Projekt zaliczeniowy

**Modelowanie Procesów Dyskretnych**

**Algorytm McNaughtona**

Michał Gucwa

113956

**Spis treści**

[1. Algorytm 2](#_Toc516772504)

[2. Wymagania 3](#_Toc516772505)

[3. Program 4](#_Toc516772506)

[3.1. Dane wejściowe 5](#_Toc516772507)

[3.2. Wykres 6](#_Toc516772508)

[3.3. Zapis i odczyt danych 7](#_Toc516772509)

[4. Testy 8](#_Toc516772510)

# Algorytm

Algorytm McNaughotana działa z zadaniami, które można przerwać i podzielić pomiędzy maszynami. Zasada działania algorytmu jest następująca (gdzie m - liczba maszyn, p – czas zadania, n – liczba zadań):

* Wyznaczyć wartość: = { }, Rozpocząć wykonywanie dowolnego zadania na dowolnym procesorze w chwili t = 0.
* Wybrać dowolne nieuszeregowane jeszcze zadanie i rozpocząć jego wykonywanie na tej samej maszynie w chwili zakończenia wykonywania poprzedniego zadania. Powtarzać ten krok do chwili, gdy wszystkie zadania zostaną uszeregowane lub t = .
* Część zadania pozostającą do wykonania po osiągnięciu t = przydzielić do innej wolnej maszyny, rozpoczynając jej wykonanie od chwili t=0. Wykonać krok 2.

Przykład:

Liczba maszyn – 3.

Liczba zadań – 5.

Zadania:

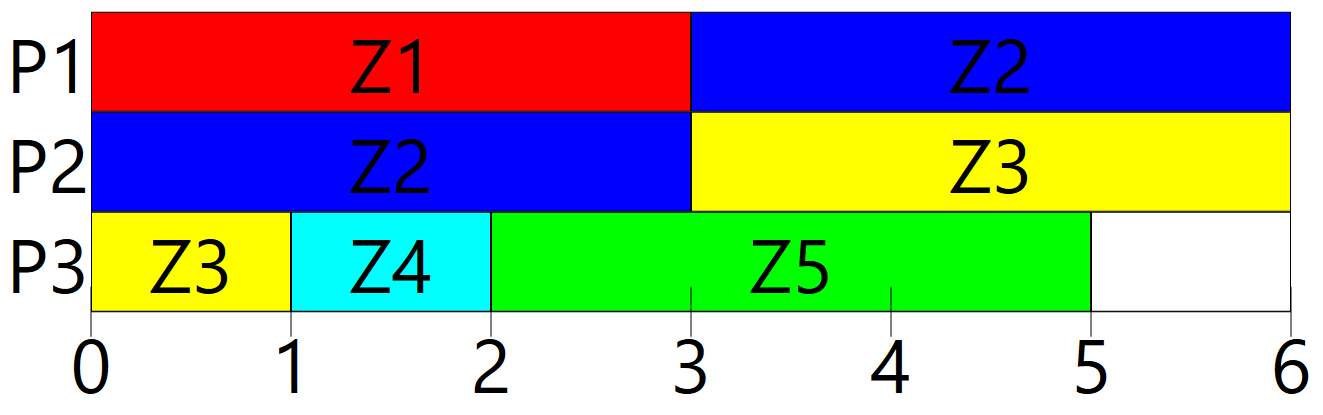
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Zadanie | Z1 | Z2 | Z3 | Z4 | Z5 |
| Czas wykonania | 3 | 6 | 4 | 1 | 3 |

Łączna długość wykonywania zadań:

Najdłuższe zadanie:

=

Wynik:



# Wymagania

Aplikacja została napisana w Visual Studio 2017, w języku C# 6.0. Do uruchomienia aplikacji niezbędne jest posiadanie oprogramowania .NET Framework w wersji minimum 4.5. Do stworzenia interfejsu aplikacji użyłem technologii WPF (Windows Presentation Foundation).

