

<b>Programowanie Komputerów 4</b>					
Rok akademicki	Termin	Rodzaj studiów	Kierunek	Prowadzący	Grupa
<b>2019/2020</b>	Zajęcia zdalne w trakcie pandemii	<b>SSI</b>	<b>INF</b>	<b>KP</b>	warunek

Temat:

# Skala szarości w C#

**Autor:**  
**Dominik Nowocień**

# Przedstawienie i omówienie realizowanego problemu

Celem projektu było stworzenie prostego edytora zapewniającego zmianę kolorów wybranego obrazu na jego odpowiednik w kolorach czarno-białych. Ze względu na dużą popularność algorytmów wyciągających średnią z kolorów (czerwonego, zielonego i niebieskiego) zdecydowałem się nie używać wyżej wymienionego sposobu tylko wyszukać inne funkcje realizujące zadany problem.

## Uzasadnienie wyboru projektu:

Łatwa możliwość późniejszej rozbudowy w celu poszerzania wiedzy i uczenia się nowych technik programowania obiektowego jak i pracy z zewnętrznymi bibliotekami.

## Zdefiniowanie wybranego fragmentu kodu realizowanego z wykorzystaniem bibliotek zewnętrznych (ASM/CPP)

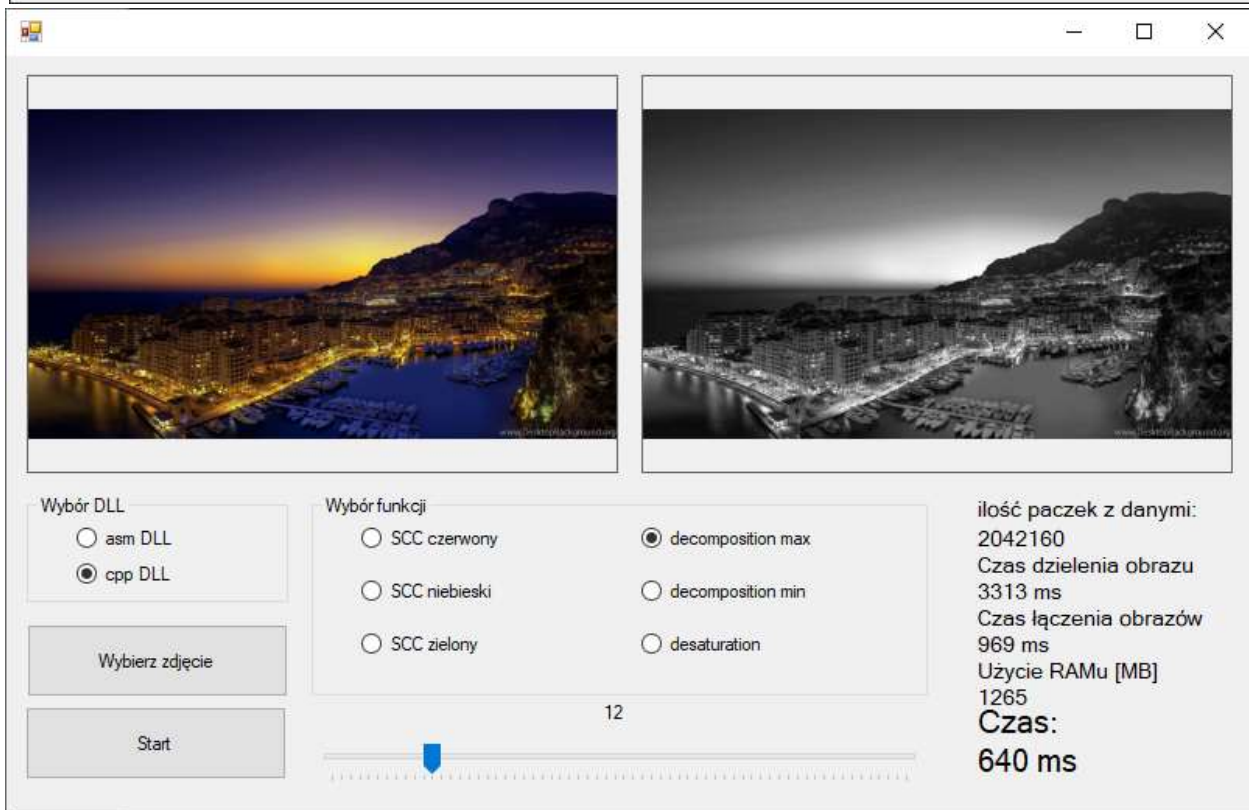
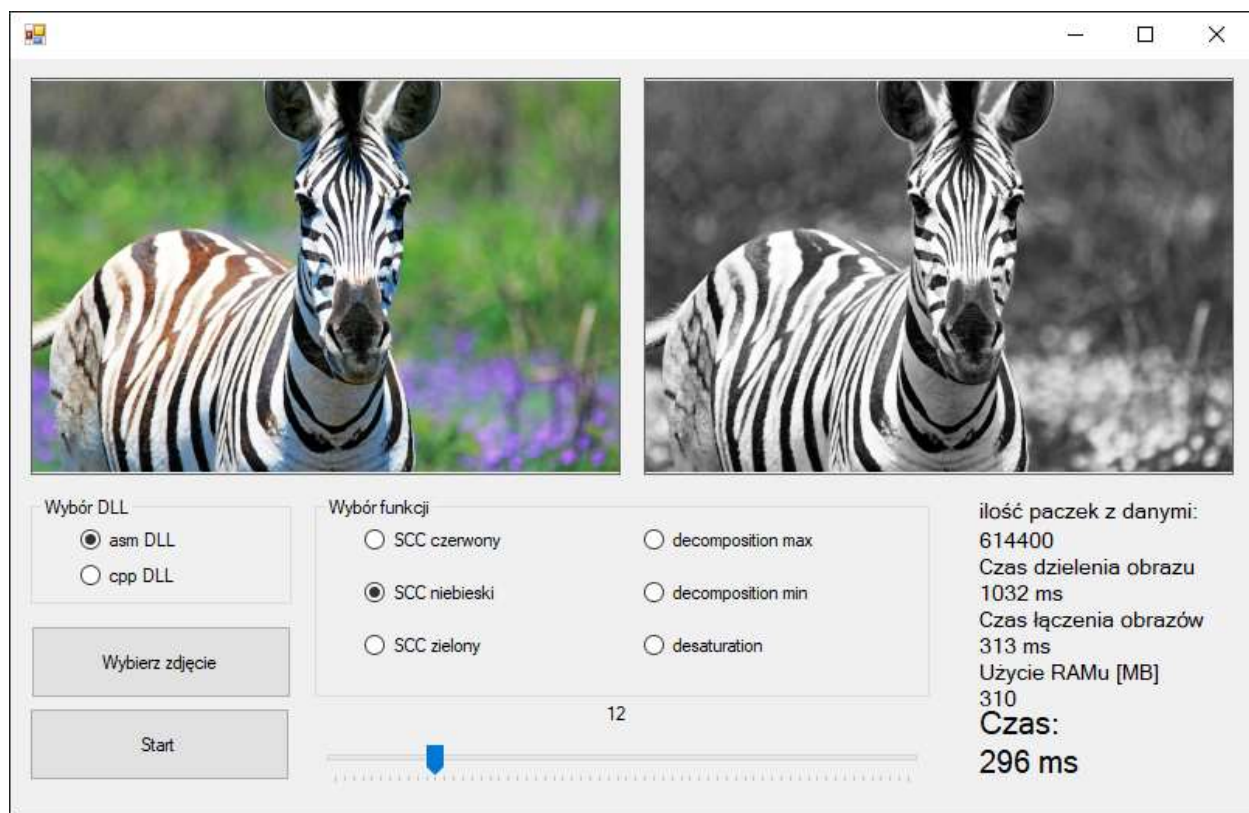
```
public void ChangeColorToGrayScale(object data)
{
    int i = ThreadService.GetI();

    while (i < Pixels.Length)
    {
        ProcessingMethod(Pixels.GetRed(i), Pixels.GetGreen(i), Pixels.GetBlue(i));
        i = ThreadService.GetI();
    }
}
```

