Национальный Исследовательский Университет

«Московский Энергетический Институт»

НИУ «МЭИ»

Направление: Электроника и наноэлектроника

Профиль: Электроника и наноэлектроника

Отчёт

по лабораторной работе №9

по «Введению в программирование»

**«Структуры»**

Вариант 27

Студент: Сырцев В.Д.

Группа: ЭР-02-19

Вариант:27

Преподаватель: Раскатова М.В

# 1.Условие задачи:

Имеются сведения о прохождении детьми прививок: фами­лия, имя, год рождения, отметки о прививках против туберку­леза, полиомиелита, кори, коклюша, столбняка. Вывести сведения о детях k-возраста, у которых отсутствует хотя бы одна прививка. Определить их число.

# 2.Анализ задачи:

Для решения этой задачи необходимо использовать структуры-типы, которые группируют несколько отдельных переменных вместе. Одним из простейших пользовательских типов данных является структура. Структура позволяет сгруппировать переменные разных типов в единое целое. Необходимо создать структуру, в которой будут разные типы переменных: wstring-для хранения имени и фамилии, vector<int>- дата рождения, int - есть ли разные прививки. Затем необходимо создать массив-тип которого является созданная нами структура. После ввода данных нужно сначала проверить является ли возраст ребенка равным введенному значению возраста, которое ввел пользователь. Затем проверяем есть ли все прививка у ребенка, если нет хотя бы одной окрашиваем его строку в красный цвет, а также увеличиваем счетчик непривитых детей.

Структура children

struct children

{

wstring name;

wstring surname;

vector<int> date;

int tuber;

int poly;

int kor;

int koklush;

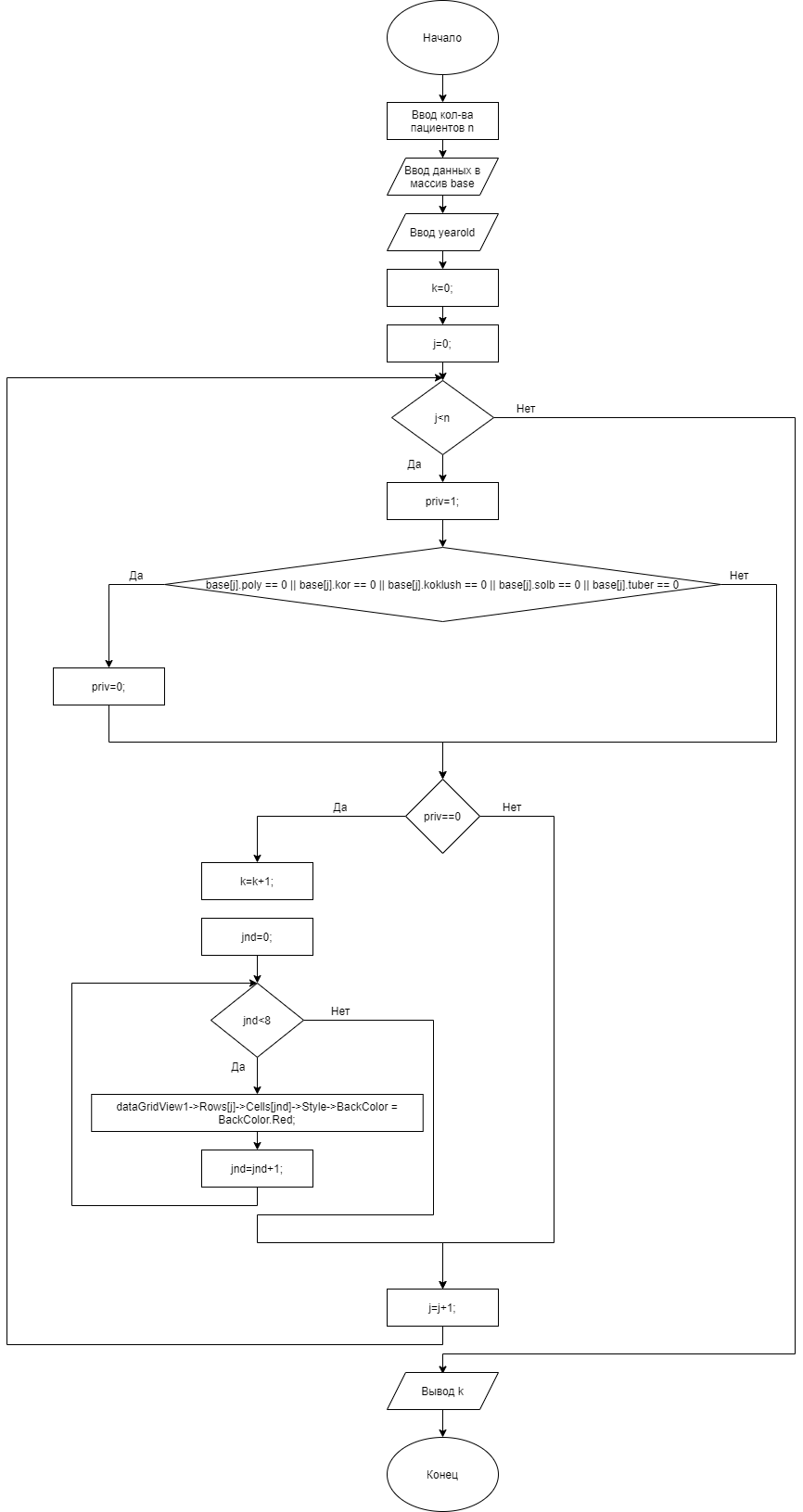
int solb;

