**Разработка на языке программирования Rust.**

Выполнил: Беликов Константин  
Группа: ИУ5-36Б  
Дата: 20.11.24г.

Описание задания:

1. По заданию, программа должна реализовывать макрос, заменяющий перегрузку одной из операций +-\*/ для структуры Point
2. Во второй части нужно было реализовать макрос, который выполняет ту же задачу, что и первый, но способен при этом перегружать сразу любое количество операций (+-\*/)

Код программы:

#[derive(Debug, Copy, Clone)]

struct Point {

    x: i32,

    y: i32,

}

//Реализация макроса для перегрузки операторов +-\*/

macro\_rules! override\_op {

    ($tr:ident, $func:ident) => (

        impl std::ops::$tr for Point {

            type Output = Self;

            fn $func(self, \_rhs: Self) -> Self {

                let \_x = std::ops::$tr::$func(self.x, \_rhs.x);

                let \_y = std::ops::$tr::$func(self.y, \_rhs.y);

                Self {x: \_x, y: \_y}

            }

        }

    )

}

//Реализация макроса, который применякт override\_op для произвольного количествы аргументов

macro\_rules! override\_ops {

    ($tr:ident, $func:ident) => (

        override\_op!($tr, $func);

    );

    ($tr:ident, $func:ident, $($coms:ident),+) => (

        override\_op!($tr, $func);

        override\_ops!($($coms),+);

    )

}

override\_ops!(Add, add, Sub, sub, Mul, mul, Div, div);

fn main() {

    let p1 = Point{x: 10, y: 15};

    let p2 = Point{x: 5, y: 3};

    let points: [Point; 4] = [p1 + p2, p1 - p2, p1 \* p2, p1 / p2];

    //Выводим результат применения операций +-\*/ над точками p1, p2

    for p in points.iter() {

        println!("{:?}", p);

    }

}

Снимки экрана:

