

Assignment#3

Submitted by:

Name: Mubbara Younas

Department: information technology

Semester: 6th

Roll_no: 089308(morning)

Subject: Artificial intelligence

Submitted to:

Prof. Sehrish khan

Report on Configuring and Executing Vacuum Cleaner World using Qt

Step 1:

Downloading the Vacuum Cleaner World Project:

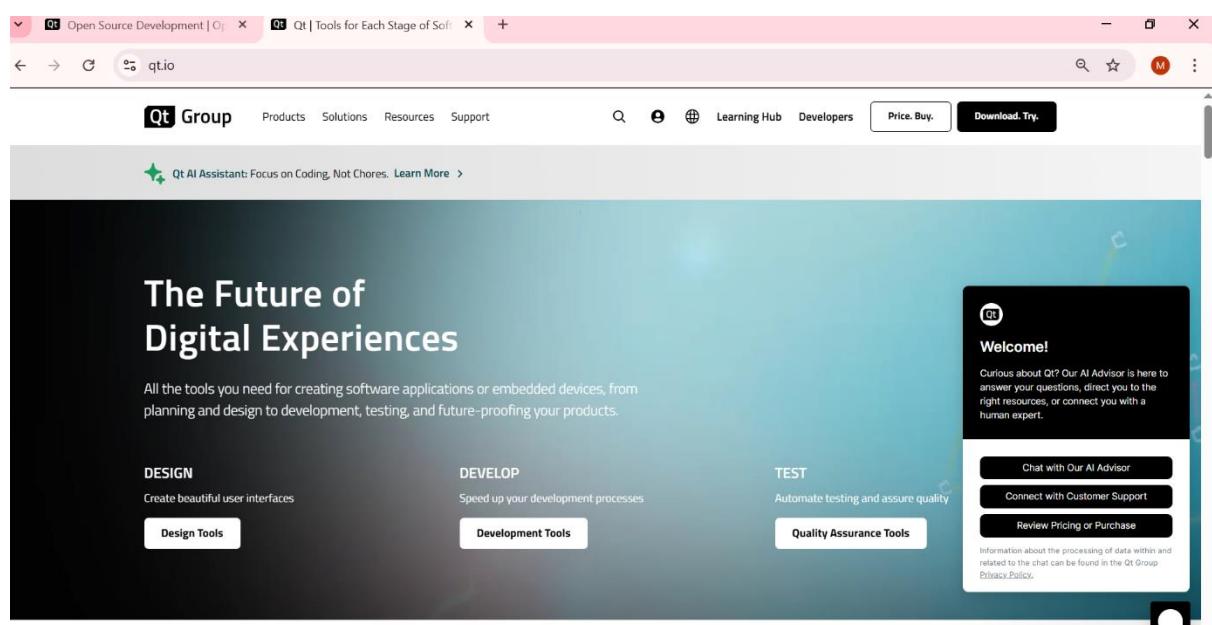
1. Open the GitHub Repository through this providing link https://github.com/Minoru/vacuum_cleaner_world-qt.
2. Click on the file and manually download the files.
3. Make a folder on desktop, and then copy paste those files in the new making folder.

Step 2:

Installing Qt open Source:

To run project, Qt needs to be installed.

1. Visit the official QT website: Qt download Page.

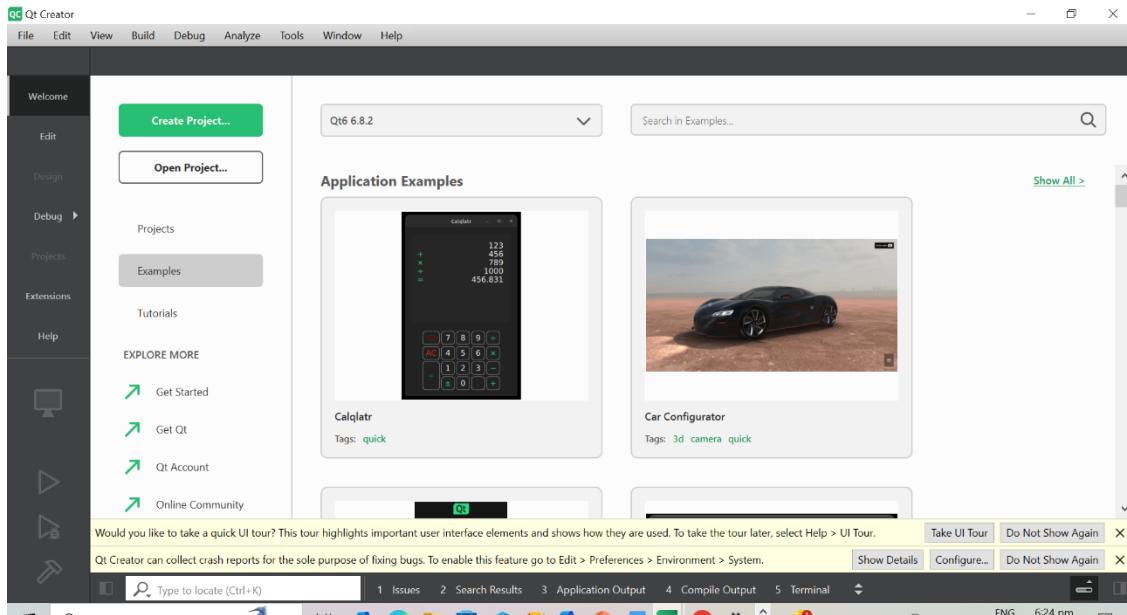


2. Download the Qt open source download.
3. Create account on this. If the already account then login.
4. Ans then click on next option.
5. After this, click on install button.
6. Completion Installation, then click on finish button.

Step 3:

Opening the Project in Qt Creator:

1. Open Qt Creator



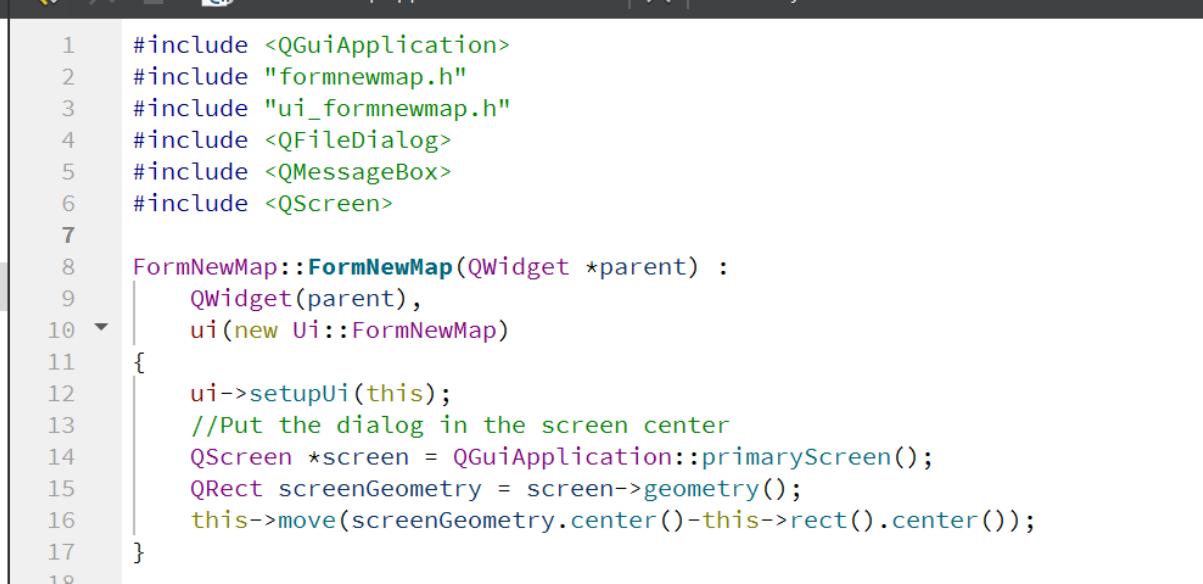
2. Click on “Open Project”.
3. Navigate to the extracted project folder.
4. Select the Agent.pro file to load the project.
5. Configure the project by selecting the appropriate compiler and kit.

Step 4:

Fixing QDesktopWidget Deprecation:

1. Open the file where QDesktopWidget is used(likely in formanewmap.cpp)

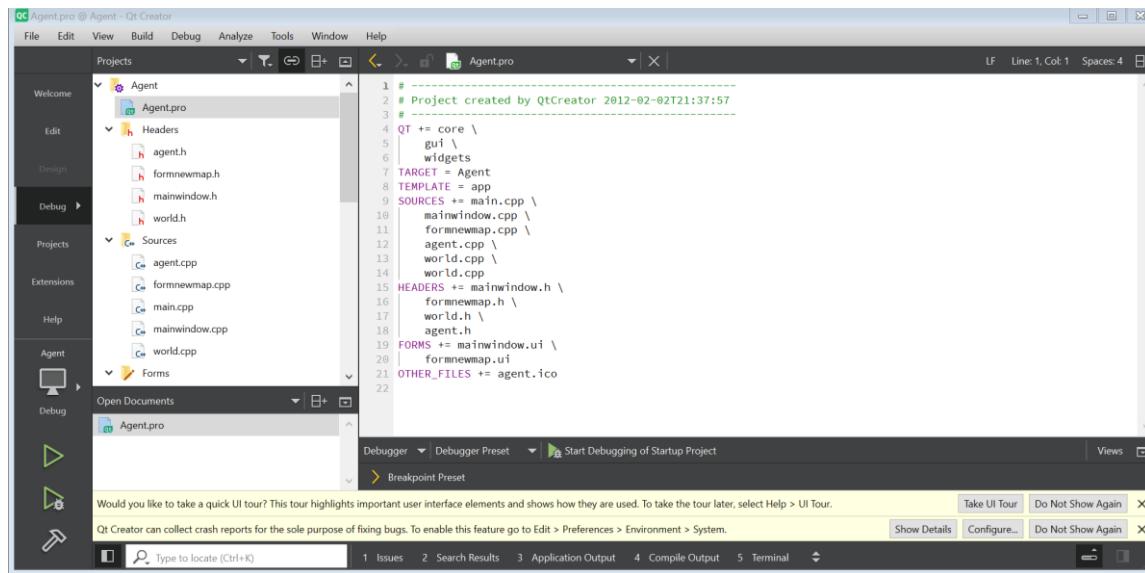
2. Replace the old code with this updated version.



```
1 #include <QGuiApplication>
2 #include "formnewmap.h"
3 #include "ui_formnewmap.h"
4 #include <QFileDialog>
5 #include <QMessageBox>
6 #include <QScreen>
7
8 FormNewMap::FormNewMap(QWidget *parent) :
9     QWidget(parent),
10    ui(new Ui::FormNewMap)
11 {
12     ui->setupUi(this);
13     //Put the dialog in the screen center
14     QScreen *screen = QGuiApplication::primaryScreen();
15     QRect screenGeometry = screen->geometry();
16     this->move(screenGeometry.center()-this->rect().center());
17 }
```

3. Save the file.

First file code(agent.pro) in the Qt Agent folder that is download from given github repository
(Agent.pro)



Second source code file(agent.h) in the Qt Agent folder header section that is downloaded from the given repository.

