

**Zintegrowany Program Rozwoju
Akademii Górniczo-Hutniczej w Krakowie**
Nr umowy: POWR.03.05.00-00-Z307/17

Instrukcja do ćwiczeń laboratoryjnych

Nazwa przedmiotu	Teoria Współbieżności
Numer ćwiczenia	4
Temat ćwiczenia	Zastosowaniem teorii śladów do szeregowania wątków współbieżnej eliminacji Gaussa

Poziom studiów	I stopień
Kierunek	Informatyka
Forma studiów	Stacjonarne
Semestr	5

dr inż. Maciej Woźniak



Wydział Informatyki, Elektroniki i Telekomunikacji
Kraków, 2019

1. Cel ćwiczenia

Celem ćwiczenia jest zapoznanie studentów z zastosowaniem teorii śladów do szeregowania wątków z zastosowaniem w klasycznych algorytmach algebry liniowej. Jako przykład omawiany jest algorytm współbieżnej eliminacji Gaussa. Ćwiczenie zakłada znajomość podstaw programowania współbieżnego oraz podstaw z algebry liniowej (poznanych na pierwszym semestrze studiów).

Celem ćwiczenia jest zapoznanie studentów z zastosowaniem teorii śladów do szeregowania wątków. Kolejnym celem ćwiczenia jest zaprezentowanie studentów z zastosowaniem modelu gramatyk grafowych do reprezentacji wątków współbieżnych, gdzie wykonanie produkcji w gramatyce grafowej oznacza wykonania wątku. Jako przykład rozważony jest problem generacji jednowymiarowego ciągu elementów z pomocą gramatyki grafowej. Kolejnym celem ćwiczenia jest zapoznanie studentów z technikami implementowania wątków w Javie. W szczególności studenci zapoznani zostaną z implementacją schedulera szeregującego wątki Javy zgodnie z klasami Foaty uzyskanymi na podstawie analizy z użyciem teorii śladów.

2. Wprowadzenie teoretyczne

Teoria śladów [1, 2] służy głównie do modelowania zachowania systemów współbieżnych. Opisuje zależności pomiędzy elementami predefiniowanego zbioru niepodzielnych zadań obliczeniowych. Podstawowym pojęciem teorii śladów jest alfabet będący skończonym zbiorem niepodzielnych zadań obliczeniowych (tasków) - to znaczy niepodzielnych i składających się jedynie z instrukcji wykonywanych sekwencyjnie. Teoria śladów jest szeroko stosowanym formalizmem pozwalającym wyprowadzić optymalne zrównoleglenie dla zadanej grupy tasków.

W trakcie ćwiczenia studenci:

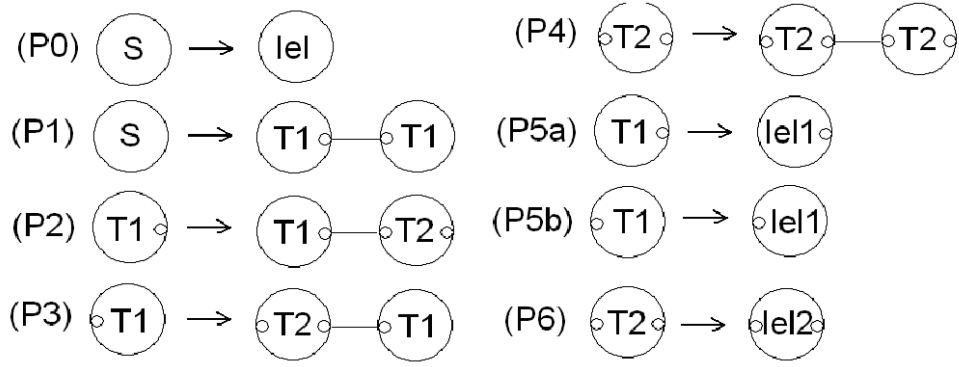
- Definiują alfabet w sensie teorii śladów dla zadanego problemu obliczeniowego
- Definiują relacje pomiędzy elementami alfabetu
- Obliczają postać normalną Foaty oraz budują graf Diekerta
- Wyprowadzają formalny model równoległości zadanego problemu obliczeniowego
- Implementują wyprowadzony wyżej model równoległy