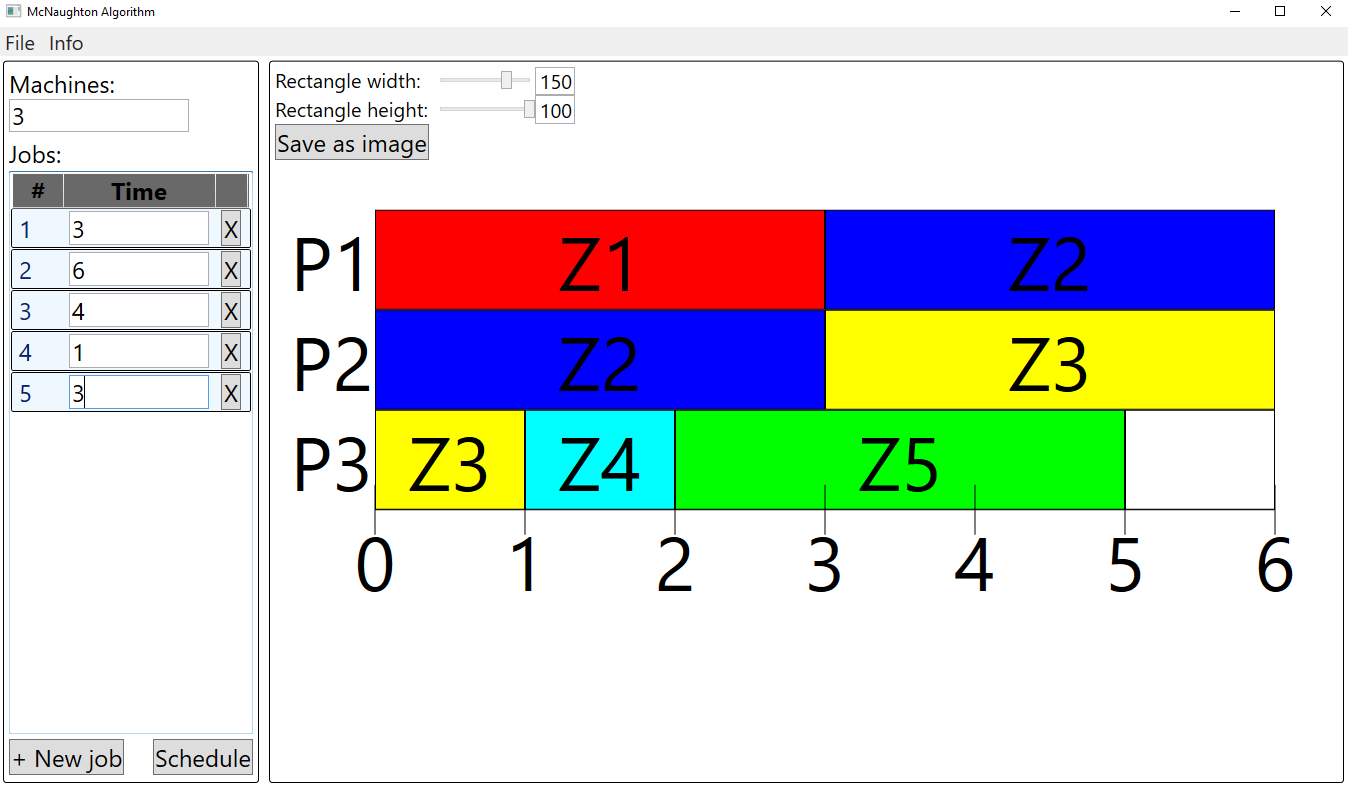
Platforma testowa:

* System operacyjny: Windows 10 x64 (1709)
* .NET Framework 4.7
* Procesor: Intel Core i5-7600K 3,8 GHz
* Pamięć RAM: 16GB



# Program

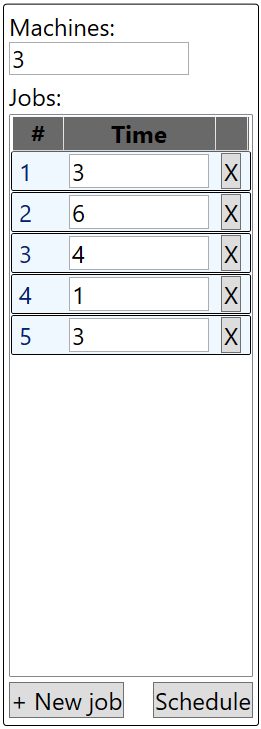
Aplikacja posiada prosty i intuicyjny interfejs. Poniższy obrazek przedstawia główne okno programu:



Rysunek 1. Główne okno programu.

## Dane wejściowe

W lewej części programu widzimy ramkę, w której podajemy dane wejściowe dla algorytmu:

*  Pole *Machines* określa ile maszyn ma być użytych w algorytmie.
* Lista *Jobs* zawiera zadania wraz z długością wykonywania każdego z nich. Kolumna *#* opisuje numer zadania a kolumna *Time* czas wykonania zadania.

