

# Sterowanie Procesami Dyskretnymi

<b>Kierunek</b> <i>Automatyka i Robotyka</i>	<b>Termin</b> <i>Wtorek 9:15</i>
<b>Skład grupy</b> <i>235538 Wojciech Kończalski, 241594 Maciej Kowalewski</i>	<b>Problem</b> <i>FSP</i>
<b>Prowadzący</b> <i>Mgr inż. Radosław Idzikowski</i>	<b>data</b> <i>5 maja 2020</i>



## RAPORT 1

### 1 Kod źródłowy

Dostępny na repozytorium GitHub:  
[https://github.com/pwr235538/ProjektySPD/blob/master/SPD\\_Lab\\_FSP/LabFSP1.cs](https://github.com/pwr235538/ProjektySPD/blob/master/SPD_Lab_FSP/LabFSP1.cs)

### 2 Wstęp

Celem zadania było zaimplementowanie algorytmu wyznaczania czasu wykonywania wielomaszynowego problemu przepływowego (Flow Shop Problem). Ponadto należało zaimplementować rozwiązania pozwalające na wyznaczanie optymalnej kolejności wykonywania zadań lub rozwiązań bliskich optymalnym - jednym z takich rozwiązań jest Algorytm Johnsona.

Problem składa się z  $N$  zadań wykonywanych na  $M$  maszynach. Dane do wykonania zadania zostały wczytane z 6 plików: data001, data002, data003, data004, data005, data006.

### 3 Wyniki

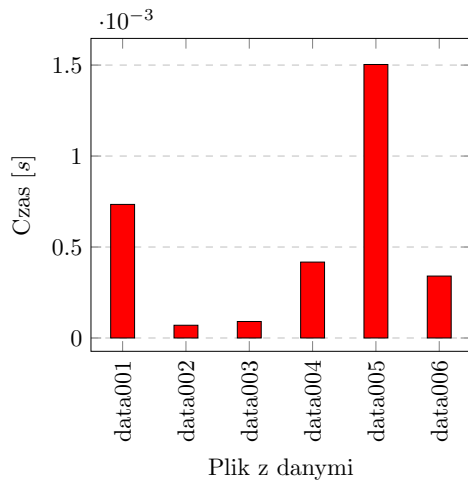
Wyniki badań przedstawiono w tabelach i na wykresach poniżej.

Tabela 1: Cmax, PRD oraz czas obliczeń oraz PRD w zależności od pliku wejściowego

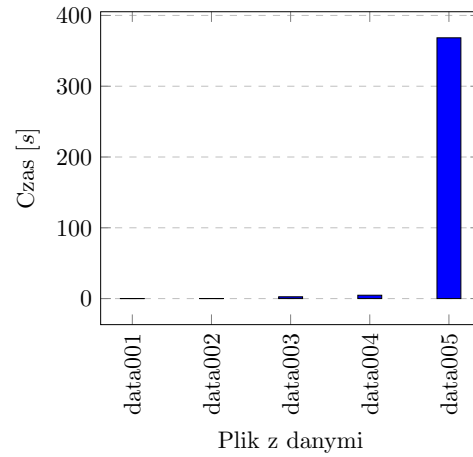
Parametr	data001	data002	data003	data004	data005	data006
<i>Cmax_oryginal</i>	412	728	591	827	683	886
<i>Cmax_johnson</i>	412	552	591	728	654	780
<i>Cmax_opt</i>	412	546	591	727	654	b.d.
<i>PRD_oryginal</i>	0%	33.3%	0%	13.8%	4.4%	b.d.
<i>PRD_johnson</i>	0%	1.1%	0%	0.1%	0%	b.d.
<i>Czas_johnson[s]</i>	0.0007342	0.0000701	0.0000907	0.0004172	0.0015032	0.0003404
<i>Czas_opt[s]</i>	0.0275121	0.0452592	2.5350658	4.7372141	368.3943606	b.d.

Tabela 2: Wyznaczona kolejność wykonywania zadań (indeksując od zera)

Plik	Johnson	Brute force (opt)
data001	0 4 2 6 5 1 3 7	0 1 2 3 4 5 6 7
data002	6 0 1 2 4 7 3 5	6 1 2 3 4 5 0 7
data003	0 7 1 9 8 4 3 5 2 6	0 1 2 3 4 5 6 7 8 9
data004	7 3 9 0 4 2 5 1 8 6	7 1 2 3 4 5 6 0 8 9
data005	7 1 3 10 4 5 11 8 6 9 0 2	7 1 2 3 4 5 6 0 8 9 10 11
data006	5 6 4 3 8 7 10 0 11 9 2 1	b.d.



(a) Algorytm Johnsona



(b) Przegląd zupełny (brute force)

Rysunek 1: Czas znajdowania rozwiązania

## 4 Wnioski

Algorytm Johnsona jest bardzo efektywnym rozwiązaniem. Jego zastosowanie wiązało się z PRD rzędu: 0%, 1.1%, 0%, 0.1%, 0%. Czas znalezienia rozwiązania mieścił się za każdym razem w zakresie 2 milisekund.

Co ciekawe dla danych z pliku data005.txt algorytm Johnsona znalazł rozwiązanie optymalne, a zrobił to w czasie około 1.5 milisekund, gdzie na przegląd zupełny potrzebne było ponad 6 minut. Po przykładzie widać również, że istnieje więcej niż 1 rozwiązanie optymalne. Właśnie w tym przypadku algorytm Johnsona wyznaczył permutację (7, 1, 3, 10, 4, 5, 11, 8, 6, 9, 0, 2), a przy zastosowaniu przeglądu zupełnego wszystkich rozwiązań wyznaczono permutację (7, 1, 2, 3, 4, 5, 6, 0, 8, 9, 10, 11). W oby przypadkach  $C_{max}$  wynosi 654.