The screenshot shows the Qt Creator interface with the 'Agent' project selected. The left sidebar shows the project structure with 'Headers' expanded, displaying files like agent.h, formnewmap.h, mainwindow.h, and world.h. The code editor on the right shows the content of agent.h, which defines a class Agent with actions like moveUp, moveDown, moveLeft, moveRight, suck, and idle. It also includes a function for the agent to look at the world.

```

1 #ifndef AGENT_H
2 #define AGENT_H
3
4 class World;
5
6 class Agent
7 {
8     public:
9         /* actions an agent can perform */
10        enum actions
11        {
12            moveUp,
13            moveDown,
14            moveLeft,
15            moveRight,
16            suck,
17            idle
18        };
19
20        /* this function is get called when it's time for agent to look at the
21           | * world and make some difference */
22        void lookAtWorld(actions act, int actions);
23    };
24
25 #endif

```

Third source code file (formnewmap.h) in the Qt Agent header section that is downloaded from given repository

The screenshot shows the Qt Creator interface with the 'Agent' project selected. The left sidebar shows the project structure with 'Headers' expanded, displaying files like agent.h, formnewmap.h, mainwindow.h, and world.h. The code editor on the right shows the content of formnewmap.h, which defines a class FormNewMap that inherits from QWidget. It includes slots for on_selectMapFileButton_clicked, on_okButton_clicked, on_cancelButton_clicked, and on_lifetimeEdit_returnPressed. It also includes signals for on_okButton_clicked, on_cancelButton_clicked, on_lifetimeEdit_returnPressed, on_mapfilenameEdit_returnPressed, and on_TestCaseEdit_returnPressed.

```

1 #ifndef FORMNEWMAP_H
2 #define FORMNEWMAP_H
3
4 #include <QWidget>
5
6 namespace Ui {
7     class FormNewMap;
8 }
9
10 class FormNewMap : public QWidget
11 {
12     Q_OBJECT
13
14 public:
15     explicit FormNewMap(QWidget *parent = 0);
16     ~FormNewMap();
17
18 private slots:
19     void on_selectMapFileButton_clicked(); // Slots named on_foo_bar
20
21     void on_okButton_clicked(); // Slots named on_foo_bar
22
23     void on_cancelButton_clicked(); // Slots named on_foo_bar
24
25     void on_lifetimeEdit_returnPressed(); // Slots named on_foo_bar
26
27     void on_mapfilenameEdit_returnPressed(); // Slots named on_foo_bar
28
29     void on_TestCaseEdit_returnPressed(); // Slots named on_foo_bar
30
31 private:
32     Ui::FormNewMap *ui;
33     signals:
34     //send data to main window
35     void sendData(QString fileName, int lifeTime, int testCase);
36 };
37
38 #endif // FORMNEWMAP_H

```

Fourth source code file (mainwindow.h) in the Qt header section that is downloaded from given github repository.

mainwindow.h @ Agent - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

Projects Agent Agent.pro Headers agent.h formnewmap.h mainwindow.h world.h Sources agent.cpp formnewmap.cpp main.cpp mainwindow.cpp world.cpp Forms

mainwindow.h

```

1 ifndef MAINWINDOW_H
2 define MAINWINDOW_H
3
4 #include <QResizeEvent>
5 #include < QMainWindow>
6 #include "world.h"
7 #include <QGraphicsScene>
8
9
10 namespace Ui {
11     class MainWindow;
12 }
13
14 class MainWindow : public QMainWindow
15 {
16     Q_OBJECT
17
18 public:
19     explicit MainWindow(QWidget *parent = 0);
20     ~MainWindow();
21     void resizeEvent(QResizeEvent * );
22
23 private slots:
24     void on_quitButton_clicked();
25
26     void on_selectMapButton_clicked();
27
28     void on_doOneStepButton_clicked();
29
30
31     void on_nextRunButton_clicked();
32
33     void on_doAllRunsButton_clicked();
34
35     void on_displayButton_clicked();
36
37
38 public slots:
39     void onNewMapData(QString filename, int li);
40     void onDrawOneStep();
41     void onRestore();
42
43 private:
44     Ui::MainWindow *ui;
45
46     bool LoadMap();
47     void DrawMap();
48     void RefreshStats();
49     void ManageSituation();
50
51     QString fileName;           // .map file
52     int lifeTime;
53     int testCase;
54
55     int currentRun;
56     double totalDirtyDegree;
57     double totalConsumedEnergy;
58
59     World *w;

```

The screenshot shows the Qt Creator IDE interface. The left sidebar has sections for Welcome, Edit (selected), Design, Debug, Projects, Extensions, Help, Agent, and Debug. The Projects section shows a project named 'Agent' with files like Agent.pro, Headers (agent.h, formnewmap.h, mainwindow.h), Sources (agent.cpp, formnewmap.cpp, main.cpp, mainwindow.cpp, world.cpp), and Forms. The main editor area displays the code for mainwindow.h:

```
mainwindow.h @ Agent - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects mainwindow.h totalDirtyDegree: double
mainwindow.h
39 void onNewMapData(QString filename, int lifetime, int testcase);
40 void onDrawOneStep();
41 void onRestore();
42
43 private:
44     Ui::MainWindow *ui;
45
46     bool LoadMap();
47     void DrawMap();
48     void RefreshStats();
49     void ManageSituation();
50
51     QString fileName; // .map file
52     int lifeTime;
53     int testCase;
54
55     int currentRun;
56     double totalDirtyDegree; // totalConsumedEnergy;
57
58     World *w;
59     QGraphicsScene* scene; // graphics scene
60
61     static const int RECTANGLE_SIZE = 40;
62
63 };
64
65 #endif // MAINWINDOW_H
```

Sixth source code file(world.h) in the Qt header section that is downloaded from the given github repository

The screenshot shows the Qt Creator IDE interface. The left sidebar has sections for Welcome, Edit (selected), Design, Debug, Projects, Extensions, Help, Agent, and Debug. The Projects section shows a project named 'Agent' with files like Agent.pro, Headers (agent.h, formnewmap.h, mainwindow.h), Sources (agent.cpp, formnewmap.cpp, main.cpp, mainwindow.cpp, world.cpp), and Forms. The main editor area displays the code for world.h:

```
world.h @ Agent - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects world.h <Select Symbol> LF Line: 1, Col: 1
mainwindow.h
1 ifndef WORLD_H
2 define WORLD_H
3
4 #include <QTranslator>
5 #include <string>
6 using std::string;
7 #include <sstream>
8 using std::istringstream;
9 #include <fstream>
10 using std::ifstream;
11 using std::ios_base;
12 #include <vector>
13 using std::vector;
14 #include <cstdlib>
15 #include <time.h>
16 #include "agent.h"
17
18 class World
19 {
20 public:
21     /* World should be created from the map file.
22      * Pointer to parent widget is required so constructor can display some
23      * warning and error dialogs */
24     World(string);
25
26     /* this function is called by agent at each step. It applies agent's action
27      * to world and updates it (e.g. randomly adds dirt here and there) */
28     void performAction(Agent::actions);
```

world.h @ Agent - Qt Creator

```

File Edit View Build Debug Analyze Tools Window Help
Projects Agent Headers Sources Forms Open Documents
world.h
13 using std::vector;
14 #include <cstdlib>
15 #include <time.h>
16 #include "agent.h"
17
18 class World
19 {
20 public:
21     /* World should be created from the map file.
22      * Pointer to parent widget is required so constructor can display some
23      * warning and error dialogs */
24     World(string);
25
26     /* this function is called by agent at each step. It applies agent's action
27      * to world and updates it (e.g. randomly adds dirt here and there) */
28     void performAction(Agent::actions);
29
30     void doneStep();
31
32     void resetMap();
33
34     QString getErrorMessage()
35     {
36         return errorMessage;
37     }
38
39     vector< vector<int> * > getWorld()
40     {
41         return world;
42     }

```

world.h @ Agent - Qt Creator

```

File Edit View Build Debug Analyze Tools Window Help
Projects Agent Headers Sources Forms Open Documents
world.h
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
vector< vector<int> * > getWorld()
{
    return world;
}

int getWorldWidth()
{
    return world_width;
}

int getWorldHeight()
{
    return world_height;
}

int getCurrentTime()
{
    return currentTime;
}

Agent::actions getLastAgentAction()
{
    return lastAgentAction;
}

int getDirtyDegree()
{
    return dirtyDegree;
}

```

world.h @ Agent - Qt Creator

```

File Edit View Build Debug Analyze Tools Window Help
Projects Agent Headers Sources Forms Open Documents
world.h
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
    return lastAgentAction;
}

int getDirtyDegree()
{
    return dirtyDegree;
}

int getConsumedEnergy()
{
    return consumedEnergy;
}

int getAgentPosX()
{
    return agentPosX;
}

int getAgentPosY()
{
    return agentPosY;
}

bool isJustBumped()
{
    return justBumped;
}

float getDirtyProbability()

```

Qt world.h @ Agent - Qt Creator

```

File Edit View Build Debug Analyze Tools Window Help
Projects Agent Headers Sources Forms Open Documents
  Agent Agent.pro Headers agent.h formnewmap.h mainwindow.h world.h
  Sources agent.cpp formnewmap.cpp main.cpp mainwindow.cpp world.cpp
  Forms
  Open Documents mainwindow.h world.h
  Line: 115, Col: 6

```

```

float getDirtyProbability()
{
    return dirtyProbability;
}

private:
/* World is a set of columns (vector of pointers to vectors, in fact). That
 * allows us to write quite intuitive code (e.g. world[x]->at(y)). Note,
 * though, that beginning of coordinates is in the upper left corner. Each
 * cell (which is int by the way) represents an amount of dirt in a given
 * cell. Initially it is all zeros. Amount of dirt is incremented randomly
 * at each step */
vector< vector<int>*> world;
int world_width, world_height;
/* agent's current position */
int agentPosX, agentPosY;
/* we need to store initial position to reset the map before each run */
int initialAgentPosX, initialAgentPosY;
/* true if agent just bumped into the wall */
bool justBumped;
/* probability of cell's dirty level increasing by one */
float dirtyProbability;
/* agent is an entity that would perform some actions on our world */
Agent* agent;

int currentTime;
unsigned long dirtyDegree;
unsigned long consumedEnergy;

```

Qt world.h @ Agent - Qt Creator

```

File Edit View Build Debug Analyze Tools Window Help
Projects Agent Headers Sources Forms Open Documents
  Agent Agent.pro Headers agent.h formnewmap.h mainwindow.h world.h
  Sources agent.cpp formnewmap.cpp main.cpp mainwindow.cpp world.cpp
  Forms
  Open Documents mainwindow.h world.h
  Line: 125, Col: 1

```

```

int initialAgentPosX, initialAgentPosY;
/* true if agent just bumped into the wall */
bool justBumped;
/* probability of cell's dirty level increasing by one */
float dirtyProbability;
/* agent is an entity that would perform some actions on our world */
Agent* agent;

int currentTime;
unsigned long dirtyDegree;
unsigned long consumedEnergy;

Agent::actions lastAgentAction;

/* how obstacles are indicated at map */
static const int OBSTACLE = -1;

static const char MAP_OBSTACLE = '0',
MAP_ROAD = '-';

QString errorMessage;

/* we use that value to keep world's random generator independent of
 * everything else */
unsigned int seed;
};

#endif

```

Seven source code file (agent.cpp) in the Qt Agent folder Sources section that is downloaded from the given github repository.

