МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Курский государственный университет»

кафедра программного обеспечения и администрирования

информационных систем

Отчёт

по лабораторной работе №3

«Перегрузка операций»

по дисциплине

###### «Объектно-ориентированные языки и системы»

|  |  |
| --- | --- |
| Выполнил(а): | студент группы 313  Козявин Максим  Сергеевич |
| Проверил: | к.т.н., доцент  кафедры ПОиАИС  Макаров К.С. |

Курск

2023

**Цель работы:** получить практические навыки по созданию перегруженных операций.

**Индивидуальное задание (вариант 4):**

В классе MyInt (см. каталог v4) перегрузите четыре целочисленных арифметических операции (+, -, \* и /) так, чтобы их можно было использовать для операций с объектами класса MyInt. Если результат какой-либо из операций выходит за границы типа int (в 32-битной системе), имеющие значения от 2 147 483 647 до -2 147 483 648, то операция должна послать сообщение об ошибке и завершить программу. Такие типы данных полезны там, где ошибки могут быть вызваны арифметическим переполнением, которое недопустимо. Для облегчения проверки переполнения Вы можете выполнять вычисления с использованием типа double. Напишите программу для проверки этого класса.

**Основные теоретические положения**

C# позволяет перегружать операции для работы с пользовательскими типами данных как с базовыми. Это позволяет работать с объектами не с использованием методов, а через стандартные операции такие как +, -, \* и др. В языке C# операции являются статическими методами, которые возвращают результат операции, а в качестве аргументов используют её операнды. Когда задаётся операция для какого-либо класса, то перегружается соответствующий метод аналогично тому, как перегрузился бы любой метод класса, если бы это потребовалось.

**Экспериментальные результаты**

Код программы для решения задачи:

//MyInt.cs

namespace v4 {

public class MyInt {

private long data;

public MyInt() {

data = 0;

}

public MyInt(int d) {

data = d;

}

public MyInt(long d) {

if (d > Int32.MaxValue || d < Int32.MinValue) {

throw new OverflowException($"Value {d} overflows Int32");

}

data = d;

}

public override string ToString() {

return data.ToString();

}

public static MyInt operator + (MyInt a, MyInt b) {

long res = a.data + b.data;

if (res > Int32.MaxValue || res < Int32.MinValue) {

throw new OverflowException($"Overflow in operation {a.data} + {b.data}");

}

return new MyInt(res);

}

public static MyInt operator - (MyInt a, MyInt b) {

long res = a.data - b.data;

if (res > Int32.MaxValue || res < Int32.MinValue) {

throw new OverflowException($"Overflow in operation {a.data} - {b.data}");

}

return new MyInt(res);

}

public static MyInt operator \* (MyInt a, MyInt b) {

long res = a.data \* b.data;

if (res > Int32.MaxValue || res < Int32.MinValue) {

throw new OverflowException($"Overflow in operation {a.data} \* {b.data}");

}

return new MyInt(res);

}

public static MyInt operator / (MyInt a, MyInt b) {

long res = a.data / b.data;

if (res > Int32.MaxValue || res < Int32.MinValue) {

throw new OverflowException($"Overflow in operation {a.data} / {b.data}");

}

return new MyInt(res);

}

}

}

// Program.cs

namespace v4 {

class Program {

public static MyInt? a;

public static MyInt? b;

static void Main(string[] args) {

while (true) {

Console.Write("Enter operation (+, -, \*, /, exit): ");

string? operation = Console.ReadLine();

if (operation == "exit")

break;

Console.Write("Enter A number: ");

string? aStr = Console.ReadLine();

Console.Write("Enter B number: ");

string? bStr = Console.ReadLine();

a = new MyInt(long.Parse(aStr ?? "0"));

b = new MyInt(long.Parse(bStr ?? "0"));

switch (operation) {

case "+":

Console.WriteLine(a + b);

break;

case "-":

Console.WriteLine(a - b);

break;

case "\*":

Console.WriteLine(a \* b);

break;

case "/":

Console.WriteLine(a / b);

break;

default:

Console.WriteLine("Wrong input.");

break;

}

}

}

}

}

Тестирование:

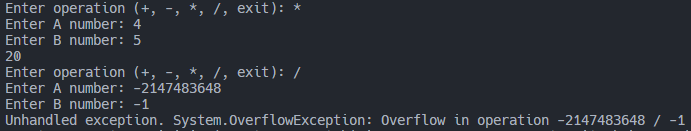


Рисунок 1 – Тест 1 задачи 1

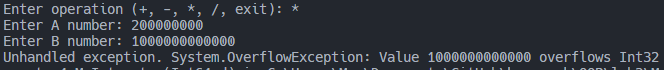


Рисунок 2 – Тест 2 задачи 1

**Вывод:** в данной лабораторной работе я получил практические навыки по созданию перегруженных операций на языке C#.