Wybrany fragment kodu jest wykonywany równolegle przez zdefiniowaną przez użytkownika ilość wątków. Polega na iterowaniu po kolejnych nieprzetworzonych paczkach obrazu aż do wyczerpania ich ilości. Całość jest wydzielonym w osobnej klasie fragmentem realizowanym w języku c#.

Realizowanie tej części sprawozdania pozwoliło na usunięcie nadmiarowości kodu w postaci pętli for na sztywno ustawionej na jedną iterację. Wnioski: sumienna praca nad dokumentacją pozwala na lepsze zrozumienie własnego kodu i znalezienie błędów.

# Pokazanie działania programu



# Wnioski

Efekt końcowy projektu spełnia jego założenia – możliwa jest zamiana obrazu kolorowego na czarno-biały z wykorzystaniem bibliotek napisanych w języku assembler i c++. Wydajność programu jest ograniczona ze względu na jednowątkowe przygotowywanie i sklejanie otrzymanego programu. Funkcję tę zostaną w przyszłości rozwinięte w ramach indywidualnej pracy z w środowisku .NET. Największą częścią pod względem czasowym okazała się konfiguracja bibliotek dll i ich podpięcie. Problemem była konfiguracja środowiska Visual Studio. Całość projektu upewniła mnie w założeniu, że praca z GITem jest dobrą praktyką i pomaga w zarządzaniu kodem, tworzeniu backupu oraz rozwijaniu projektów niepewnych kierunkach bez ponoszenia konsekwencji utraty wartościowego kodu. Zdecydowałem się również na próbę stworzenia menagera wątków zamiast realizacji kodu w oparciu gotowe rozwiązania. Zdecydowanie utrudniło to moją pracę jednak osiągnięty efekt dydaktyczny był warty wysiłku.

## Wnioski po rozszerzeniu o projekt na PK4

Rozwijanie projektu w ramach przedmiotu programowanie komputerów 4 pozwoliło mi zweryfikować swoje założenie co do użycia wstrzykiwania zależności. Jest to całkiem ciekawa metoda pracy jednak w większości przypadków w tym projekcie jest całkowicie nieuzasadniona jednak ze względu na chęć „zabawy kodem” zdecydowałem się nie wycofywać z początkowej koncepcji.

Dużej modyfikacji uległa logika głównego okna aplikacji przez próbę trzymania SRP.

Utworzone zostały dodatkowe klasy \*Helper, a cały kod został podzielony pod względem funkcjonalności. Do projektu została dołożona funkcjonalność logów ze względu na chęć zrozumienia zagadnienia wyrażeń regularnych. Praca z gitem była przydatna ze względu na częste błędne ścieżki rozwoju kodu.