Qt agent.cpp @ Agent - Qt Creator

```

File Edit View Build Debug Analyze Tools Window Help
Projects Agent Headers Sources Forms Open Documents
  Agent Agent.pro Headers agent.h formnewmap.h mainwindow.h world.h
  Sources agent.cpp formnewmap.cpp main.cpp mainwindow.cpp world.cpp
  Forms
  Open Documents agent.cpp mainwindow.h world.h
  Line: 16, Col: 1

```

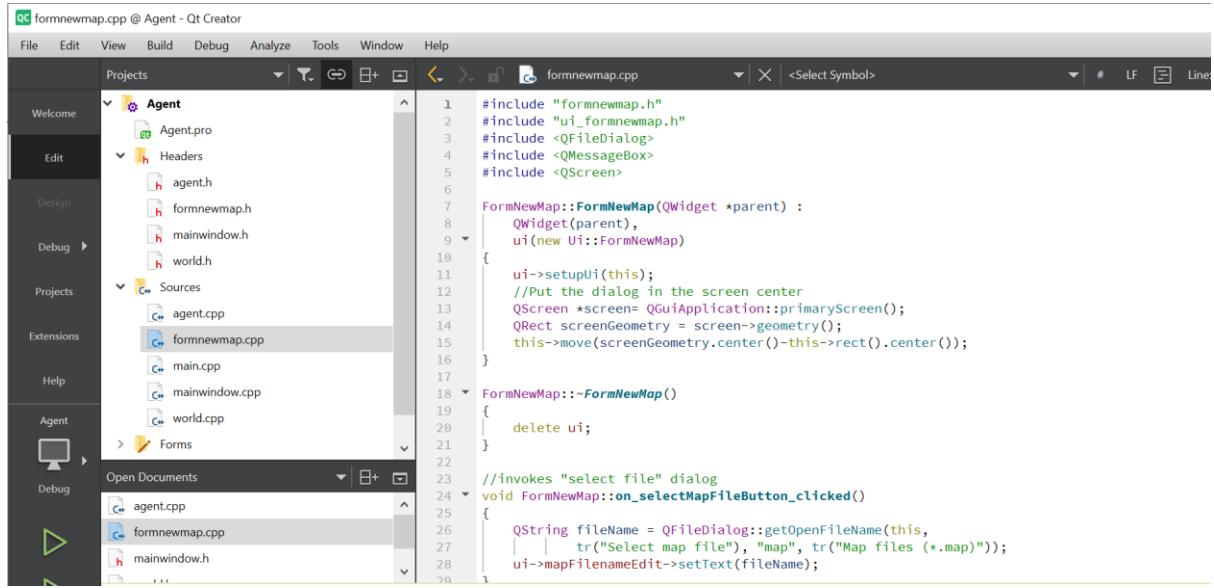
```

#include "agent.h"
#include "world.h"

Agent::actions Agent::act(bool isBumped, int dirtAmount, actions lastAction)
{
    /* Example agent moves randomly sucking as much dirt as possible */
    if(dirtAmount > 0)
    {
        return suck;
    }
    else
    {
        return (static_cast<actions> (rand() % 4));
    }
}

```

Eight source code file (formnewmap.cpp) in the Qt Agent folder Sources section that is downloaded from the given github repository.



The screenshot shows the Qt Creator interface with the project "Agent" open. The "Sources" folder contains the file "formnewmap.cpp", which is currently selected and displayed in the code editor. The code implements a dialog window for selecting a map file.

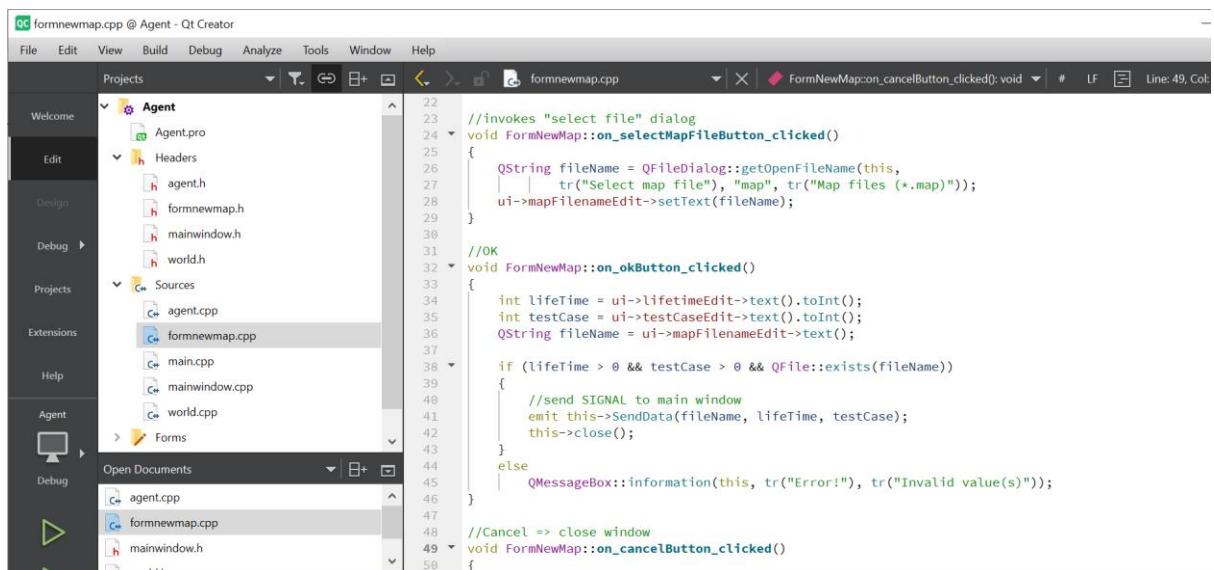
```
#include "formnewmap.h"
#include "ui_formnewmap.h"
#include <QFileDialog>
#include < QMessageBox>
#include <QScreen>

FormNewMap::FormNewMap(QWidget *parent) :
    QWidget(parent),
    ui(new Ui::FormNewMap)
{
    ui->setupUi(this);
    //Put the dialog in the screen center
    QScreen *screen= QGuiApplication::primaryScreen();
    QRect screenGeometry = screen->geometry();
    this->move(screenGeometry.center()-this->rect().center());
}

FormNewMap::~FormNewMap()
{
    delete ui;
}

//invokes "select file" dialog
void FormNewMap::on_selectMapFileButton_clicked()
{
    QString fileName = QFileDialog::getOpenFileName(this,
                                                    tr("Select map file"), "map", tr("Map files (*.map)"));
    ui->mapFilenameEdit->setText(fileName);
}


```



The screenshot shows the same Qt Creator interface, but the code editor now displays the implementation of the `on_selectMapFileButton_clicked()` slot. It shows the logic for opening a file dialog and setting the selected file name in a line edit.

```
//invokes "select file" dialog
void FormNewMap::on_selectMapFileButton_clicked()
{
    QString fileName = QFileDialog::getOpenFileName(this,
                                                    tr("Select map file"), "map", tr("Map files (*.map)"));
    ui->mapFilenameEdit->setText(fileName);
}

//OK
void FormNewMap::on_okButton_clicked()
{
    int lifeTime = ui->lifetimeEdit->text().toInt();
    int testCase = ui->testCaseEdit->text().toInt();
    QString fileName = ui->mapFilenameEdit->text();

    if (lifeTime > 0 && testCase > 0 && QFile::exists(fileName))
    {
        //send SIGNAL to main window
        emit this->SendData(fileName, lifeTime, testCase);
        this->close();
    }
    else
        QMessageBox::information(this, tr("Error!"), tr("Invalid value(s)"));

    //Cancel => close window
}

void FormNewMap::on_cancelButton_clicked()
{
```

```
41     emit this->SendData(fileName, lifeTime, testCase);
42     this->close();
43 }
44 else
45 {
46     QMessageBox::information(this, tr("Error!"), tr("Invalid value(s)"));
47 }
48 //Cancel => close window
49 void FormNewMap::on_cancelButton_clicked()
50 {
51     this->close();
52 }
53
54 void FormNewMap::on_lifetimeEdit_returnPressed()
55 {
56     on_okButton_clicked();
57 }
58
59 void FormNewMap::on_mapFilenameEdit_returnPressed()
60 {
61     on_selectMapFileButton_clicked();
62 }
63
64 void FormNewMap::on_TestCaseEdit_returnPressed()
65 {
66     on_okButton_clicked();
67 }
68
```

Nine source code file(main.cpp) in the Qt Agent folder sources section downloaded from the given github repository.

```
1 #include <QtWidgets/QApplication>
2 #include "mainwindow.h"
3
4 int main(int argc, char *argv[])
5 {
6     QApplication a(argc, argv);
7     MainWindow w;
8     w.show();
9
10    return a.exec();
11}
12
```

Tenth source code file(mainwindow.cpp) in the Qt Agent folder sources section downloaded from the given github repository.

mainwindow.cpp @ Agent - Qt Creator

```
File Edit View Build Debug Analyze Tools Window Help
```

Projects Agent Headers Sources Forms Open Documents

```
1 #include "mainwindow.h"
2 #include "ui_mainwindow.h"
3 #include "formnewmap.h"
4 #include <QMessageBox>
5 #include <QTimer>
6
7 MainWindow::MainWindow(QWidget *parent) :
8     QMainWindow(parent),
9     ui(new Ui::MainWindow)
10 {
11     ui->setupUi(this);
12
13     scene = new QGraphicsScene();
14     ui->graphicsView->setScene(scene);
15 }
16
17 MainWindow::~MainWindow()
18 {
19     delete ui;
20 }
21
22 //Quit
23 void MainWindow::on_quitButton_clicked()
24 {
25     this->close();
26 }
27
28 //Select map
29 void MainWindow::on_selectMapButton_clicked()
```

mainwindow.cpp @ Agent - Qt Creator

```
File Edit View Build Debug Analyze Tools Window Help
```

