Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное

учреждение высшего образования

ИРКУТСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ

ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

Институт информационных технологий и анализа данных

Факультет Информационные системы и технологии

Отчёт по лабораторной работе № 3

по дисциплине Технология программирования

Выполнил студент ИСТб-20-3 М. Д. Ярков 27.10.2021

номер группы подпись И. О. Фамилия Дата

Принял Доцент З. А. Бахвалова . .2021

должность подпись И. О. Фамилия Дата

Иркутск 2021

### Постановка задачи

### Создать класс для цвета, заданного в пространстве HSV, а также под каждую характеристику отдельный тип. Реализовать операции:

### Добавление/вычитание красного цвета

### Добавление/вычитание синего цвета

### Добавление/вычитание зеленого цвета

### Добавление/вычитание насыщености

### Добавление/вычитание яркости

### Перевод в RGB

### Словесное описание алгоритма

1. При запуске программы создаётся объект color класса Color, инициализируется интерфейс и производятся изменения значений на форме
2. При нажатии кнопки «Изменить цвета» или нажатии Enter во время изменения красного, зеленого или синего цвета, вызывается метод ChangeRgb:
   1. Производится изменение полей R, G, B объекта color на те, которые были введены:
      1. При изменении R, G или B класс Color автоматически проверяет вхождение изменённого числа в пределы от 0 до 255, и если число недопустимо, появляется ошибка
      2. В случае вхождения в пределы введённое число добавляется во внутреннюю переменную, а также обновляются значения HSV в соответствии с формулой 1
   2. Вызывается метод UpdateLabels:
      1. Производится изменение цвета индикатора на цвет, который содержится в color
      2. Производится изменение надписей интерфейса для отображения значений HSV и RGB
3. При нажатии кнопки «Изменить насыщ. / яркость» или нажатии Enter во время изменения насыщенности или яркости, вызывается метод ChangeHsv:
   1. Производится изменение полей S, V объекта color на те, которые были введены:
      1. При изменении S, V класс Color автоматически проверяет вхождение изменённого числа в пределы от 0 до 100 (для H максимальное значение 360), и если число недопустимо, появляется ошибка
      2. В случае вхождения в пределы введённое число добавляется во внутреннюю переменную, а также обновляются значения RGB в соответствии с формулой 2
   2. Вызывается метод UpdateLabels