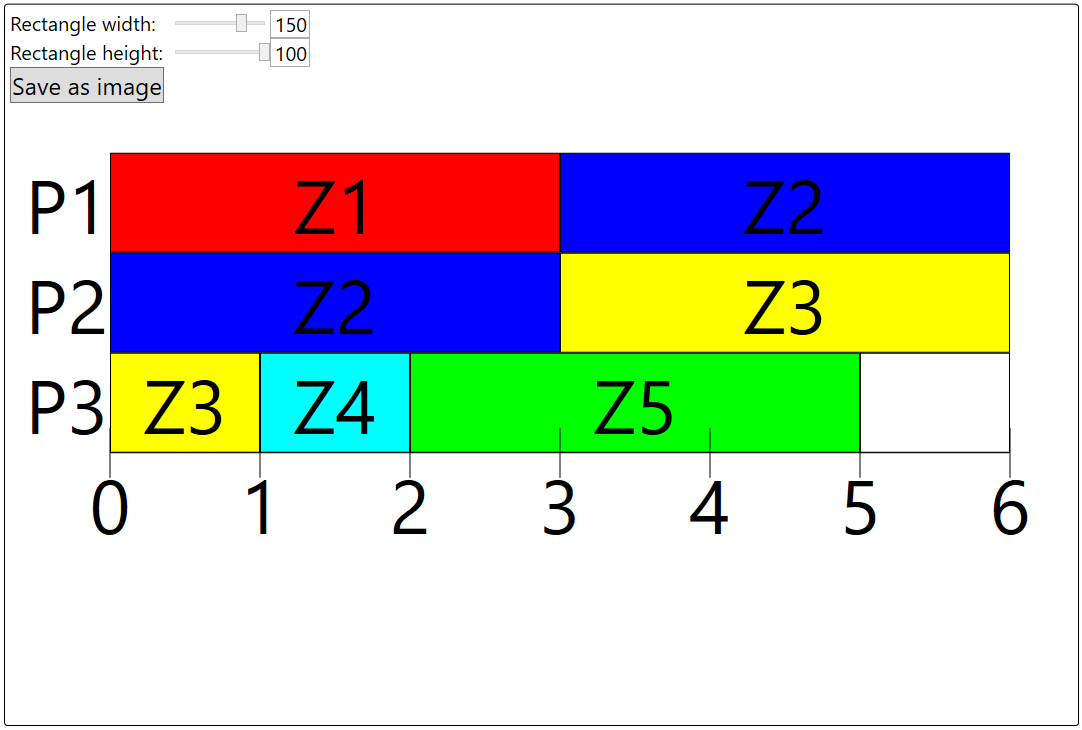
Przyciskiem *X* możemy usunąć dane zadanie, a przyciskiem *+ New Job* możemy dodać nowe zadanie do listy.

Przycisk *Schedule* uruchamia algorytm i rysuje wykres Ganta.

Rysunek 2. Dane wejściowe

## Wykres

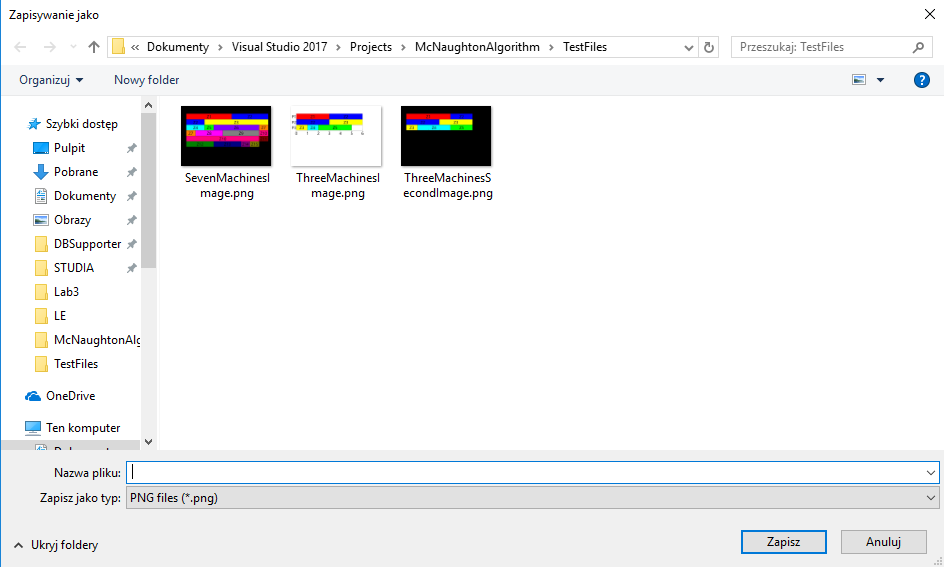
W prawej części aplikacji znajduje się miejsce, w którym zostanie utworzony wykres Ganta dla zadanych danych wejściowych:



Rysunek 3. Wykres Ganta

Suwakami *Rectangle width* ora *Rectangle height* możemy manipulować wielkością wykresu. Wartości suwaków odpowiadają kolejno szerokości oraz wysokości pojedynczego prostokąta, reprezentującego jednostkę czasu.