Projects Agent Headers Sources Forms Open Documents

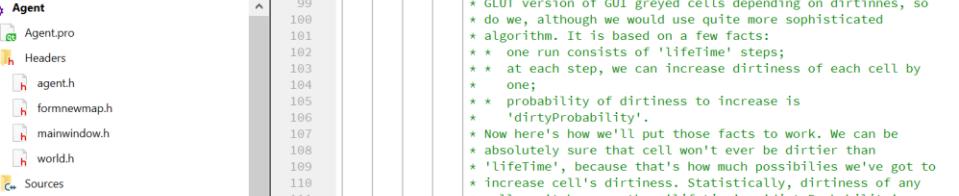
```
26
27
28 //Select map
29 void MainWindow::on_selectMapButton_clicked()
30 {
31     FormNewMap* form = new FormNewMap(0);
32
33     connect(form, SIGNAL(SendData(QString, int, int)), this, SLOT(onNewMapData(QString, int, int)));
34
35     form->show();
36 }
37
38 //SLOT, receives data from "select map" form
39 void MainWindow::onNewMapData(QString filename, int lifetime, int testcase)
40 {
41     fileName = filename;
42     lifeTime = lifetime;
43     testCase = testcase;
44
45     if (LoadMap())
46     {
47         DrawMap();
48
49         QRectF sceneRect = ui->graphicsView->sceneRect();
50         ui->graphicsView->fitInView(sceneRect, Qt::KeepAspectRatio);
51
52         ui->stepsEdit->setText(QString("%1").arg(lifeTime));
53     }
54 }
55
56
57
58
59 bool MainWindow::LoadMap()
60 {
61     /* FIXME: maybe toUtf8() is a better choice */
62     w = new World(fileName.toLocal8Bit().constData());
63
64     if (w->getErrorMessage().isEmpty())
65     {
66         currentRun = 1;
67         totalDirtyDegree = 0;
68         totalConsumedEnergy = 0;
69
70         ui->doOneStepButton->setEnabled(true);
71         ui->doOneRunButton->setEnabled(true);
72         ui->doAllRunsButton->setEnabled(true);
73         ui->displayButton->setEnabled(true);
74
75     }
76 }
```

mainwindow.cpp @ Agent - Qt Creator

```
File Edit View Build Debug Analyze Tools Window Help
```

Projects Agent Headers Sources Forms Open Documents

```
48
49
50
51
52
53
54
55
56
57
58
59 bool MainWindow::LoadMap()
60 {
61     /* FIXME: maybe toUtf8() is a better choice */
62     w = new World(fileName.toLocal8Bit().constData());
63
64     if (w->getErrorMessage().isEmpty())
65     {
66         currentRun = 1;
67         totalDirtyDegree = 0;
68         totalConsumedEnergy = 0;
69
70         ui->doOneStepButton->setEnabled(true);
71         ui->doOneRunButton->setEnabled(true);
72         ui->doAllRunsButton->setEnabled(true);
73         ui->displayButton->setEnabled(true);
74
75     }
76 }
```



```
mainwindow.cpp @ Agent - Qt Creator
File Edit View Build Debug Analyze Tools Window Help

Projects Projects mainWindow.cpp MainWindow:DrawMap(): void Line: 126, Col: 45
Welcome Agent 99
Edit Headers 100
Design agent.h 101
Debug formnewmap.h 102
Projectsmainwindow.h 103
Extensions world.h 104
Help Sources 105
Agent agent.cpp 106
Forms formnewmap.cpp 107
mainwindow.cpp 108
main.cpp 109
world.cpp 110
mainwindow.cpp 111
agent.cpp 112
formnewmap.h 113
mainwindow.h 114
world.h 115
mainwindow.cpp 116
main.cpp 117
world.cpp 118
mainwindow.cpp 119
agent.h 120
formnewmap.h 121
mainwindow.h 122
world.h 123
mainwindow.cpp 124
main.cpp 125
mainwindow.cpp 126
mainwindow.cpp 127

* GLUT version of GUI greyed cells depending on dirtiness, so
* do we, although we would use quite more sophisticated
* algorithm. It is based on a few facts:
* * one run consists of 'lifeTime' steps;
* * at each step, we can increase dirtiness of each cell by
* one;
* * probability of dirtiness to increase is
* 'dirtyProbability'.
* Now here's how we'll put those facts to work. We can be
* absolutely sure that cell won't ever be dirtier than
* 'lifeTime', because that's how much possibilities we've got to
* increase cell's dirtiness. Statistically, dirtiness of any
* cell can't be more than 'lifetime' * 'dirtyProbability'.
*
* Given all that, the following algorithm follows naturally.
*
* Note, though, that sophisticated algorithm makes sense only
* for big (more than 255) lifetimes - on short ones it strives
* to make cells bi-colour (or tri-colour if you're lucky :).
* Because of that, we provide a fallback.
*/
int dirt = w->getWorld()->at(i)->at(j);
float dirtProb = w->getDirtyProbability();
int dirtColor = 255;
if(lifetime > 255)
    if(dirt / dirtProb < lifetime)
    {
        double dirtiness = lifetime - dirt / dirtProb;
        dirtColor *= dirtiness / lifetime;
    }

```



The screenshot shows the Qt Creator interface with the following details:

- File Menu:** File, Edit, View, Build, Debug, Analyze, Tools, Window, Help.
- Projects View:** Shows the project structure:
 - Agent (selected)
 - Headers
 - agent.h
 - formnewmap.h
 - mainwindow.h
 - world.h
 - Sources
 - agent.cpp
 - formnewmap.cpp
 - main.cpp
 - mainwindow.cpp (selected)
 - world.cpp
 - Forms
- Open Documents:** Shows the files currently open:
 - formnewmap.cpp
 - main.cpp
 - mainwindow.cpp
- Code Editor:** Displays the C++ code for `mainwindow.cpp`. The code handles dirtiness calculations and drawing rectangles on a scene.

mainwindow.cpp @ Agent - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

Projects Agent

Welcome Edit Design Debug Projects Extensions Help Agent Debug

mainwindow.cpp

```
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160 pen->setColor(QColor(dirtColor, dirtColor, dirtColor));
brush->setColor(QColor(dirtColor, dirtColor, dirtColor));
}
scene->addRect(i * RECTANGLE_SIZE, j * RECTANGLE_SIZE,
| RECTANGLE_SIZE, RECTANGLE_SIZE, *pen, *brush);
delete pen;
delete brush;
}

/* clear() doesn't make scene shrink
 * That led to the problems in case when small map was loaded after large
 * one: scene didn't shrink, thus scaling didn't work properly */
scene->setSceneRect(scene->itemsBoundingRect());
```

mainwindow.cpp @ Agent - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

Projects Agent

Welcome Edit Design Debug Projects Extensions Help Agent Debug

mainwindow.cpp

```
161
162
163
164
165
166
167
168
169 QVector<QPoint> triangle;
QColor color(0, 255, 0);
if (w->isJustBumped())
| color.setRgb(255, 0, 0);

Agent::actions action = w->getLastAgentAction();
QPen pen(color);
QBrush brush(color);
int posX = w->getAgentPosX() * RECTANGLE_SIZE,
| posY = w->getAgentPosY() * RECTANGLE_SIZE;
switch (action)
```

mainwindow.cpp @ Agent - Qt Creator

File Edit View Build Debug Analyze Tools Window Help

Projects Agent

Welcome Edit Design Debug Projects Extensions Help Agent Debug

mainwindow.cpp

```
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
break;

case Agent::idle:
| /* Don't need to do anything - all the colors are set up already */
break;

case Agent::suck:
| /* box should be yellow */
pen.setColor(QColor(255, 255, 0));
brush.setColor(QColor(255, 255, 0));
break;

case Agent::moveUp:
triangle.push_back(QPoint(posX + RECTANGLE_SIZE / 2,
| posY + RECTANGLE_SIZE / 10));
triangle.push_back(QPoint(posX + RECTANGLE_SIZE / 10,
| posY + RECTANGLE_SIZE * 9 / 10));
triangle.push_back(QPoint(posX + RECTANGLE_SIZE * 9 / 10,
| posY + RECTANGLE_SIZE * 9 / 10));
break;

case Agent::moveDown:
triangle.push_back(QPoint(posX + RECTANGLE_SIZE / 2,
| posY + RECTANGLE_SIZE * 9 / 10));
triangle.push_back(QPoint(posX + RECTANGLE_SIZE / 10,
| posY + RECTANGLE_SIZE / 10));
triangle.push_back(QPoint(posX + RECTANGLE_SIZE * 9 / 10,
| posY + RECTANGLE_SIZE / 10));
break;

case Agent::moveLeft:
triangle.push_back(QPoint(posX + RECTANGLE_SIZE / 10,
| posY + RECTANGLE_SIZE / 2));
triangle.push_back(QPoint(posX + RECTANGLE_SIZE * 9 / 10,
| posY + RECTANGLE_SIZE / 10));
triangle.push_back(QPoint(posX + RECTANGLE_SIZE * 9 / 10,
| posY + RECTANGLE_SIZE * 9 / 10));
break;

case Agent::moveRight:
triangle.push_back(QPoint(posX + RECTANGLE_SIZE * 9 / 10,
| posY + RECTANGLE_SIZE / 2));
triangle.push_back(QPoint(posX + RECTANGLE_SIZE / 10,
| posY + RECTANGLE_SIZE / 10));
triangle.push_back(QPoint(posX + RECTANGLE_SIZE / 10,
| posY + RECTANGLE_SIZE * 9 / 10));
break;
```

```
mainwindow.cpp @ Agent - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects Projects mainWindow.cpp MainWindow::refreshStats() void
Welcome Agent Agent.pro
Edit Headers agent.h
Design formnewmap.h
Debug mainWindow.h
World world.h
Projects Sources agent.cpp
Extensions formnewmap.cpp
Help main.cpp
Agent mainWindow.cpp
Forms world.cpp
Open Documents formnewmap.cpp
main.cpp
mainWindow.cpp
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
```

```
mainwindow.cpp @ Agent - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects mainWindow.cpp MainWindow::RefreshStats(): void
Welcome Agent mainWindow.cpp Line: 263, Col: 41
Edit Headers
Design mainWindow.h
Debug world.h
Projects Sources
agent.cpp
formnewmap.cpp
main.cpp
mainwindow.cpp
world.cpp
Help Forms
Agent mainWindow.cpp
Debug
Forms
Open Documents formnewmap.cpp
main.cpp
mainwindow.cpp
```

```
if (w != NULL && w->getErrorMessage().isEmpty())
{
    QString stats = QString("<table width=\"100%\"><tr>") +
        tr("<td>Run %1, time step %2</td>") +
        tr("<td>Completed runs: %3</td></tr><tr>") +
        tr("<td>Action: %4</td>") +
        tr("<td>Total dirty degree: %5</td></tr><tr>") +
        tr("<td>Dirty degree: %6</td>") +
        tr("<td>Total consumed energy: %7</td></tr><tr>") +
        tr("<td>Consumed energy: %8</td>") +
        tr("<td>Average dirty degree: %9</td></tr><tr>") +
        tr("<td>Average consumed energy: %10</td>") +
        QString("</tr></table>");

QString currentAction;
switch (w->getLastAgentAction())
{
    case Agent::moveUp:
        currentAction = "Up";
        break;
    case Agent::moveDown:
        currentAction = "Down";
        break;
    case Agent::moveLeft:
        currentAction = "Left";
        break;
    case Agent::moveRight:
        currentAction = "Right";
        break;
}
```