};

# 3.Состав данных

|  |  |  |  |
| --- | --- | --- | --- |
| **Имя** | **Смысл** | **Тип** | **Структура** |
| **Входные данные** | | | |
| base | Массив, типа children, содержащий фамилию, имя, дату рождения, наличие прививок. | children | Одномерный массив |
| yearold | Возраст, у которого необходимо проверить наличие прививок | Целое | Простая переменная |
| **Выходные данные** | | | |
| k | Кол-во детей, у которых нет хотя бы одной прививка | Целое | Простая переменная |
| **Промежуточные данные** | | | |
| j | Счетчик по массиву base | Целое | Простая переменная |

# 4.Блок-схема:



# 5.Программа:

int yearold = (int)numericUpDown2->Value;

children\* base=new children[n];

for (int j = 0; j < n; j++)

{

base[j].surname = zagonka(Convert::ToString(dataGridView1->Rows[j]->Cells[0]->Value));

base[j].name= zagonka(Convert::ToString(dataGridView1->Rows[j]->Cells[1]->Value));

base[j].date = dated(Convert::ToString(dataGridView1->Rows[j]->Cells[2]->Value));

base[j].tuber = Convert::ToInt16(dataGridView1->Rows[j]->Cells[3]->Value);

base[j].poly = Convert::ToInt16(dataGridView1->Rows[j]->Cells[4]->Value);

base[j].kor = Convert::ToInt16(dataGridView1->Rows[j]->Cells[5]->Value);

base[j].koklush = Convert::ToInt16(dataGridView1->Rows[j]->Cells[6]->Value);

base[j].solb = Convert::ToInt16(dataGridView1->Rows[j]->Cells[7]->Value);

}

vector<int>currentdate;

currentdate=dated(Convert::ToString(dateTimePicker1->Value));

int k = 0;

for (int j = 0; j < n; j++)

{

bool priv=1;

if ((base[j].date[2] > currentdate[2]-18 && base[j].date[2] <= currentdate[2]) && (base[j].date[1] > 0 && base[j].date[1] < 13) && (base[j].date[0] > 0 && base[j].date[0]<=31))

{

if (currentdate[2] - base[j].date[2] == yearold && (currentdate[0] - base[j].date[0] >= 0 && currentdate[1] - base[j].date[1] ==0 || currentdate[1] - base[j].date[1] > 0))

{

if (base[j].poly == 0 || base[j].kor == 0 || base[j].koklush == 0 || base[j].solb == 0 || base[j].tuber == 0)

{

priv = 0;

}

}

if (priv == 0)

{

k++;

for (int jnd = 0; jnd < 8; jnd++)

{

dataGridView1->Rows[j]->Cells[jnd]->Style->BackColor = BackColor.Red;

}

}

else

{

for (int jnd = 0; jnd < 8; jnd++)

{

dataGridView1->Rows[j]->Cells[jnd]->Style->BackColor = BackColor.White;

}

}

}

else

{

MessageBox::Show("Неверный ввод");