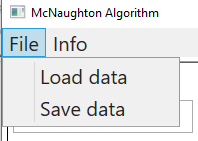
Przycisk *Save as image* umożliwia zapisanie wykresu do pliku o rozszerzeniu \*.png. Po jego naciśnięciu pojawi się odpowiednie okno dialogowe umożliwiające wybranie nazwy pliku i jego położenia na dysku:



Rysunek 4. Zapisywanie wykresu do pliku

## Zapis i odczyt danych

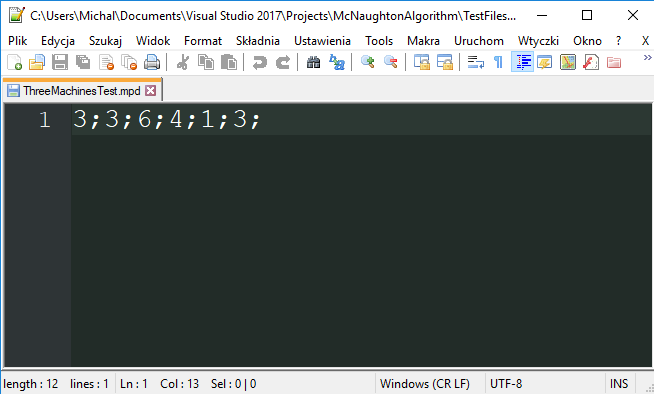
Aplikacja umożliwia zapisanie wprowadzonych danych do pliku oraz ich odczyt. Aby zapisać lub wczytać dane z pliku należy wybrać z menu programu (znajdującego się w górnej części programu) pozycję *File* a następnie interesującą nas opcję:

- *Load data* umożliwia wczytanie danych

- *Save data* umożliwia zapis danych do pliku

Rysunek 5. Menu programu

Po wybraniu interesującej nas opcji pojawi się okienko dialogowe umożliwiające zapis/odczyt danych z pliku. Pliki obsługiwane przez program mają rozszerzenie \*.mpd a ich struktura jest następująca:

Plik składa się z liczb oddzielonych od siebie średnikiem ‘’;’’.

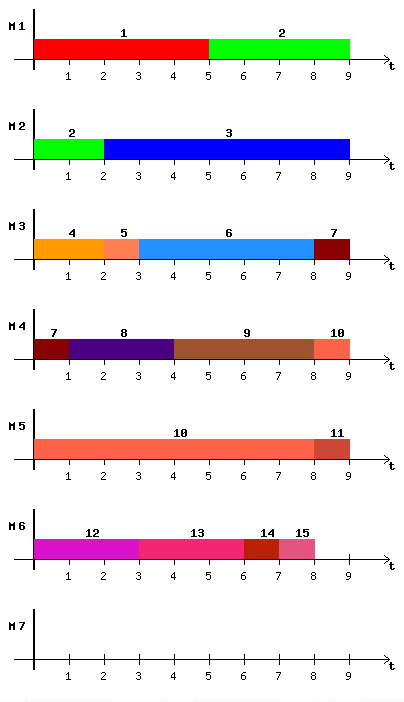
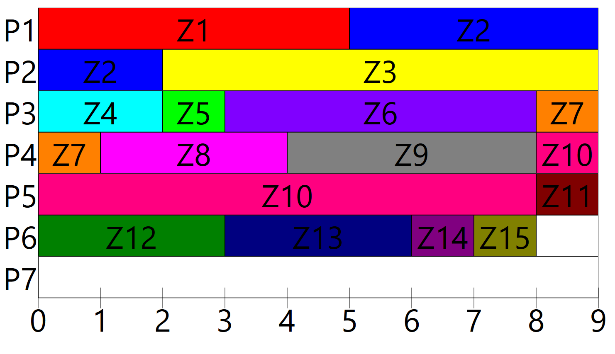
Rysunek 6. Przykładowy plik .mpd otworzony w edytorze tekstowym

Pierwsza liczba w ciągu określa ilość maszyn użytych w algorytmie a kolejne liczby określają czasy poszczególnych zadań. Na przykładowym obrazku mamy więc do czynienia z 3 maszynami i 5 zadaniami o czasach: 3, 6, 4, 1, 3.

# Testy

Aplikacja została przetestowana na kilku zestawach testowych. Poprawność wykonanych wykresów została porównana z wynikami na stronie poleconej przez prowadzącego: <http://kkapd.f11.com.pl/zsw/McNaughton/Mc_Naughton.php>

Wszystkie testy przebiegły poprawnie. Aby ułatwić tworzenie wykresów, w mojej aplikacji ograniczyłem się do zadań o całkowitych czasach wykonania.



Rysunek 7. Test - wynik mojego programu

Rysunek 8. Test - wynik programu internetowego