



"Implementacja Algorytmu Brownna-Łomickiego"

Projekt Modelowanie Procesów Dyskretnych

Michał Oleś
120271
Informatyka Stosowana Stopień 2
1 rok 1 semestr

Spis treści

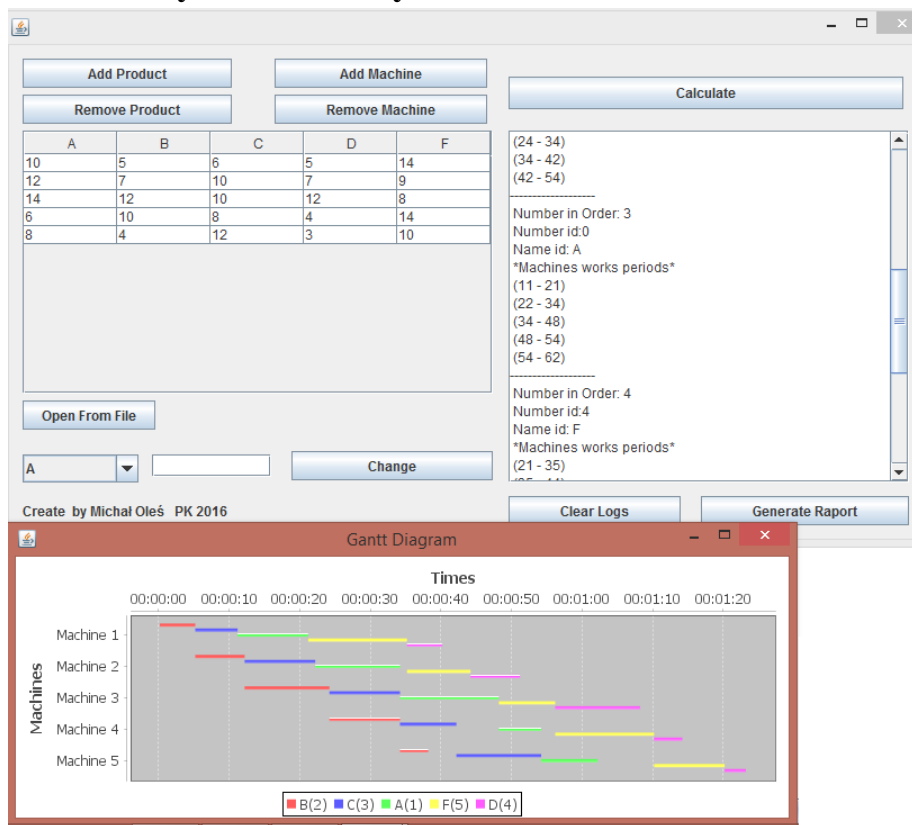
| | |
|---|---|
| 1.Zadanie projektowe..... | 2 |
| 2.Graficzny interface użytkownika..... | 2 |
| 3.Charakterystyka środowiska oraz opis technologii..... | 3 |
| 4. Opis Algorytmu..... | 4 |
| 5.Przeprowadzone testy..... | 5 |
| 6.Możliwy rozwój aplikacji..... | 5 |
| 7.Wnioski..... | 5 |
| 8.Bibliografia..... | 5 |

1.Zadanie projektowe

Projekt miał na celu zaimplementowanie algorytmu Browna-Łomickiego wraz z graficznym interfacem użytkownika.

Aplikacja miała umożliwiać wczytywanie danych z pliku oraz generować raport z wykresem Gantta.

2. Graficzny interfejs użytkownika



Aplikacja umożliwia zmianę zarówno liczby maszyn jak i ilość zadań. Pozwala ręcznie wprowadzać dane oraz wczytać je z pliku (Open From File). Umożliwia ponadto edycje etykiet produktów.