The screenshot shows the Qt Creator IDE interface. The top menu bar includes File, Edit, View, Build, Debug, Analyze, Tools, Window, and Help. The left sidebar has sections for Welcome, Edit, Design, Debug, Projects, Extensions, Help, Agent, and Debug. The Projects view shows a project named "Agent" with sub-folders "Headers" containing "agent.h", "formnewmap.h", "mainwindow.h", and "world.h"; and "Sources" containing "agent.cpp", "formnewmap.cpp", "main.cpp", and "mainwindow.cpp". The main area displays the code for "mainwindow.cpp". The code is a switch statement handling various agent actions:

```
case Agent::moveUp:  
| currentAction = "Up";  
break;  
case Agent::moveDown:  
| currentAction = "Down";  
break;  
case Agent::moveLeft:  
| currentAction = "Left";  
break;  
case Agent::moveRight:  
| currentAction = "Right";  
break;  
case Agent::suck:  
| currentAction = "Suck";  
break;  
default:  
| currentAction = "Idle";  
break;  
}  
  
currentAction.append((w->isJustBumped())?(QString(" Bump")):(NULL));  
  
int completedRuns = currentRun;  
if(w->getCurrentTime() < lifeTime) {  
| completedRuns--;  
}  
  
double avgDirtyDegree = 0.0, avgConsumedEnergy = 0.0;  
if(completedRuns != 0) {
```

```

mainwindow.cpp @ Agent - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects Projects mainWindow.cpp ▾ X MainWindow::refreshStats() void Line: 302, Col: 2
Welcome Agent
Edit Headers
    agent.h
    formnewmap.h
    mainwindow.h
    world.h
Sources Sources
    agent.cpp
    formnewmap.cpp
    main.cpp
    mainWindow.cpp
    world.cpp
Forms Forms
    mainWindow.cpp
    formnewmap.cpp
    main.cpp
    mainWindow.cpp
    world.cpp
Open Documents Open Documents
    mainWindow.cpp
    formnewmap.cpp
    main.cpp
    mainWindow.cpp
    world.cpp
    mainWindow.cpp
    mainWindow.h
    world.cpp
    ...

```

```

int completedRuns = currentRun;
if(w->getCurrentTime() < lifeTime) {
    completedRuns--;
}

double avgDirtyDegree = 0.0, avgConsumedEnergy = 0.0;
if(completedRuns != 0) {
    avgDirtyDegree = totalDirtyDegree / completedRuns;
    avgConsumedEnergy = totalConsumedEnergy / completedRuns;
}
ui->statsLabel->setText(stats.
    arg(currentRun).
    arg(w->getCurrentTime()).
    arg(completedRuns).
    arg(currentAction).
    arg(totalDirtyDegree, 10, 'f', 0).
    arg(w->getDirtyDegree()).
    arg(totalConsumedEnergy, 10, 'f', 0).
    arg(w->getConsumedEnergy()).
    arg(avgDirtyDegree, 7, 'f', 3).
    arg(avgConsumedEnergy, 7, 'f', 3));
} else
{
    ui->statsLabel->setText("Map not loaded...");
}

```

Eleventh source code file(world.cpp) in the Qt Agent folder sources section downloaded from the given github repository.

```

world.cpp @ Agent - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects Projects world.cpp ▾ X <Select Symbol> Line: 1, Col: 1
Welcome Agent
Edit Headers
    agent.h
    formnewmap.h
    mainwindow.h
    world.h
Sources Sources
    agent.cpp
    formnewmap.cpp
    main.cpp
    mainWindow.cpp
    world.cpp
Forms Forms
    mainWindow.cpp
    formnewmap.cpp
    main.cpp
    mainWindow.cpp
    world.cpp
    mainWindow.h
    world.cpp
    ...

```

```

1 #include "world.h"
2
3 #ifndef _MSC_VER
4 /* Windows doesn't have rand() so provide an implementation from FreeBSD libc
5 * which is distributed under the following terms:
6 *
7 * Copyright (c) 1990, 1993
8 *      The Regents of the University of California. All rights reserved.
9 *
10 * Redistribution and use in source and binary forms, with or without
11 * modification, are permitted provided that the following conditions
12 * are met:
13 * 1. Redistributions of source code must retain the above copyright
14 *    notice, this list of conditions and the following disclaimer.
15 * 2. Redistributions in binary form must reproduce the above copyright
16 *    notice, this list of conditions and the following disclaimer in the
17 *    documentation and/or other materials provided with the distribution.
18 * 4. Neither the name of the University nor the names of its contributors
19 *    may be used to endorse or promote products derived from this software
20 *    without specific prior written permission.
21 *
22 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
23 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
24 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
25 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
26 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
27 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
28 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
29 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY WHETHER IN CONTRACT, STRICT
30 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
31 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
32 * SUCH DAMAGE.
33 */

```

world.cpp @ Agent - Qt Creator

```

File Edit View Build Debug Analyze Tools Window Help
Projects world.cpp <Select Symbol>
Line: 49, Col: 1
Welcome Agent
Edit Headers
Sources
Forms
Open Documents
mainwindow.cpp
mainwindow.h
world.cpp
...
22 * THIS SOFTWARE IS PROVIDED BY THE REGENTS AND CONTRIBUTORS ``AS IS'' AND
23 * ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
24 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
25 * ARE DISCLAIMED. IN NO EVENT SHALL THE REGENTS OR CONTRIBUTORS BE LIABLE
26 * FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL
27 * DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS
28 * OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
29 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT
30 * LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY
31 * OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
32 * SUCH DAMAGE.
33 *
34 * Posix rand function added May 1999 by Wes Peters <wes@softweyr.com>.
35 */
36
37 static int do_rand(unsigned long *ctx)
38 {
39     /*
40     * Compute x = (7^5 * x) mod (2^31 - 1)
41     * without overflowing 31 bits:
42     *      (2^31 - 1) = 127773 * (7^5) + 2836
43     * From "Random number generators: good ones are hard to find",
44     * Park and Miller, Communications of the ACM, vol. 31, no. 10,
45     * October 1988, p. 1195.
46     */
47     long hi, lo, x;
48
49     /* Can't be initialized with 0, so use another value. */
50     if (*ctx == 0)
51         /*ctx = 123459876;
52         hi = *ctx / 127773;
53         lo = *ctx % 127773;
54         x = 16807 * lo - 2836 * hi;
55         if (x < 0)
56             x += 0xffffffff;
57         return ((*ctx = x) % ((unsigned long)RAND_MAX + 1));
58     */
59
60     int rand(unsigned int *ctx)
61     {
62         unsigned long val = (unsigned long) *ctx;
63         int r = do_rand(&val);
64
65         *ctx = (unsigned int) val;
66         return (r);
67     }
68
69 #endif
70 World::World(string)
71 {
72     /* we can be sure filename is valid - GUI checked it */
73
74     /* world doesn't even exist yet, so agent couldn't possibly bump into
75     * something */
76     justBumped = false;
77
78     /* ...and of course time doesn't tick yet */
79     currentTime = 0;
80
81     /* read map file and initialise world */
82     string line;
83     ifstream mapFile(filename.c_str());
84     getline(mapFile, line);
85     while (line.substr(0,2).compare("//") == 0 || line.empty()
86           || line.find_first_not_of(' ') == string::npos)
87     {
88         /* First lines in a file may be:
89         * * C++-style comments;
90         * * just empty lines (may contain spaces).
91         * We'll just ignore this stuff. */
92         getline(mapFile, line);

```

world.cpp @ Agent - Qt Creator

```

File Edit View Build Debug Analyze Tools Window Help
Projects world.cpp <Select Symbol>
Line: 66, Col: 1
Welcome Agent
Edit Headers
Sources
Forms
Open Documents
mainwindow.cpp
mainwindow.h
world.cpp
...
39 /*
40 * Compute x = (7^5 * x) mod (2^31 - 1)
41 * without overflowing 31 bits:
42 *      (2^31 - 1) = 127773 * (7^5) + 2836
43 * From "Random number generators: good ones are hard to find",
44 * Park and Miller, Communications of the ACM, vol. 31, no. 10,
45 * October 1988, p. 1195.
46 */
47 long hi, lo, x;
48
49 /* Can't be initialized with 0, so use another value. */
50 if (*ctx == 0)
51     /*ctx = 123459876;
52     hi = *ctx / 127773;
53     lo = *ctx % 127773;
54     x = 16807 * lo - 2836 * hi;
55     if (x < 0)
56         x += 0xffffffff;
57     return ((*ctx = x) % ((unsigned long)RAND_MAX + 1));
58 */
59
60 int rand(unsigned int *ctx)
61 {
62     unsigned long val = (unsigned long) *ctx;
63     int r = do_rand(&val);
64
65     *ctx = (unsigned int) val;
66     return (r);
67 }
68
69 #endif
70 World::World(string)
71 {
72     /* we can be sure filename is valid - GUI checked it */
73
74     /* world doesn't even exist yet, so agent couldn't possibly bump into
75     * something */
76     justBumped = false;
77
78     /* ...and of course time doesn't tick yet */
79     currentTime = 0;
80
81     /* read map file and initialise world */
82     string line;
83     ifstream mapFile(filename.c_str());
84     getline(mapFile, line);
85     while (line.substr(0,2).compare("//") == 0 || line.empty()
86           || line.find_first_not_of(' ') == string::npos)
87     {
88         /* First lines in a file may be:
89         * * C++-style comments;
90         * * just empty lines (may contain spaces).
91         * We'll just ignore this stuff. */
92         getline(mapFile, line);

```

world.cpp @ Agent - Qt Creator

```

File Edit View Build Debug Analyze Tools Window Help
Projects world.cpp <Select Symbol>
Line: 91, Col: 1
Welcome Agent
Edit Headers
Sources
Forms
Open Documents
mainwindow.cpp
mainwindow.h
world.cpp
...
64     *ctx = (unsigned int) val;
65     return (r);
66 }
67
68 #endif
69
70 World::World(string filename)
71 {
72     /* we can be sure filename is valid - GUI checked it */
73
74     /* world doesn't even exist yet, so agent couldn't possibly bump into
75     * something */
76     justBumped = false;
77
78     /* ...and of course time doesn't tick yet */
79     currentTime = 0;
80
81     /* read map file and initialise world */
82     string line;
83     ifstream mapFile(filename.c_str());
84     getline(mapFile, line);
85     while (line.substr(0,2).compare("//") == 0 || line.empty()
86           || line.find_first_not_of(' ') == string::npos)
87     {
88         /* First lines in a file may be:
89         * * C++-style comments;
90         * * just empty lines (may contain spaces).
91         * We'll just ignore this stuff. */
92         getline(mapFile, line);