3. Plan ćwiczenia

Dla danego zbioru produkcji w gramatyce grafowej, odpowiedzialnego za generację jednowymiarowego ciągu elementów.

oraz przykładowego wywodu w gramatyce (gdzie kolorem czerwonym zaznaczono miejsce zastosowania kolejnej produkcji)

$$(P1)-(P2)-(P3)-(P5a)-(P6)-(P5b)-(P6) \quad (1)$$



Rysunek 1. Gramatyka grafowa obowiązująca na ćwiczeniu 4

$$\begin{aligned}
 & S_{(P1)} \rightarrow T1_{(P2)}-T1 \rightarrow T1-T2-T1_{(P3)} \rightarrow T1_{(P5a)}-T2-T2-T1 \rightarrow \\
 & \rightarrow |e|_1-T2_{(P6)}-T2-T1 \rightarrow |e|_1-|e|_2-T2-T1_{(P5b)} \rightarrow |e|_1-|e|_2-T2_{(P6)}-|e|_1 \rightarrow \\
 & \rightarrow |e|_1-|e|_2-|e|_2-|e|_1
 \end{aligned}
 \tag{2}$$

opracowujemy alfabet w sensie teorii śladów reprezentujący wykonanie poszczególnych tasków – produkcji

$$A = \{P1, P2, P3, P5a, P6, P5b, P6\} \tag{3}$$

Następnie na podstawie analizy zależności pomiędzy produkcjami opracowywana jest relacja zależności (plus symetria)

$$D = \text{sym}\{\{(P1, P2), (P1, P3), (P2, P5a), (P2, P6), (P3, P5b), (P3, P6)\}^+ \cup I_A\} \tag{4}$$

Dla słowo w alfabecie reprezentującego wywód w gramatyce

$$P1, P2, P3, P5a, P6, P5b, P6 \tag{5}$$

opracowywana jest postać normalna Foaty

$$FNF = [P1][P2, P3][P5a, P6, P5b, P6] \tag{6}$$

Bazując na opracowanej postaci normalnej Foaty oraz przedstawionym zbiorze produkcji tworzony jest program w języku JAVA generujący wątki odpowiadające produkcjom w gramatyce grafowej oraz szeregujący je zgodnie z klasami Foaty. Na zajęciach omawiane są poszczególne elementy programu.

Pełny kod programu znajduje się w repozytorium.

<https://github.com/macwozni/1DMeshParallel>

Po omówieniu wszystkich elementów, zadawany jest inny wywód w gramatyce (inne słowo w sensie alfabetu teorii śladów) i studenci proszeni są o wygenerowanie nowych klas Foaty, oraz napisanie nowego mechanizmu szeregowania.

Proszę wykonać:

1. Zdefiniować alfabet w sensie teorii śladów dla nowego wywodu
2. Zdefiniować relację zależności D
3. Obliczyć klasy Foaty
4. Zaimplementować scheduler dla nowych klas Foaty

4. Sposób oceny

Studenci opracowują sprawozdanie, które powinno zawierać wszystkie elementy opisane w podpunktach 1-4 oraz kod programu. Zadanie oceniane jest następująco:

- Poprawność formalna przedstawionego rozwiązania dla każdego zadania to 15% (razem $4 \times 15\% = 60\%$),
- Redakcja sprawozdania 10%,
- Poprawność implementacji 30%

Literatura

- [1] V. Diekert, Y. M'etivier - Partial commutation and traces, [w:] Handbook of Formal Languages, Springer, 1997, str. 457-553
- [2] Diekert V., Rozemberg G. - The book of traces, 1995
- [3] Dowolny podręcznik z algebry liniowej lub metod numerycznych z zakresu sekwencyjnego algorytmu eliminacji Gaussa
- [4] Wykład z przedmiotu "Teoria współbieżności" – rozdział dotyczący teorii śladów
- [5] Bruce Eckel, "Thinking in Java" - rozdział o wątkach