

Sprawozdanie
„Klasy Foaty dla liczników”
Teoria Współbieżności

gr. śr 17:50

17.11.2021

1. Cel ćwiczeń

Zadanie polegało na implementację programu, który na wejściu otrzymywał alfabet zdarzeń oraz słowo z tego alfabetu. Zdarzenia polegały na zmianie wartości licznika. Program na podstawie wejścia ma tworzyć relacje zależności i niezależności oraz obliczać postać normalną Foaty wraz z wyrysowaniem grafu Dickerta. Program miał być napisany w języku funkcyjnym, niestety program napisany przeze mnie nie spełnia tego założenia.

2. Dane wejściowe

Dane wejściowe to alfabet, zestaw transakcji i słowo z tego języka. Przyjąłem następujący format danych:

- 1 linijka: alfabet – litery oddzielone białym znakiem;
- 2 do $n+1$ linijki: transakcje – w każdej linijce osobna transakcja zapisana w postaci: $x=x+y$ gdzie n to długość alfabetu;
- $n+2$ linijka: słowo z języka

```
a b c d
x=x+y
y=y+2z
x=3x+z
z=y-z
baadbc
```

Tekst 1: Przykładowe dane wejściowe.

Program wczytuje plik i zapisuje dane w dwóch listach alphabet (litery alfabetu) i functions (krotka zawierająca zmienną i równanie) oraz stringu zawierającym słowo.

3. Uruchomienie programu

Program został napisany w języku python. Wywołanie komendy *python zadanie5.py* powoduje uruchomienie programu dla 4 zestawów danych dostarczonych wraz z zadaniem. Gdy jako argument podamy ścieżkę do pliku zawierającą dane w formacie podanym wyżej program uruchomi się dla tych konkretnych danych. Do uruchomienia programu potrzebujemy interpretera pythona oraz zainstalowaną bibliotekę graphviz.

4. Tworzenie relacji

Funkcja *make_sets* tworzy dwa zbiory relacji. Bada ona po kolei każdą z każdą literą sprawdzając czy są zależne. Sprawdzenie polega na określeniu czy zmienna z jednej

funkcji występuje w drugiej i na odwrót. Jeśli zależne dodajemy do zbioru D, jeśli nie to dodajemy do zbioru I. Dodatkowo w D pojawia się relacja identycznościowa.

5. Wyznaczanie FNF

FNF wyznaczam na trzy sposoby.

Pierwszy sposób to analiza słowa i rozpoczynając od początku dodawanie zachłanne do klasy Foaty z zachowaniem, że wszystkie litery muszą być niezależne w danej klasie oraz możemy dodać literę, jeśli wszystkie litery, z którymi jest zależna a występują wcześniej w słowie są już dodane do pewnych klas Foaty.

Drugi sposób to metoda opisana w załączonej książce. Tworzę stosy dla każdej litery z alfabetu i analizując słowo od tyłu wrzucam odpowiednie znaczniki do stosów. Następnie odczytuje klasę Foaty z liter ze szczytów stosów i ściągam odpowiednie znaczniki.

Trzecia metoda to wywoływanie DFSa na zminimalizowanym grafie Dickerta. DFS idąc w dół określa klasę wierzchołka na następną w porównaniu do rodzica. Żeby działało to prawidłowo musimy zapewnić, że dana litera będzie tylko w jednej klasie i będzie to ta najdalsza z wyznaczonych. Realizuję to poprzez sprawdzenie czy litera jest we wcześniejszej klasie Foaty, jeśli tak to usuwam ją stamtąd i dodaje do dalszej. Jeśli natomiast litera jest w dalszej to zatrzymuje DFSa i nie dodaje tej litery i tych które są poniżej w drzewie. Skoro litera jest już gdzieś dalej w klasach Foaty to znaczy, że DFS był już na niej wywoływany i litery pod nią w grafie są już dodane w dalszych klasach Foaty.