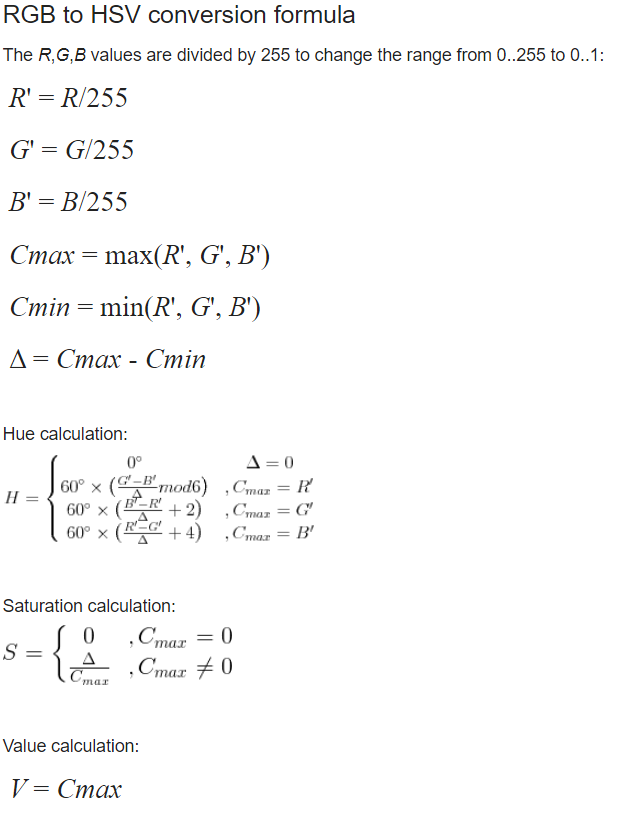


Рисунок 1 – Математическое описание алгоритма перевода RGB в HSV

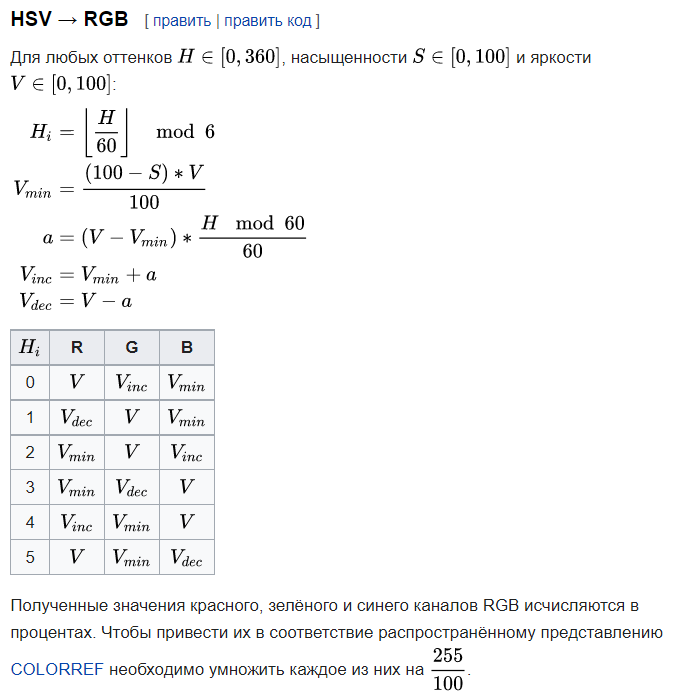


Рисунок 2 – Математическое описание алгоритма перевода HSV в RGB

### Таблица спецификаций

|  |  |  |
| --- | --- | --- |
| Имя метода / поля / переменной / конструктора | Назначение | Тип |
| Класс Form1 | | |
| color | объект класса | объект readonly Color |
| переменные отсутствуют | | |
| Form1() | инициализация при запуске программы | Конструктор |
| переменные отсутствуют | | |
| UpdateLabels() | вывод данных на форму | метод private void |
| переменные отсутствуют | | |
| ChangeRgb(object sender, EventArgs e)) | изменение значений RGB | метод private void |
| sender | обязательная переменная, не используется | переменная object |
| e | обязательная переменная, не используется | переменная EventArgs |
| KeyPressedRgb(object sender, KeyEventArgs e) |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

### Таблица тестов

| Номер теста | Что проверяем | Входные данные | Выходные данные |
| --- | --- | --- | --- |
| 1 | Изменение любого цвета в разрешённых промежутках | 30  60  120 | Установлены соответствующие цвета, индикатор перекрашивается в другой цвет, обновлено представление в HSV |
| 2 | Изменение насыщенности/яркости в разрешённых промежутках | 95  60 | Установлена соответствующая насыщенность/яркость, индикатор перекрашивается в другой цвет, обновлено представление в RGB |
| 3 | Попытка установить значение за возможными пределами | -200 | Автоматическое изменение значения поля на ближайшее к введённому разрешённое число |

### Код программы

Код программы находится по ссылке https://github.com/mxkmn/RepoForMyTeacher\_CSharpEdition/tree/main/Lab3Var12