```

world.cpp @ Agent - Qt Creator

```

File Edit View Build Debug Analyze Tools Window Help
Projects world.cpp < > X World:World(string)
File Edit View Build Debug Analyze Tools Window Help
Welcome Projects world.cpp < > X World:World(string)
Edit Agent Headers world.cpp < > X Line: 110, Col: 1
Design Sources world.cpp < > X Line: 110, Col: 1
Debug
Projects world.cpp < > X Line: 110, Col: 1
Extensions
Help
Agent
  Agent.pro
  Headers
    agent.h
    formnewmap.h
    mainwindow.h
    world.h
  Sources
    agent.cpp
    formnewmap.cpp
    main.cpp
    mainwindow.cpp
    world.cpp
Forms
  mainwindow.cpp
  mainwindow.h
  world.cpp
Open Documents world.cpp < > X Line: 110, Col: 1
mainwindow.cpp
mainwindow.h
world.cpp

```

```

83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111

getline(mapFile, line);
while (line.substr(0,2).compare("//") == 0 || line.empty()
| | || line.find_first_not_of(' ') == string::npos)
{
    /* First lines in a file may be:
     * * C++-style comments;
     * * just empty lines (may contain spaces).
     * We'll just ignore this stuff. */
    getline(mapFile, line);

    if(mapFile.eof())
    {
        // FIXME: display error message here
        break;
    }
}

/* Invariant: comment block is passed, 'line' should contain initial agent
 * position, dirty probability and random seed now. */

istringstream parameters(line);
/* Throw an exception if:
 * (eofbit) end of file reached
 * (badbit) some other error occurred */
parameters.exceptions(ifstream::eofbit | ifstream::badbit);
try
{
    parameters >> agentPosX >> agentPosY >> dirtyProbability;
}
/* We're really only interested in catching ios_base::failure here, but GCC

```

world.cpp @ Agent - Qt Creator

```

File Edit View Build Debug Analyze Tools Window Help
Projects world.cpp < > X World:World(string)
File Edit View Build Debug Analyze Tools Window Help
Welcome Projects world.cpp < > X World:World(string)
Edit Agent Headers world.cpp < > X Line: 135, Col: 1
Design Sources world.cpp < > X Line: 135, Col: 1
Debug
Projects world.cpp < > X Line: 135, Col: 1
Extensions
Help
Agent
  Agent.pro
  Headers
    agent.h
    formnewmap.h
    mainwindow.h
    world.h
  Sources
    agent.cpp
    formnewmap.cpp
    main.cpp
    mainwindow.cpp
    world.cpp
Forms
  mainwindow.cpp
  mainwindow.h
  world.cpp
Open Documents world.cpp < > X Line: 135, Col: 1
mainwindow.cpp
mainwindow.h
world.cpp

```

```

188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
8010
8011
8012
8013
8014
8015
8016
8017
8018
8019
8020
8021
8022
8023
8024
8025
8026
8027
8028
8029
8030
8031
8032
8033
8034
8035
8036
8037
8038
8039
8040
8041
8042
8043
8044
8045
8046
8047
8048
8049
8050
8051
8052
8053
8054
8055
8056
8057
8058
8059
8060
8061
8062
8063
8064
8065
8066
8067
8068
8069
8070
8071
8072
8073
8074
8075
8076
8077
8078
8079
8080
8081
8082
8083
8084
8085
8086
8087
8088
8089
8090
8091
8092
8093
8094
8095
8096
8097
8098
8099
80100
80101
80102
80103
80104
80105
80106
80107
80108
80109
80110
80111
80112
80113
80114
80115
80116
80117
80118
80119
80120
80121
80122
80123
80124
80125
80126
80127
80128
80129
80130
80131
80132
80133
80134
80135
80136
80137
80138
80139
80140
80141
80142
80143
80144
80145
80146
80147
80148
80149
80150
80151
80152
80153
80154
80155
80156
80157
80158
80159
80160
80161
80162
80163
80164
80165
80166
80167
80168
80169
80170
80171
80172
80173
80174
80175
80176
80177
80178
80179
80180
80181
80182
80183
80184
80185
80186
80187
80188
80189
80190
80191
80192
80193
80194
80195
80196
80197
80198
80199
80200
80201
80202
80203
80204
80205
80206
80207
80208
80209
80210
80211
80212
80213
80214
80215
80216
80217
80218
80219
80220
80221
80222
80223
80224
80225
80226
80227
80228
80229
80230
80231
80232
80233
80234
80235
80236
80237
80238
80239
80240
80241
80242
80243
80244
80245
80246
80247
80248
80249
80250
80251
80252
80253
80254
80255
80256
80257
80258
80259
80260
80261
80262
80263
80264
80265
80266
80267
80268
80269
80270
80271
80272
80273
80274
80275
80276
80277
80278
80279
80280
80281
80282
80283
80284
80285
80286
80287
80288
80289
80290
80291
80292
80293
80294
80295
80296
80297
80298
80299
80290
80291
80292
80293
80294
80295
80296
80297
80298
80299
80300
80301
80302
80303
80304
80305
80306
80307
80308
80309
80310
80311
80312
80313
80314
80315
80316
80317
80318
80319
80320
80321
80322
80323
80324
80325
80326
80327
80328
80329
80330
80331
80332
80333
80334
80335
80336
80337
80338
80339
80330
80331
80332
80333
80334
80335
80336
80337
80338
80339
80340
80341
80342
80343
80344
80345
80346
80347
80348
80349
80340
80341
80342
80343
80344
80345
80346
80347
80348
80349
80350
80351
80352
80353
80354
80355
80356
80357
80358
80359
80350
80351
80352
80353
80354
80355
80356
80357
80358
80359
80360
80361
80362
80363
80364
80365
80366
80367
80368
80369
80360
80361
80362
80363
80364
80365
80366
80367
80368
80369
80370
80371
80372
80373
80374
80375
80376
80377
80378
80379
80370
80371
80372
80373
80374
80375
80376
80377
80378
80379
80380
80381
80382
80383
80384
80385
80386
80387
80388
80389
80380
80381
80382
80383
80384
80385
80386
80387
80388
80389
80390
80391
80392
80393
80394
80395
80396
80397
80398
80399
80390
80391
80392
80393
80394
80395
80396
80397
80398
80399
80400
80401
80402
80403
80404
80405
80406
80407
80408
80409
80400
80401
80402
80403
80404
80405
80406
80407
80408
80409
80410
80411
80412
80413
80414
80415
80416
80417
80418
80419
80420
80421
80422
80423
80424
80425
80426
80427
80428
80429
80430
80431
80432
80433
80434
80435
80436
80437
80438
80439
80440
80441
80442
80443
80444
80445
80446
80447
80448
80449
80450
80451
80452
80453
80454
80455
80456
80457
80458
80459
80460
80461
80462
80463
80464
80465
80466
80467
80468
80469
80470
80471
80472
80473
80474
80475
80476
80477
80478
80479
80480
80481
80482
80483
80484
80485
80486
80487
80488
80489
80490
80491
80492
80493
80494
80495
80496
80497
80498
80499
80500
80501
80502
80503
80504
80505
80506
80507
80508
80509
80510
80511
80512
80513
80514
80515
80516
80517
80518
80519
80520
80521
80522
80523
80524
80525
80526
80527
80528
80529
80530
80531
80532
80533
80534
80535
80536
80537
80538
80539
80540
80541
80542
80543
80544
80545
80546
80547
80548
80549
80550
80551
80552
80553
80554
80555
80556
80557
80558
80559
80560
80561
80562
80563
80564
80565
80566
80567
80568
80569
80570
80571
80572
80573
80574
80575
80576
80577
80578
80579
80580
80581
80582
80583
80584
80585
80586
80587
80588
80589
80590
80591
80592
80593
80594
80595
80596
80597
80598
80599
80600
80601
80602
80603
80604
80605
80606
80607
80608
80609
80610
80611
80612
80613
80614
80615
80616
80617
80618
80619
80620
80621
80622
80623
80624
80625
80626
80627
80628
80629
80630
80631
80632
80633
80634
80635
80636
80637
80638
80639
80640
80641
80642
80643
80644
80645
80646
80647
80648
80649
80650
80651
80652
80653
80654
80655
80656
80657
80658
80659
80660
80661
80662
80663
80664
80665
80666
80667
80668
80669
80670
80671
80672
80673
80674
80675
80676
80677
80678
80679
80680
80681
80682
80683
80684
80685
80686
80687
80688
80689
80690
80691
80692
80693
80694
80695
80696
80697
80698
80699
80700
80701
80702
80703
80704
80705
80706
80707
80708
80709
80710
80711
80712
80713
80714
80715
80716
80717
80718
80719
80720
80721
80722
80723
80724
80725
80726
80727
80728
80729
80730
80731
80732
80733
80734
80735
80736
80737
80738
80739
80740
80741
80742
80743
80744
80745
80746
80747
80748
80749
80750
80751
80752
80753
80754
80755
80756
80757
80758
80759
80760
80761
80762
80763
80764
80765
80766
80767
80768
80769
80770
80771
80772
80773
80774
80775
80776
80777
80778
80779
80780
80781
80782
80783
80784
80785
80786
80787
80788
80789
80790
80791
80792
80793
80794
80795
80796
80797
80798
80799
80800
80801
80802
80803
80804
80805
80806
80807
80808
80809
808010
808011
808012
808013
808014
808015
808016
808017
808018
808019
808020
808021
808022
808023
808024
808025
808026
808027
808028
808029
808030
808031
808032
808033
808034
808035
808036
808037
808038
808039
808040
808041
808042
808043
808044
808045
808046
808047
808048
808049
808050
808051
808052
808053
808054
808055
808056
808057
808058
808059
808060
808061
808062
808063
808064
808065
808066
808067
808068
808069
808070
808071
808072
808073
808074
808075
808076
808077
808078
808079
808080
808081
808082
808083
808084
808085
808086
808087
808088
808089
808090
808091
808092
808093
808094
808095
808096
808097
808098
808099
8080100
8080101
8080102
8080103
8080104
8080105
8080106
8080107
8080108
8080109
8080110
8080111
8080112
8080113
8080114
8080115
8080116
8080117
8080118
8080119
8080120
8080121
8080122
8080123
8080124
8080125
8080126
8080127
8080128
8080129
8080130
8080131
8080132
8080133
8080134
8080135
8080136
8080137
8080138
8080139
8080140
8080141
8080142
8080143
8080144
8080145
8080146
8080147
8080148
8080149
8080150
8080151
8080152
8080153
8080154
8080155
8080156
8080157
8080158
8080159
8080160
8080161
8080162
8080163
8080164
8080165
8080166
8080167
8080168
8080169
8080170
8080171
8080172
8080173
8080174
8080175
8080176
8080177
8080178
8080179
8080180
8080181
8080182
8080183
8080184
8080185
8080186
8080187
8080188
8080189
8080190
8080191
8080192
8080193
8080194
8080195
8080196
8080197
8080198
8080199
8080200
8080201
8080202
8080203
8080204
8080205
8080206
8080207
8080208
8080209
8080210
8080211
8080212
8080213
8080214
8080215
8080216
8080217
8080218
8080219
8080220
8080221
8080222
8080223
8080224
8080225
8080226
8080227
8080228
8080229
8080220
8080221
8080222
8080223
8080224
8080225
8080226
8080227
8080228
8080229
8080230
8080231
8080232
8080233
8080234
8080235
8080236
8080237
8080238
8080239
8080230
8080231
8080232
8080233
8080234
8080235
8080236
8080237
8080238
8080239
8080240
8080241
8080242
8080243
8080244
8080245
8080246
8080247
8080248
8080249
8080240
8080241
8080242
8080243
8080244
8080245
8080246
8080247
8080248
8080249
8080250
8080251
8080252
8080253
8080254
8080255
8080256
8080257
8080258
8080259
8080250
8080251
8080252
8080253
8080254
8080255
8080256
8080257
8080258
8080259
8080260
8080261
8080262
8080263
8080264
8080265
8080266
8080267
8080268
8080269
8080260
8080261
8080262
8080263
8080264
8080265
8
```

world.cpp @ Agent - Qt Creator

```
File Edit View Build Debug Analyze Tools Window Help
```