6. Minimalizacja Grafu Dickerta

Graf Dickerta buduje jako listę sąsiedztwa. Minimalizacja polega na sprawdzeniu dla każdej krawędzi czy jest potrzebna. Krawędź (u, v) jest niepotrzebna, jeśli wywołując DFS na wierzchołku u jesteśmy w stanie dojść do wierzchołka v w podgrafie, który jest taki jak wyjściowy tylko bez krawędzi (u, v) . W przeciwnym przypadku krawędź jest potrzebna. Minimalizując usuwam krawędzie niepotrzebne.

7. Wizualizacja grafu

Wizualizację tworzę wykorzystując moduł graphviz. Tworzę instancję klasy Digraph i dodaje do niej zminimalizowany graf poprzez zapisanie wierzchołków i krawędzi. Wypisuję graf w postaci dot oraz zapisuję wizualizację.

8. Testy

Przygotowałem 4 zestawy testowe.

I zestaw:

$A = \{a, b, c, d\}$

(a) $x := x + y$

(b) $y := y + 2z$

(c) $x := 3x + z$

(d) $z := y - z$

$\omega = baadcb$

Wyniki:

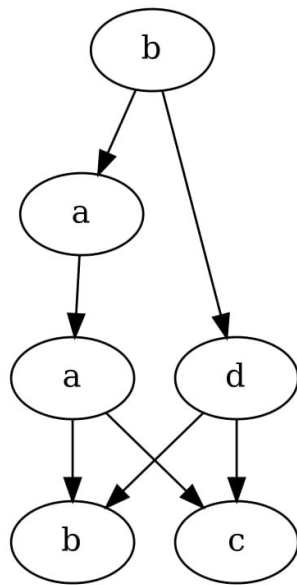
$D = \{(b, a), (c, c), (d, c), (c, d), (d, b), (b, d), (c, a), (b, b), (a, b), (a, a), (a, c), (d, d)\}$

$I = \{(b, c), (a, d), (d, a), (c, b)\}$

$FNF = [b][ad][a][bc]$

```
digraph {
    0 [label=b]
    1 [label=a]
    2 [label=a]
    3 [label=d]
    4 [label=b]
    5 [label=c]
    0 -> 1
    0 -> 3
    1 -> 2
    2 -> 4
    2 -> 5
    3 -> 4
    3 -> 5
}
```

Tekst 2: Graf Dickerta w formacie dot dla zestawu I.



Rysunek 1: Graf Dickerta dla zestawu I.

II zestaw:

$A = \{a, b, c, d, e, f\}$

(a) $x := x + 1$

(b) $y := y + 2z$

(c) $x := 3x + z$

(d) $w := w + v$

(e) $z := y - z$

(f) $v := x + v$

$\omega = \text{acdcfbbe}$

Wyniki:

$D = \{(c, c), (a, f), (f, c), (f, d), (d, f), (f, a), (e, c), (e, b), (e, e), (f, f), (c, a), (c, e), (b, b), (b, e), (a, a), (a, c), (c, f), (d, d)\}$

$I = \{(b, a), (b, f), (d, c), (d, a), (e, d), (e, a), (b, c), (b, d), (c, b), (e, f), (c, d), (a, b), (a, e), (a, d), (d, b), (d, e), (f, b), (f, e)\}$

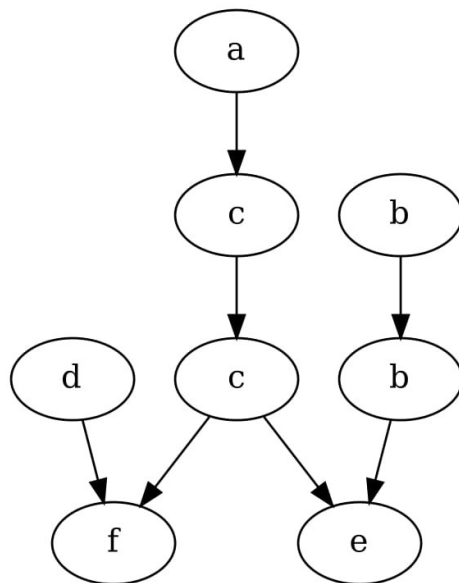
$\text{FNF} = [\text{adb}][\text{cb}][\text{c}][\text{fe}]$

```

digraph {
  0 [label=a]
  1 [label=c]
  2 [label=d]
  3 [label=c]
  4 [label=f]
  5 [label=b]
  6 [label=b]
  7 [label=e]
  0 -> 1
  1 -> 3
  2 -> 4
  3 -> 4
  3 -> 7
  5 -> 6
  6 -> 7
}