Файл с автотестами:

|  |
| --- |
| using Microsoft.VisualStudio.TestTools.UnitTesting;  using Lab3Var12;  using System;  using System.Collections.Generic;  using System.Linq;  using System.Text;  using System.Threading.Tasks;  namespace Lab3Var12.Tests {  [TestClass()]  public class ColorTests {  [TestMethod()]  public void ColorsCreateTest1() {  Assert.ThrowsException<System.ArgumentException>(() => new Color(ColorModels.HSV, 100, 100, 101));  }  [TestMethod()]  public void ColorsCreateTest2() {  Assert.ThrowsException<System.ArgumentException>(() => new Color(ColorModels.HSV, 100, 100, -1));  }  [TestMethod()]  public void ColorsCreateTest3() {  Assert.ThrowsException<System.ArgumentException>(() => new Color(ColorModels.RGB, 100, 100, 256));  }  [TestMethod()]  public void ColorsCreateTest4() {  Assert.ThrowsException<System.ArgumentException>(() => new Color(ColorModels.RGB, -101, 100, 100));  }  [TestMethod()]  public void ColorsCreateTest5() {  Assert.ThrowsException<System.ArgumentException>(() => new Color(ColorModels.HSV, 361, -100, 0));  }  [TestMethod()]  public void ColorsCreateTest6() {  Assert.ThrowsException<System.ArgumentException>(() => new Color(ColorModels.HSV, -1, 100, 100));  }  [TestMethod()]  public void ColorsCreateTest7() {  Color color = new Color();  Assert.AreEqual(180, color.H);  }  [TestMethod()]  public void ColorsCreateTest8() {  Color color = new Color(ColorModels.HSV, 100, 100, 0);  Assert.AreEqual(100, color.S);  }  [TestMethod()]  public void ColorsCreateTest9() {  Color color = new Color(ColorModels.RGB, 100, 255, 0);  Assert.AreEqual(255, color.G);  }  }  } |

### Тесты

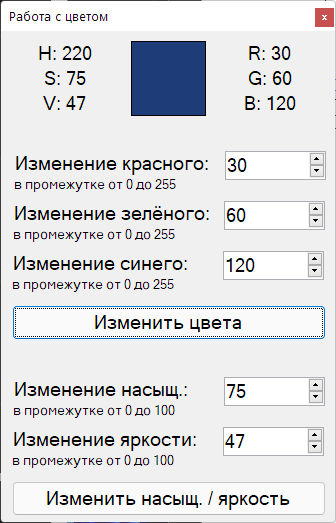


Рисунок 3 – тест 1

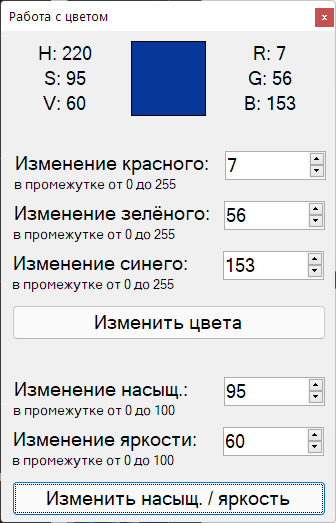


Рисунок 4 – тест 2

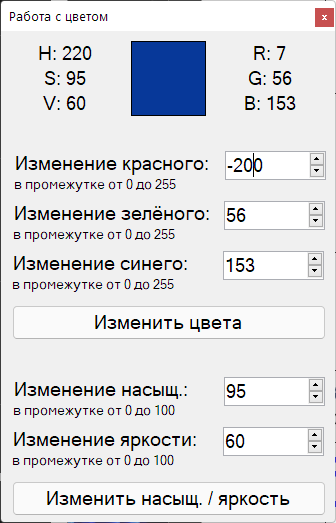


Рисунок 5 – тест 3 (введено число, не входящее в промежуток)

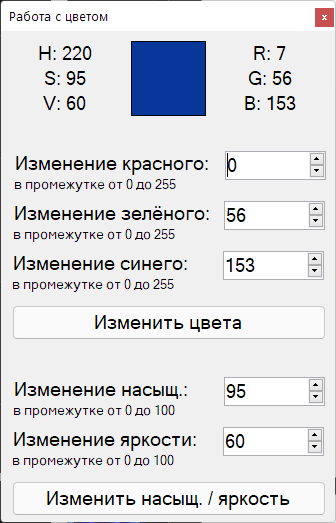


Рисунок 6 – тест 3 (введённое число автоматически изменилось на минимально возможное)