}

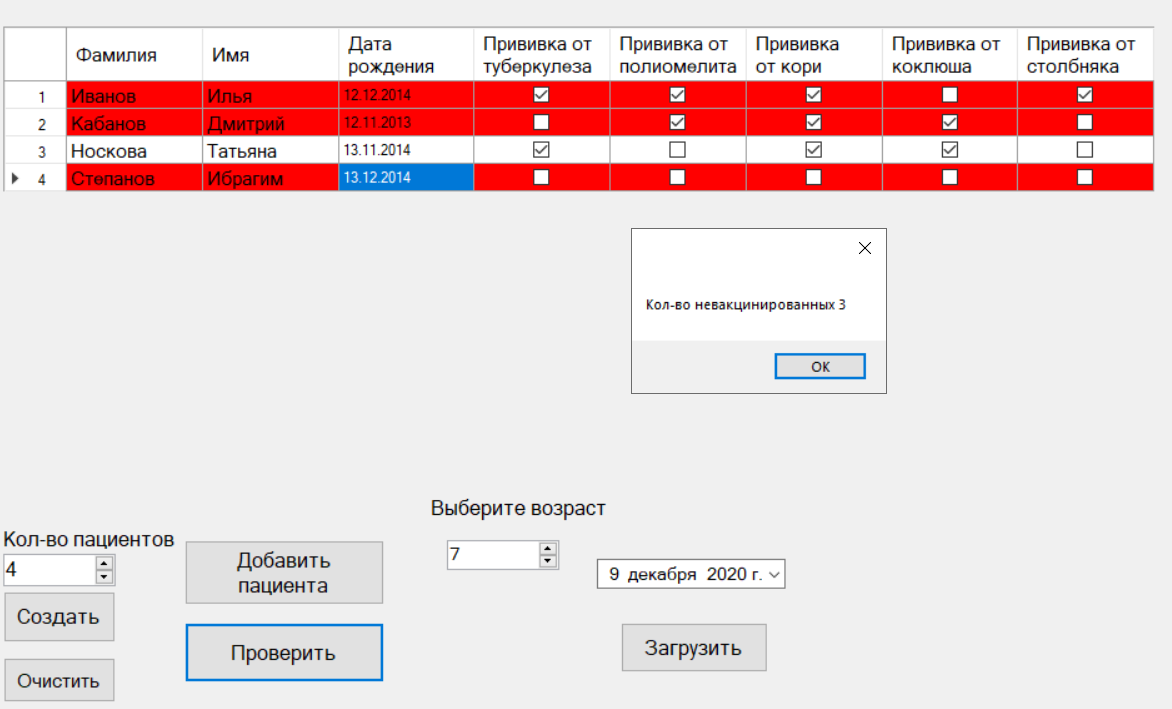
}

MessageBox::Show("Кол-во невакцинированных "+ k);

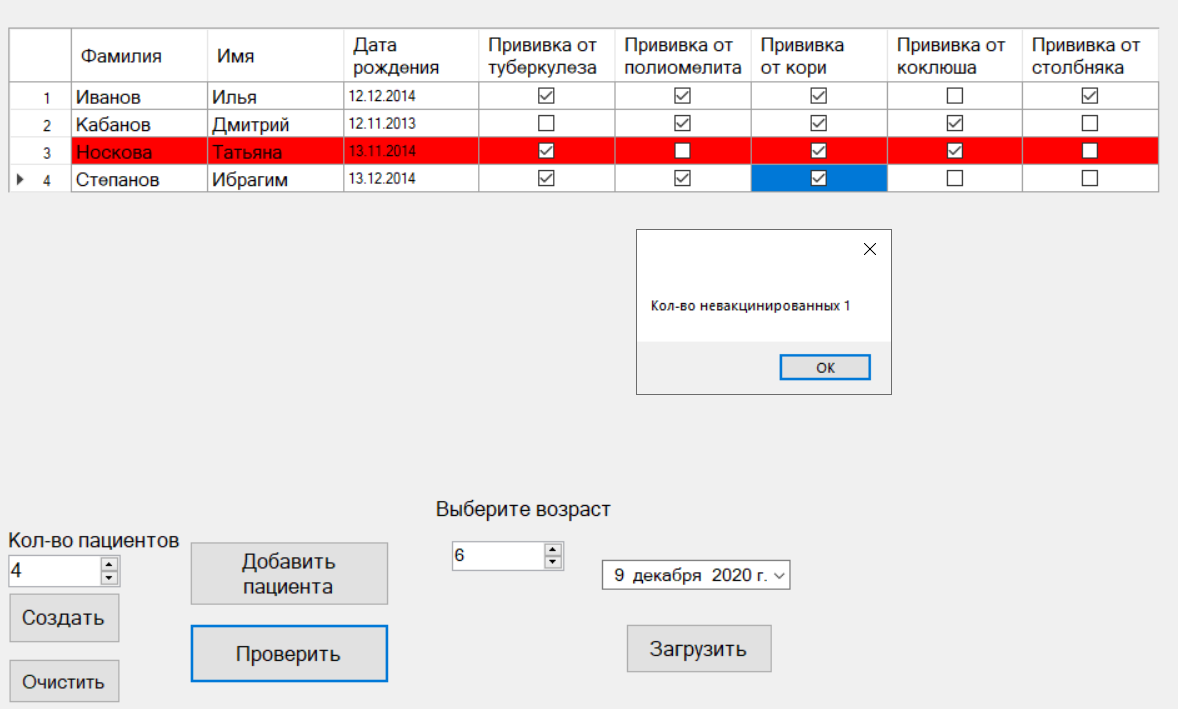
delete[]base;