```

Tekst 3: Graf Dickerta w formacie dot dla zestawu II.



Rysunek 2: Graf Dickerta dla zestawu II.

III zestaw:

$A = \{a, b, c, d, e, f, g, h\}$

(a) $x := x + v$

(b) $x := z + y$

(c) $y := x - z$

(d) $y := xy + v$

(e) $z := 3z$

(f) $z := xz + y$

(g) $v := x + z$

(h) $v := v - y$

$\omega = \text{adegfbha}$

Wyniki:

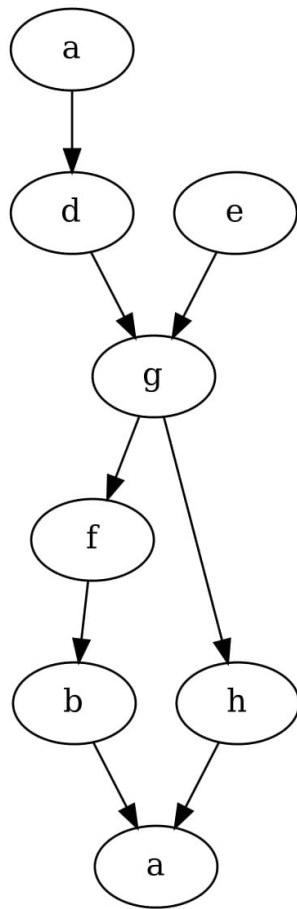
$D = \{(b, f), (d, c), (f, d), (g, f), (e, b), (g, h), (h, c), (b, d), (c, e), (b, b), (g, d), (g, b), (a, c), (c, c), (f, c), (d, g), (e, c), (c, a), (b, c), (c, f), (f, e), (h, g), (a, g), (d, a), (d, f), (c, h), (f, g), (e, e), (d, h), (c, b), (h, a), (e, g), (c, d), (b, e), (a, a), (d, b), (g, e), (d, d), (a, f), (b, g), (f, a), (g, g), (h, h), (a, h), (f, f), (a, b), (h, d), (e, f), (b, a), (a, d), (g, a), (f, b)\}$

$I = \{(e, d), (b, h), (e, a), (f, h), (h, b), (e, h), (h, e), (h, f), (g, c), (a, e), (d, e), (c, g)\}$

$\text{FNF} = [\text{ae}][\text{d}][\text{g}][\text{fh}][\text{b}][\text{a}]$

```
digraph {
    0 [label=a]
    1 [label=d]
    2 [label=e]
    3 [label=g]
    4 [label=f]
    5 [label=b]
    6 [label=h]
    7 [label=a]
    0 -> 1
    1 -> 3
    2 -> 3
    3 -> 4
    3 -> 6
    4 -> 5
    5 -> 7
    6 -> 7
}
```

Tekst 4: Graf Dickerta w formacie dot dla zestawu III.



Rysunek 3: Graf Dickerta dla zestawu III.