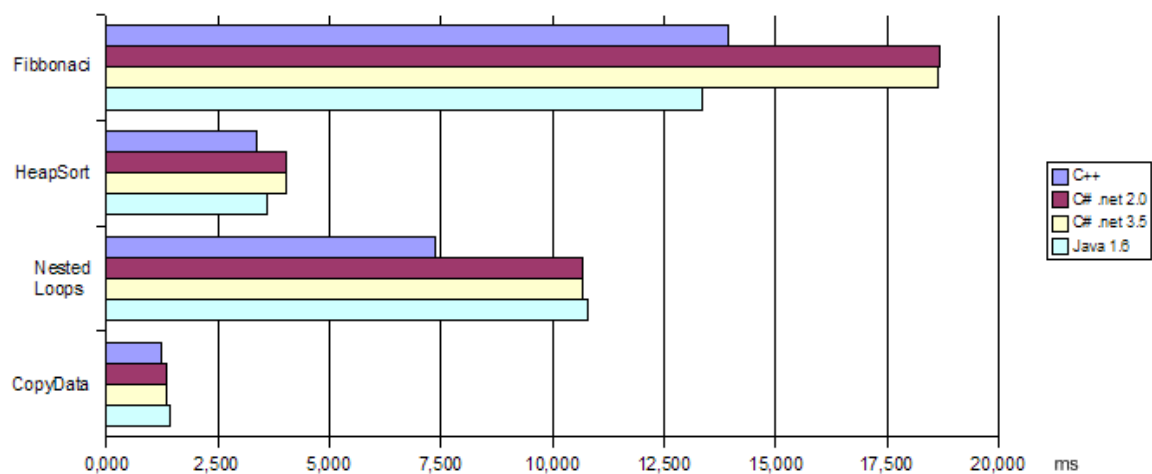
Aby zacząć obliczenia należy wcisnąć przycisk Calculate. Pojawią się wyliczenia oraz w nowym oknie wykres Gantta.

Aby wygenerować raport należy wcisnąć Generate Raport. Raport otworzy się w przeglądarce WWW.

3. Charakterystyka środowiska oraz opis technologii.

Do zaimplementowania algorytmu wykorzystałem język Java z uwagi na jego dużą przenoszalność aplikacji oraz bogatą bazę bibliotek.

Język Java zapewnia prawie taką samą szybkość obliczeń jak język C++ tak więc można go wykorzystać do bardzo dużych zadań np. 50 produktów 50 maszyn.



Do generowania wykresu Gantta wykorzystałem darmową bibliotekę jFreeChart.

4. Opis Algorytmu

I-zbiór numerów wyrobów $I=\{1..m\}$

m-liczba wyrobów

n -liczba maszyn

W-dowolny podzbiór zbioru I

$T(i,w)$ - czas wykonania zbioru W na i-tym stanowisku

Algorytm znajduje iteracyjnie kolejne najbardziej optymalne do wykonania sekwencje wykonywania produktów $[\min(W\{\dots\})]$.

gdzie

$i \in \{1..n\}$

$W\{\dots\}$ to $\max (g_1 \dots g_n)$

$$g_i = T(i, w) + \sum_{j \neq w} t_{ij} + \min_{j \neq w} \sum_{k=i+1}^n t_{kj}$$

Po znalezieniu najkorzystniejszego do wykonania produktu uwzględniamy go w następnych krokach,

Przykład.

Jeśli mamy zbiór produktów 1,2,3,4.

To najpierw obliczamy $W[1], W[2], W[3], W[4]$. Jeśli najmniejszą wartość ma np. 3 to w następnym kroku liczymy $W[3,1], W[3,2], W[3,4]$ i tak dalej do wyczerpania elementów.

Dla n musimy wykonać n-1 wykonań tego algorytmu.

Wartość W ostatniej iteracji to całkowity czas wykonania partii wyrobów.

5.Przeprowadzone testy

Do klasy implementującej algorytm zostały napisane testy jednostkowe sprawdzające poprawność obliczeń.

Dane to testu:

| n | 1 | 2 | 3 | 4 | 5 |
|---|----|----|----|----|----|
| 1 | 10 | 5 | 6 | 5 | 14 |
| 2 | 12 | 7 | 10 | 7 | 9 |
| 3 | 14 | 12 | 10 | 12 | 8 |
| 4 | 6 | 10 | 8 | 4 | 14 |
| 5 | 8 | 4 | 12 | 3 | 10 |

Wynik testu: 84

Wynik jest zgodny z wynikiem zawartym w źródle 1 oraz z moimi ręcznymi obliczeniami.

6.Możliwy rozwój aplikacji

Aplikacja napisana jest w sposób umożliwiający bezproblemową implementację algorytmu z innym GUI graficznym oraz z inną biblioteką generującą wykres Gantta.

Algorytm można optymalizować za pomocą algorytmów do tego stworzonych.

Z uwagi na fakt, że kod aplikacji został udostępniony: <https://github.com/mikoxp/pk-BrownLomicki>

istnieje możliwość, że zostanie udoskonalona lub stanie się komponentem większej całości.

7.Wnioski

Rozwiązanie problemu metodą Browna-Łomickiego daje tak dobre wyniki, że można wykorzystywać ją do optymalizacji w nawet największych zakładach produkcyjnych.

Fakt że może dawać wyniki nie do końca optymalne rekompensuje małą złożoność obliczeniową algorytmu.

8.Bibliografia

1.Bogusław Filipowicz "Badania Operacyjne Wybrane metody obliczeniowe i algorytmy część 1" Kraków 1997