Projects Agent

- Agent
- Headers
- Sources

world.cpp

```
try
{
    parameters >> seed;
}
catch (std::exception f)
{
    //TODO: decide keep this warning or not
    /*if(! parameters.eof())
    {
        QMessageBox::warning(parent, QTranslator::tr("Malformed map"),
        QTranslator::tr("seed wasn't the last value on the parameters line"));*/
}

/* Finally, it's time to read the map itself */
/* Initializing lineno to -1 because we didn't read first line (which must
 * have index zero) yet */
int lineno = -1;
while(! mapfile.eof())
{
    lineno++;
    getline(mapFile, line);
    istringstream map(line);
    /* Throw an exception when eof reached */
    map.exceptions(ifstream::eofbit);
    int columnno = 0;
    while(! map.eof())
    {
        char c;
        int val;
```

world.cpp @ Agent - Qt Creator

```
File Edit View Build Debug Analyze Tools Window Help
```

Projects Agent

- Agent
- Headers
- Sources

world.cpp

```
/* Throw an exception when eof reached */
map.exceptions(ifstream::eofbit);
int columnno = 0;
while(! map.eof())
{
    char c;
    int val;
    try
    {
        map >> c;
    }
    catch (std::exception f)
    {
        if(map.eof())
            break;
    };
    switch(c)
    {
        case MAP_OBSTACLE:
            val = OBSTACLE;
            break;

        case MAP_ROAD:
            val = 0;
            break;

        default:
```

world.cpp @ Agent - Qt Creator

```
File Edit View Build Debug Analyze Tools Window Help
```

Projects Agent

- Agent
- Headers
- Sources

world.cpp

```
break;

        case MAP_ROAD:
            val = 0;
            break;

        default:
            errorMessage = QTranslator::tr("Unknown char in map at line %1 : %2").arg(linen
            return;
    }
    if(lineno == 0)
    {
        /* It's the first iteration, so we need to allocate space for
         * columns */
        world.push_back(new vector<int>);
        world.back()->push_back(val);
    }
    else
    {
        /* columnno can't be equal to world_width because columnno
         * counts from zero */
        if(columnno >= world_width)
        {
            errorMessage = QTranslator::tr("Line %1 is wider than others (should be %2)").i
            return;
```



```
world.cpp @ Agent - Qt Creator
File Edit View Build Debug Analyze Tools Window Help
Projects world.cpp Line: 269, Col: 1
Welcome Agent
Edit Headers
Design
Debug
Projects
Extensions
Help
Agent
Forms
Open Documents
mainwindow.cpp
mainwindow.h
world.cpp

243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270

agent = new Agent();
}

void World::performAction(Agent::actions action)
{
    justBumped = false;

    switch(action)
    {
        case Agent::moveLeft:
            if(agentPosX == 0 || agentPosX == world_width-1
               || world[agentPosX-1]>at(agentPosY) == OBSTACLE)
            {
                justBumped = true;
            }
            else
            {
                agentPosX--;
            };
            consumedEnergy++;
            break;

        case Agent::moveRight:
            if(agentPosX == 0 || agentPosX == world_width-1
               || world[agentPosX+1]>at(agentPosY) == OBSTACLE)
            {
                justBumped = true;
            }
            else
            {
                agentPosX++;
            };
            consumedEnergy++;
            break;
    }
}
```

The screenshot shows the Qt Creator interface with the following details:

- File Menu:** File, Edit, View, Build, Debug, Analyze, Tools, Window, Help.
- Projects Tab:** Shows the project structure under "Agent".
- Editor Area:** Displays the `world.cpp` file content. The code handles agent movement and energy consumption. It includes cases for moving right, up, and down, as well as logic for bumping into obstacles and updating consumed energy.

```
consumedEnergy++;
break;

case Agent::moveRight:
    if(agentPosX == 0 || agentPosX == world_width-1
        || world[agentPosX+1]->at(agentPosY) == OBSTACLE)
    {
        justBumped = true;
    }
    else
    {
        agentPosX++;
    };
    consumedEnergy++;
    break;

case Agent::moveUp:
    if(agentPosY == 0 || agentPosY == world_height-1
        || world[agentPosX]->at(agentPosY-1) == OBSTACLE)
    {
        justBumped = true;
    }
    else
    {
        agentPosY--;
    };
    consumedEnergy++;
    break;
```

world.cpp @ Agent - Qt Creator

```

File Edit View Build Debug Analyze Tools Window Help
Projects world.cpp ▾ X | World:performAction(Agent::actions): void ▾ # LF Line: 303, Col: 1
Welcome Agent
Edit Headers
Design agent.h
Debug formnewmap.h
Projects mainwindow.h
Help world.h
Agent Sources
Forms
Open Documents world.cpp
Debug mainwindow.cpp
Forms
mainwindow.h
303
304
305
306
307
308
309
310
311
312
313
314
315
consumedEnergy++;
break;

case Agent::moveDown:
if(agentPosY == 0 || agentPosY == world_height-1
|| world[agentPosX]->at(agentPosY+1) == OBSTACLE)
{
justBumped = true;
}
else
{
agentPosY++;
};
consumedEnergy++;
break;

case Agent::suck:
if(world[agentPosX]->at(agentPosY) > 0)
{
world[agentPosX]->at(agentPosY)--;
};
consumedEnergy += 2;
break;

case Agent::idle:
/* that one is easy */
break;
}

```

world.cpp @ Agent - Qt Creator

```

File Edit View Build Debug Analyze Tools Window Help
Projects world.cpp ▾ X | World:performAction(Agent::actions): void ▾ # LF Line: 327, Col: 30
Welcome Agent
Edit Headers
Design agent.h
Debug formnewmap.h
Projects mainwindow.h
Help world.h
Agent Sources
Forms
Open Documents world.cpp
Debug mainwindow.cpp
Forms
mainwindow.h
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
consumedEnergy++;
break;

case Agent::suck:
if(world[agentPosX]->at(agentPosY) > 0)
{
world[agentPosX]->at(agentPosY)--;
};
consumedEnergy += 2;
break;

case Agent::idle:
/* that one is easy */
break;

default:
/* we can't possibly reach that state because of static compiler
 * checks, so it's safe to do nothing */
break;
}

lastAgentAction = action;

/* Run through the map and add some dirt randomly */
for(int col = 0; col < world_width; col++)
for(int row = 0; row < world_height; row++)

```

world.cpp @ Agent - Qt Creator

```

File Edit View Build Debug Analyze Tools Window Help
Projects world.cpp ▾ X | World::resetMap(): void ▾ # LF Line: 374, Col: 1
Welcome Agent
Edit Headers
Design agent.h
Debug formnewmap.h
Projects mainwindow.h
Help world.h
Agent Sources
Forms
Open Documents world.cpp
Debug mainwindow.cpp
Forms
mainwindow.h
348
349
350
351 void World::resetMap()
{
352
353 /* clean all cells */
354 for(int col = 0; col < world_width; col++)
355 for(int row = 0; row < world_height; row++)
356 if(world[col]->at(row) != OBSTACLE)
357 world[col]->at(row) = 0;

358
359 /* reset agent's position */
360 agentPosX = initialAgentPosX;
361 agentPosY = initialAgentPosY;
362
363 lastAgentAction = Agent::idle;
364
365 currentTime = 0;
366 justBumped = 0;
367 dirtyDegree = 0;
368 consumedEnergy = 0;
369
370 delete agent;
371 agent = new Agent();
372
373
374
}

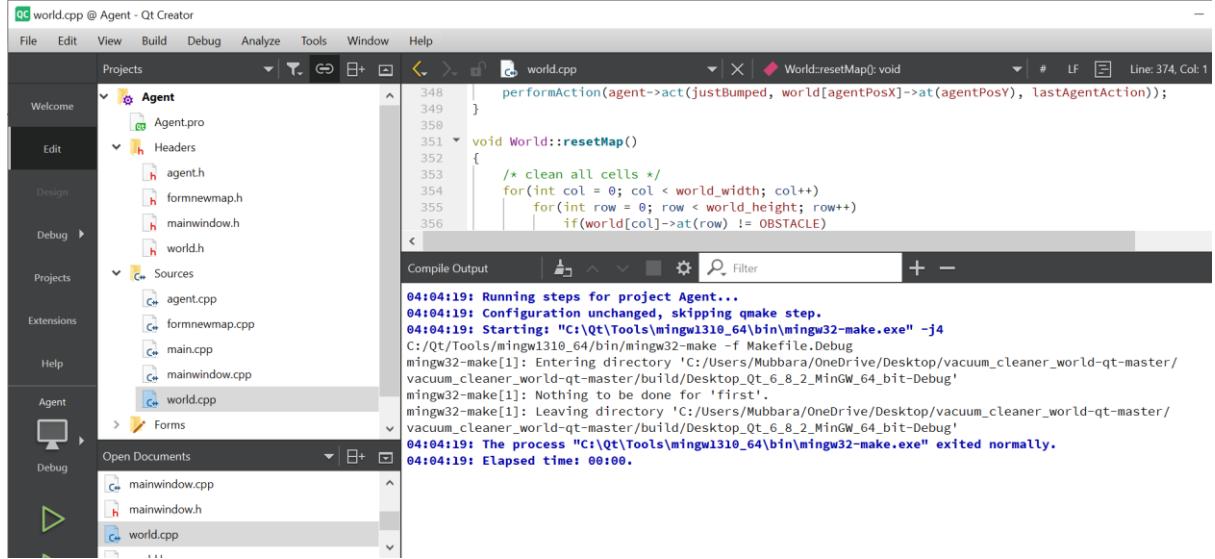
```

These are the files that is downloaded from the given github repository.

