

Об'єктно орієнтоване програмування
Лабораторна робота №12
Створення додатків з нестандартним
інтерфейсом



Теоретичні відомості

Приклад. Написати програму, яка на прозорому вікні зображає новорічну ялинку. Також реалізовано один новорічний ліхтарик червоного кольору. Вікно можна переміщувати за допомогою лівої кнопки миші, а відкривати контекстне меню – правою кнопкою.

Лістинг 12.1 – Файл treelight.h

```
#ifndef TREELIGHT_H
#define TREELIGHT_H

#include <QPainter>

class TreeLight
{
    int x, y;
    int r;
    int inc;
public:
    TreeLight(int x, int y, int r);
    void Draw(QPainter *p);
    void Next();
};

#endif // TREELIGHT_H
```

Лістинг 12.2 – Файл treelight.cpp

```
#include "treelight.h"

TreeLight::TreeLight(int x, int y, int r)
{
    this->x = x;
    this->y = y;
    this->r = r;
    inc = 2;
}

void TreeLight::Draw(QPainter *p)
{
    p->drawEllipse(x-r/2, y-r/2, r, r);
}

void TreeLight::Next()
{
    r += inc;
    if (r > 20 || r < 0)
```

```
        inc = - inc;
    }
```

Лістинг 10.8 – Файл mainwindow.h

```
#ifndef MAINWINDOW_H
#define MAINWINDOW_H

#include <QMainWindow>
#include "treelight.h"
namespace Ui {
class MainWindow;
}

class MainWindow : public QMainWindow
{
    Q_OBJECT

public:
    explicit MainWindow(QWidget *parent = nullptr);
    ~MainWindow();

private:
    Ui::MainWindow *ui;
    bool eventFilter(QObject* watched, QEvent* event);
    QPoint clickpos;
    QTimer *paintTimer;
    TreeLight *l1;
private slots:
    void on_action_1();
    void updatePixmap();
};

#endif // MAINWINDOW_H
```

Лістинг 10.8 – Файл mainwindow.cpp

```
#include "mainwindow.h"
#include "ui_mainwindow.h"
#include <QPainter>
#include <QDir>
#include <QMouseEvent>
#include <QMenu>
#include <QTimer>
#include "treelight.h"

MainWindow::MainWindow(QWidget *parent) :
    QMainWindow(parent),
    ui(new Ui::MainWindow)
{
    ui->setupUi(this);
    setWindowFlags(Qt::Widget | Qt::FramelessWindowHint);
    setParent(nullptr); // Create TopLevel-Widget
    setAttribute(Qt::WA_NoSystemBackground, true);
```

```

        setAttribute(Qt::WA_TranslucentBackground, true);
        setAttribute(Qt::WA_PaintOnScreen); // not needed in Qt 5.2
and up

        ui->widget->installEventFilter(this);

        paintTimer = new QTimer(this);
        paintTimer->start(10);
        connect(paintTimer, SIGNAL(timeout()), this,
SLOT(updatePixmap()));
        ll = new TreeLight(125, 150, 1);
    }

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

bool MainWindow::eventFilter(QObject* watched, QEvent* event)
{
    if(watched == ui->widget && event->type() == QEvent::Paint)
    {
        QPainter p(ui->widget);
        QBrush b(Qt::red);
        QPen pn(Qt::red);
        p.setPen(pn);
        p.setBrush(b);

        QImage img(QDir::currentPath() + "/tree.png");
        p.drawImage(ui->widget->rect(), img, img.rect());

        ll->Draw(&p);
    }

    if(watched == ui->widget && event->type() ==
QEvent::MouseMove)
    {
        QMouseEvent *me = reinterpret_cast<QMouseEvent
*>(event);
        move(me->globalPos() - clickpos);
    }
    if(watched == ui->widget && event->type() ==
QEvent::MouseButtonPress)
    {
        QMouseEvent *me = reinterpret_cast<QMouseEvent
*>(event);
        if (me->button() == Qt::MouseButton::LeftButton)
        {
            clickpos = reinterpret_cast<QMouseEvent *>(event)-
>pos();

```

```

    }
    if (me->button() == Qt::MouseButton::RightButton)
    {
        QMenu myMenu;
        myMenu.addAction("Вийти", this,
        SLOT(on_action_1()));
        myMenu.exec(me->globalPos());
    }

    return false;
}

void MainWindow::on_action_1()
{
    this->close();
}

void MainWindow::updatePixmap()
{
    l1->Next();
    repaint();
}
}

```



Рисунок 12.1 – Результат роботи програми

Варіанти завдань

Завдання:

Доопрацюйте вище наведений приклад, розширивши його можливості. Створіть власну унікальну ялинку. Рекомендовані наступні напрями доопрацювання:

- Збільшити кількість ліхтариків, різноманіття їх форми та кольору;
- Розробіть кілька алгоритмів анімації ліхтариків (об'єднайте ліхтарі в гірлянди тощо), додайте можливість вибору алгоритму за допомогою контекстного меню.
- Додайте анімований сніг. Сніг може лежати і переливатись кольорами або йти.
- Реалізуйте будь-які інші ідеї, проявляйте творчість.

*Розробник кращої новорічної ялинки отримає додаткові бали.
Вітаю з прийдешнім новим роком і Різдвом христовим.*