

Министерство образования Республики Беларусь  
Учреждение образования  
«Брестский государственный технический университет»  
Кафедра ИИТ

Лабораторная работа №3  
по дисциплине: **ООТПиСП**  
Тема: Создание интерфейса простых приложений

**Выполнил**  
студент 3 курса  
Корнаसेвич И. Д.

**Проверил**  
Булей Е. В.

**Цель работы** отработка умений и навыков описания событий в приложениях.

**Задание** Напишите программу, которая находит в массиве 15x15 числа являющиеся степенью 3. Создайте интерфейс программы: в таблице 15x15 числа получены случайным образом; создать кнопки выполняемых действий; результат действий подсвечивать цветом; в поле надписи выводить количество найденных чисел.

#### mainwindow.h

```
1  #ifndef MAINWINDOW_H
2  #define MAINWINDOW_H
3
4  #include <QMainWindow>
5  #include <QTableWidgetItem>
6
7  QT_BEGIN_NAMESPACE
8  namespace Ui { class MainWindow; }
9  QT_END_NAMESPACE
10
11 class MainWindow : public QMainWindow
12 {
13     Q_OBJECT
14
15 public:
16     MainWindow(QWidget *parent = nullptr);
17     ~MainWindow();
18
19 private:
20     Ui::MainWindow *ui;
21
22     void updateNumbersFound(int newCount);
23
24     bool isPowOfN(int x, int n);
25
26     void highlightTable(int n);
27
28     void createTable();
29
30 private slots:
31     void fillTableRandomly();
32 };
33 #endif // MAINWINDOW_H
```

#### mainwindow.cpp

```
1  #include "mainwindow.h"
2  #include "ui_mainwindow.h"
3
4  #include <QRandomGenerator64>
5  #include <QAbstractButton>
6
7  MainWindow::MainWindow(QWidget *parent)
8      : QMainWindow(parent)
9      , ui(new Ui::MainWindow)
10 {
11     ui->setupUi(this);
12
13     createTable();
14     fillTableRandomly();
15     highlightTable(3);
16
17     ui->tableWidget->horizontalHeader()->setSectionResizeMode(QHeaderView::Stretch);
18     ui->tableWidget->verticalHeader()->setSectionResizeMode(QHeaderView::Stretch);
19
20     connect(ui->pushButton, &QAbstractButton::pressed, this, &MainWindow::fillTableRandomly);
21     connect(ui->pushButton, &QAbstractButton::pressed, [this]() { this->highlightTable(3); });
22 }
23
24 MainWindow::~MainWindow() {
25     delete ui;
26 }
27
28 void MainWindow::updateNumbersFound(int newCount)
29 {
```

```

30     ui->label->setText("Found: " + QString::number(newCount));
31 }
32
33 bool MainWindow::isPowOfN(int x, int n){
34     while (x % n == 0 && x != 0) {
35         x /= n;
36     }
37     return x == 1;
38 }
39
40 void MainWindow::highlightTable(int n)
41 {
42     auto numbersFound = 0;
43     auto model = ui->tableWidget->model();
44     auto width = model->columnCount();
45     auto height = model->rowCount();
46     for (auto i = 0; i < height; i++){
47         for (auto j = 0; j < width; j++){
48             auto index = model->index(i, j);
49             auto value = index.data().toInt();
50             if (isPowOfN(value, n)){
51                 numbersFound++;
52                 ui->tableWidget->item(i, j)->setBackground(Qt::lightGray);
53             } else {
54                 ui->tableWidget->item(i, j)->setBackground(Qt::white);
55             }
56         }
57     }
58     updateNumbersFound(numbersFound);
59 }
60
61 void MainWindow::createTable()
62 {
63     auto model = ui->tableWidget->model();
64     auto width = model->columnCount();
65     auto height = model->rowCount();
66     for (auto i = 0; i < height; i++){
67         for (auto j = 0; j < width; j++){
68             QTableWidgetItem *item=new QTableWidgetItem;
69             ui->tableWidget->setItem(i, j, item);
70         }
71     }
72 }
73
74 void MainWindow::fillTableRandomly()
75 {
76     auto model = ui->tableWidget->model();
77     auto width = model->columnCount();
78     auto height = model->rowCount();
79     auto rand = QRandomGenerator::securelySeeded();
80     for (auto i = 0; i < height; i++){
81         for (auto j = 0; j < width; j++){
82             ui->tableWidget->item(i, j)->setData(0, rand.bounded(1, 99));
83         }
84     }
85 }

```

---

MainWindow

32	7	44	96	23	50	91	12	41	91	95	29	85	50	21
53	57	7	27	5	8	89	94	44	11	1	44	91	15	89
61	22	66	50	26	67	91	42	42	74	85	6	78	6	64
78	34	18	39	4	24	51	8	93	31	21	94	10	96	12
98	93	39	30	96	36	46	23	64	60	94	25	16	51	77
64	79	89	77	89	74	93	41	80	75	83	82	17	18	76
31	45	61	39	79	94	57	40	38	1	72	87	60	29	69
60	23	94	34	13	43	45	23	47	75	53	74	10	76	82
64	20	35	38	53	34	88	2	11	24	70	64	2	34	27
49	18	80	43	21	15	7	25	66	10	86	4	94	90	23
60	1	48	78	85	48	35	67	60	20	78	86	13	59	13
87	84	2	7	46	56	13	26	93	36	13	56	83	64	6
69	10	48	57	8	4	90	47	30	22	56	44	11	20	29
43	16	31	51	22	12	12	37	59	4	16	92	30	81	27
21	72	19	96	78	81	26	31	33	37	72	51	8	49	93

Found: 8

PushButton

MainWindow

10	81	61	69	64	49	54	52	46	80	79	30	46	6	21
67	18	34	51	80	77	35	87	51	63	72	68	72	58	91
54	98	15	80	65	2	92	6	54	90	69	64	7	81	94
34	73	96	48	14	21	35	1	33	56	24	74	6	85	50
89	67	78	98	37	7	55	98	49	90	90	37	77	55	70
11	98	55	39	89	60	35	38	56	42	51	10	44	85	77
19	8	58	29	71	34	97	30	78	31	37	17	80	19	2
14	31	94	98	95	26	21	59	43	23	17	36	76	26	21
87	97	43	17	10	45	7	88	67	56	94	22	17	24	7
64	97	20	92	15	56	94	84	25	55	61	92	11	49	19
40	79	88	95	79	70	53	69	58	20	70	38	1	37	47
21	85	35	74	93	97	79	69	5	71	40	57	35	66	6
79	89	96	68	37	89	76	3	83	74	27	65	95	76	30
68	34	98	27	66	24	86	73	74	57	70	66	25	87	51
46	10	33	82	68	85	21	34	86	91	46	36	20	29	37

Found: 7

PushButton

**Вывод** Создана программа, позволяющая генерировать таблицу чисел и находить в ней те числа, которые являются степенью 3.