Step 5:

Building and Running the project:

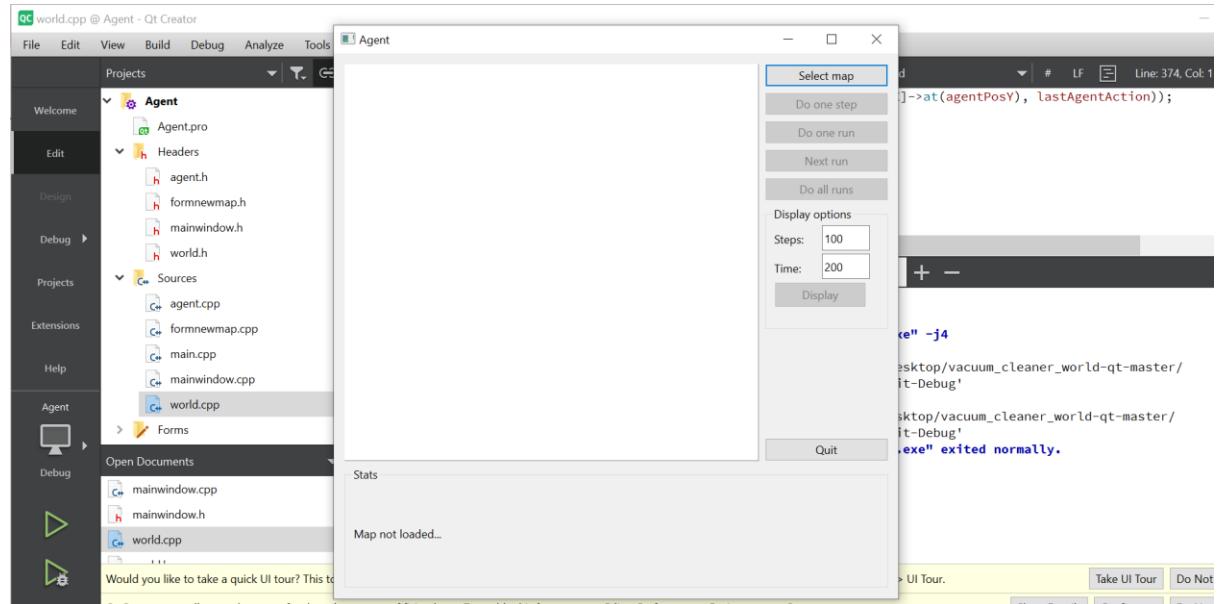
1. Click on “Configure Project” when prompted.
2. Click on Build >Build Project(ctrl + B).



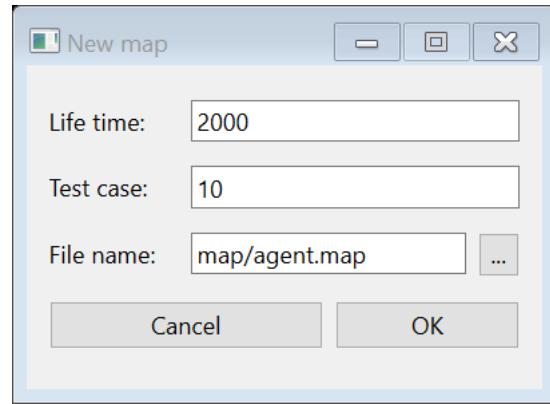
The screenshot shows the Qt Creator interface with the "world.cpp" file open in the editor. The "Projects" tab is selected, displaying the "Agent" project structure. The "Sources" section contains files like agent.cpp, formnewmap.cpp, main.cpp,mainwindow.cpp, and world.cpp. The "Compile Output" tab shows the build logs:

```
04:04:19: Running steps for project Agent...
04:04:19: Configuration unchanged, skipping qmake step.
04:04:19: Starting: "C:\Qt\Tools\mingw32_64\bin\mingw32-make.exe" -j4
C:/Qt/Tools/mingw310_64/bin/mingw32-make -f Makefile.Debug
mingw32-make[1]: Entering directory 'C:/Users/Mubbara/OneDrive/Desktop/vacuum_cleaner_world-qt-master'
vacuum_cleaner_world-qt-master/build/Desktop_Qt_6_8_2_MinGW_64_bit-Debug'
mingw32-make[1]: Nothing to be done for 'first'.
mingw32-make[1]: Leaving directory 'C:/Users/Mubbara/OneDrive/Desktop/vacuum_cleaner_world-qt-master'
vacuum_cleaner_world-qt-master/build/Desktop_Qt_6_8_2_MinGW_64_bit-Debug'
04:04:19: The process "C:\Qt\Tools\mingw310_64\bin\mingw32-make.exe" exited normally.
04:04:19: Elapsed time: 00:00.
```

3. The build is successful, click on Run(Ctrl + R)

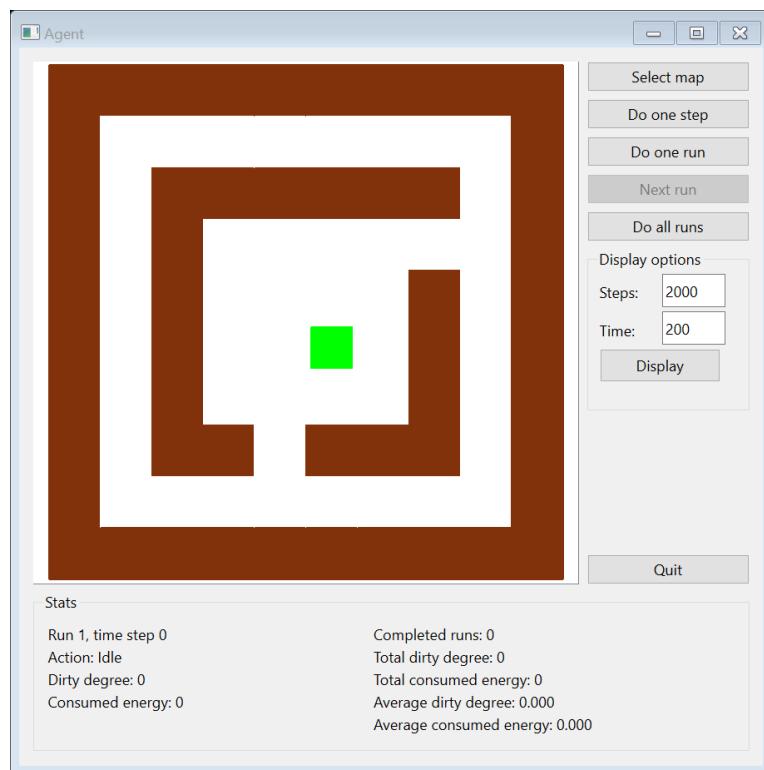


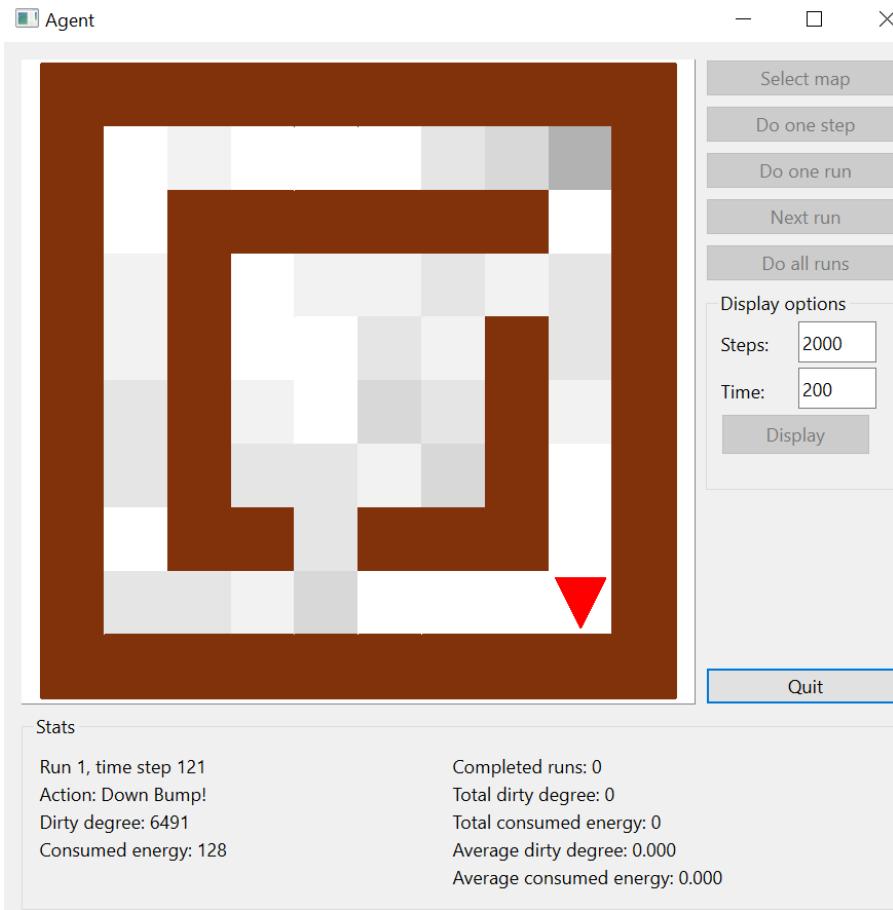
4. Click on select map for simulation.



Here we change the path , take the path where the map file exist and click ok button.

5. After this, the following map will displayed





6. The above image shows, the Vacuum cleaner World simulation successfully run.

Purpose of this project:

This project simulates an intelligent vacuum cleaning agent operating in a grid-based environment. The simulation aims to model how an autonomous cleaning agent can interact with its surroundings, make decisions, and optimize its cleaning performance.

Key Objectives of This Simulation:

1. Autonomous Decision-Making:

- The agent perceives its environment(e.g. detecting dirt, obstacles, or walls).
- It decides whether to clean, move in a specific direction, or stay idle.
- This mimics how AI-driven robot vacuums(e.g., Roomba)operate in real-world homes.

2. Reinforcement Learning & Optimization:

- The agent could learn optimal cleaning paths over time to improve efficiency.
- AI techniques like Reinforcement Learning(RL) can help the agent improve decision-making.
- The project can analyze cleaning efficiency, energy consumption, and time taken.

3. Simulation of Real-World Environments:

- The world class represents different environments with varying dirt levels and obstacles.

- Dirt is added randomly, introducing uncertainty, making the simulation more realistic.
- The simulation allows testing and optimization of cleaning strategies in different settings.

4. Path Planning & Navigation:

- The agent moves based on a simple random decision model, but future versions could implement AI-driven path-planning.
- Avoiding unnecessary movements can lead to energy-efficient cleaning.

5. Human Interaction & Experimentation:

- Users can load different map configurations and test the agents performance.
- It serves as a platform for AI research, robotics development, and simulation studies.