IV zestaw:

$A = \{a, b, c, d, e, f\}$

(a) $x := x + 1$

(b) $y := y + 2z$

(c) $x := 3x + z$

(d) $w := w + v$

(e) $z := y - z$

(f) $v := x + v$

$\omega = eabeafbefffeecabbdbd$

Wyniki:

$D = \{(c, c), (a, f), (f, c), (f, d), (d, f), (f, a), (e, c), (e, b), (e, e), (f, f), (c, a), (c, e), (b, b), (b, e), (a, a), (a, c), (c, f), (d, d)\}$

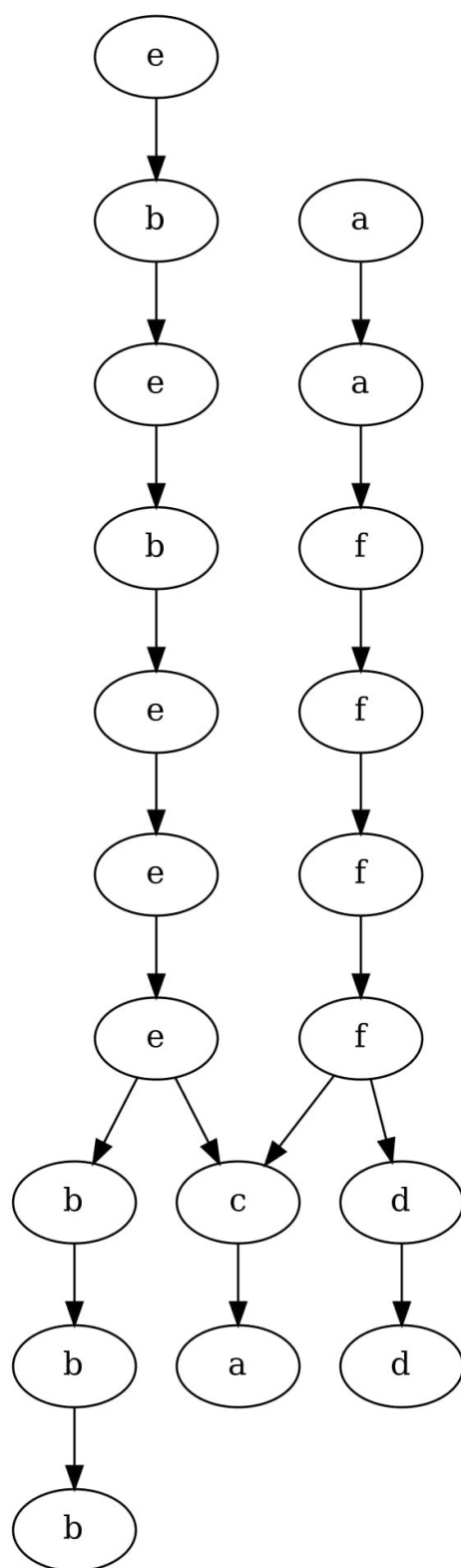
$I = \{(b, a), (b, f), (d, c), (d, a), (e, d), (e, a), (b, c), (b, d), (c, b), (e, f), (c, d), (a, b), (a, e), (a, d), (d, b), (d, e), (f, b), (f, e)\}$
 $FNF = [ea][ba][ef][bf][ef][fe][ed][cbd][ab][b]$

```

digraph {
    0 [label=e]
    1 [label=a]
    2 [label=b]
    3 [label=e]
    4 [label=a]
    5 [label=f]
    6 [label=b]
    7 [label=e]
    8 [label=f]
    9 [label=f]
    10 [label=f]
    11 [label=e]
    12 [label=e]
    13 [label=c]
    14 [label=a]
    15 [label=b]
    16 [label=b]
    17 [label=d]
    18 [label=b]
    19 [label=d]
    0 -> 2
    1 -> 4
    2 -> 3
    3 -> 6
    4 -> 5
    5 -> 8
    6 -> 7
    7 -> 11
    8 -> 9
    9 -> 10
    10 -> 17
    10 -> 13
    11 -> 12
    12 -> 13
    12 -> 15
    13 -> 14
    15 -> 16
    16 -> 18
    17 -> 19
}

```

Tekst 5: Graf Dickerta w formacie dot dla zestawu IV.



Rysunek 4: Graf Dickerta dla zestawu IV.