# 6.Результаты:

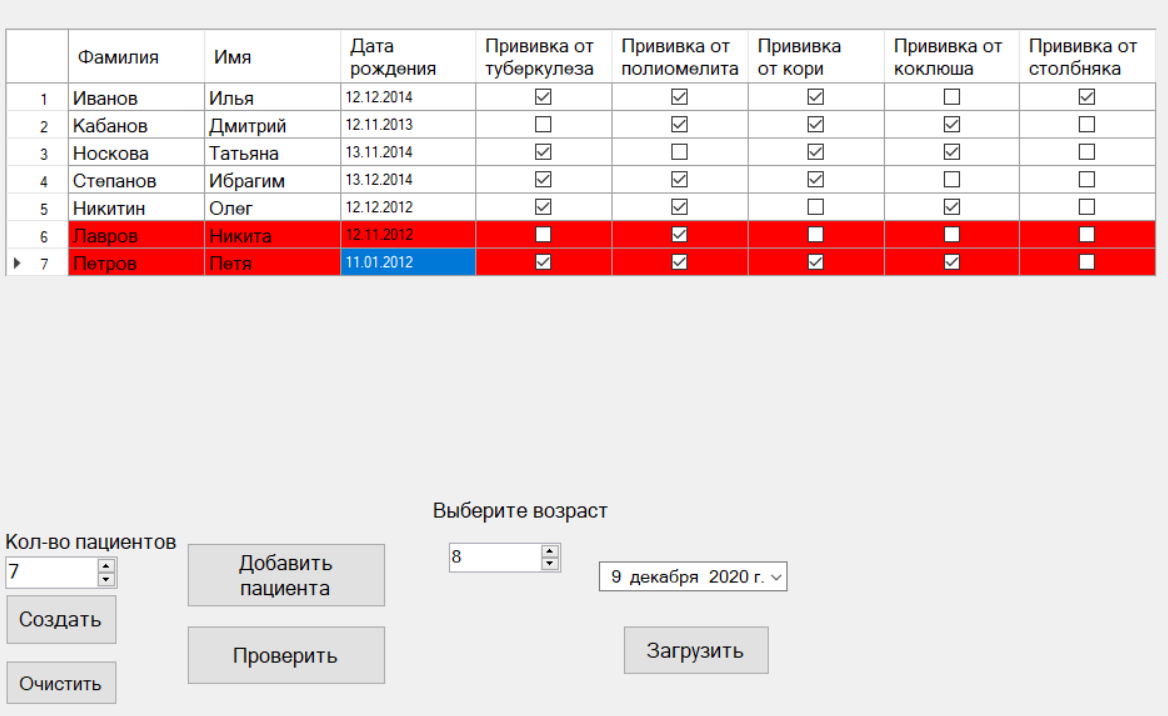
1.Тест



2.Тест



3.Тест



# Вывод:

Программа работает корректно.