && x < xMax && y < yMax)
            line[y].replace(x * 2, 2, val);
    }
    void draw(Point& p, string val) { draw(p.x, p.y, val); }
    void clear(string val = "  ") {
        for (int y = 0; y < yMax; y++)
            for (int x = 0; x < xMax; x++)
                draw(x, y, val);
    }
    void print(const char* title = "<My Canvas>") {
        system("cls");
        cout << title << endl;
        for (int y = 0; y < yMax; y++)
            cout << line[y] << endl;
        cout << endl;
    }
};
```

79 % | No issues found | Ln: 34 | Ch: 3 | TABS | CRLF

MonsterWorldGame111.cpp | Monster111.h | MonsterWorld111.h | VariousMonsters111.h | Human111.h | Canvas111.h* | Point111.h* | Matrix111.h

Week11_Homework1 | Point

```cpp
#pragma once
#include <iostream>
#include <cmath>
using namespace std;

class Point
{
    int x, y;
    friend class Monster;
    friend class Canvas;
public:
    Point(int xx = 0, int yy = 0) :x(xx), y(yy) { }

    int& operator[] (int id) {
        if (id == 0) return x;
        else if (id == 1) return y;
        else exit(0);
    }
    operator double() { return sqrt((double)x * x + y * y); }
    void operator() (int maxx, int maxy) {
        if (x < 0) x = 0;
        if (x >= maxx) x = maxx - 1;
        if (y < 0) y = 0;
        if (y >= maxy) y = maxy - 1;
    }

    Point operator- () { return Point(-x, -y); }
    bool operator== (Point& p) { return x == p.x && y == p.y; }
    bool operator!= (Point& p) { return x != p.x || y != p.y; }
    Point operator- (Point& p) { return Point(x - p.x, y - p.y); }
    Point operator+ (Point& p) { return Point(x + p.x, y + p.y); }
    void operator+= (Point& p) { x += p.x, y += p.y; }
    void operator-= (Point& p) { x -= p.x, y -= p.y; }
};
```

79 % | 0 | 1 | Ln: 34 | Ch: 3 | TABS | CRLF

MonsterWorldGame111.cpp | Monster111.h | MonsterWorld111.h | VariousMonsters111.h | Human111.h | Canvas111.h* | Point111.h* | Matrix111.h

Week11_Homework1 | Matrix | print(const char * str)

```cpp
#pragma once
#include <iostream>
#include <iomanip>
using namespace std;

class Matrix
{
    int rows, cols;
    int** mat;
public:
    Matrix(int r = 0, int c = 0) : rows(r), cols(c), mat(NULL) {
        mat = new int* [rows];
        for (int i = 0; i < rows; i++) mat[i] = new int[cols];
    }
    ~Matrix() {
        if (mat != NULL)
        {
            for (int i = 0; i < rows; i++)
                delete[] mat[i];
            delete[] mat;
        }
    }
    int& elem(int x, int y) { return mat[y][x]; }
    int Rows() { return rows; }
    int Cols() { return cols; }
    int** Data() { return mat; };
    void print(const char* str = "Mat") {
        cout << str << " " << rows << "x" << cols << endl;
        for (int i = 0; i < rows; i++)
        {
            for (int j = 0; j < cols; j++)
                cout << setw(4) << mat[i][j];
            cout << "\n";
        }
    }
```

79 % | No issues found | Ln: 31 | Ch: 34 | Col: 43 | TABS | CRLF

MonsterWorldGame111.cpp | Monster111.h | MonsterWorld111.h | VariousMonsters111.h | Human111.h | Canvas111.h* | Point111.h* | Matrix111.h

Week11_Homework1 | (Global Scope)

```cpp
        }
    }
    void setRand(int val = 100) {
        if (mat != NULL)
        {
            for (int i = 0; i < rows; i++)
                for (int j = 0; j < cols; j++)
                    mat[i][j] = (rand() % val);
        }
    }
};
```

79 % | No issues found | Ln: 5 | Ch: 1 | TABS | CRLF