Analiza Algorytmów 2016/2017
(zadania na laboratorium)

Wybór lidera (do 15 III)

Zadanie 1 — W dowolnym języku programowania zaimplementuj symulator umożliwiający przetestowanie algorytmu wyboru lidera $ELECTION(\vec{P})$ opisanego na pierwszym wykładzie. Niech L będzie zmienną losową oznaczającą liczę slotów potrzebnych do czasu wyboru lidera. Dla scenariuszy: 1) liczba stacji n jest znana, 2) znane jest jedynie górne ograniczenie $u \geq n$, a faktyczna liczba stacji to $n = 2$, $n = u/2$, $n = u$, wykorzystaj symulator aby

a. narysować rozkład empiryczny (histogram) dla zmiennej L ,

b. oszacować $\mathbb{E}[L]$ oraz $\mathbb{V}ar[L]$.

Czy wyniki symulacji są zgodne z wynikami teoretycznymi? **(10p.)**

Zadanie 2 — Za pomocą symulacji potwierdź wynik Lematu 3 z pierwszego wykładu **(10p.)**:

$$Pr[S_{L,u,n}] \geq \lambda \approx 0.579 .$$

Przybliżone zliczanie (do 29 III)

Zadanie 3 — Zaimplementuj algorytm $COUNTING(k, \mathcal{M})$ podany na wykładzie. Dla $k = 2$ i multizbiorów $\mathcal{M}$ zawierających $n = 1, 2, \ldots, 10^4$ różnych elementów narysuj wykres (wystarczą same punkty) mający na osi poziomej prawdziwą wartości n a na osi pionowej stosunek $\hat{n}/n$, gdzie $\hat{n}$ oznacza estymację otrzymaną z algorytmu. Narysuj podobne wykresy dla $k = 2, 3, 10, 100, 400$. Dla uproszczenia możesz przyjąć, że $\mathcal{M}$ nie zawiera powtórzeń i składa się z kolejnych liczb naturalnych. Czy obecność powtórzeń ma wpływ na wartość estymacji $\hat{n}$? **(10p.)**

Zadanie 4 — Dobierz wartość parametru k tak by w 95% przypadków $|\frac{\hat{n}}{n} - 1| < 10\%$. **(5p.)**

Zadanie 5 — Przetestuj działanie algorytmu dla różnych funkcji haszujących $h : S \rightarrow \{0, 1\}^B$ i różnych wartości parametru B. Postaraj się znaleźć funkcję haszującą h dla której wyniki algorytmu są istotnie gorsze. Postaw hipotezę jaka powinna być wartość parametru B w zależności od liczby różnych elementów $|S| = n$, aby wyniki estymacji były istotnie zaburzone. **(10p.)**

Zadanie 6 — (**do 5 IV**) Twoim zadaniem jest porównanie teoretycznych wyników dotyczących koncentracji estymatora $\hat{n}$ wykorzystanego w algorytmie $COUNTING(k, \mathcal{M})$ uzyskanych przez **a)** nierówność Czebyszewa oraz **b)** nierówność Chernoffa, z wynikami symulacji. Dla $n = 1, 2, \ldots 10^4$, $k = 400$ i trzech poziomów istotności $\alpha = 5\%,\ 1\%,\ 0.5\%$ przedstaw na wykresie wartości $\hat{n}/n$ oraz wartości $1 - \delta$ i $1 + \delta$ takie, że **(10p.)**

$$Pr\left[1 - \delta < \frac{\hat{n}}{n} < 1 + \delta\right] > 1 - \alpha\ .$$

Samostabilizacja

Zadanie 7 — (**do 26 IV**) Zaimplementuj symulator algorytmu Mutual Exclusion z wykładu. Dla ustalonego n oznaczającego liczbę procesów w pierścieniu, zweryfikuj, że startując z dowolnej konfiguracji początkowej algorytm przejdzie do legalnej konfiguracji. Jeśli z pewnej konfiguracji można przejść do kilku możliwych konfiguracji w zależności od tego, który proces wykona krok jako pierwszy, każde wykonanie powinno zostać zweryfikowane. Jaka jest największa liczba kroków do czasu osiągnięcia legalnej konfiguracji dla ustalonego n? Dla jakich wartości n możesz uzyskać odpowiedź w sensownym czasie? **Do zdobycia jest** $f(N)$ **punktów**[*].

Zadanie 8 — (**do 10 V**) Rozważmy graf $G = (V, E)$. Dwa wierzchołki $v, w \in V$ nazywamy niezależnymi, jeśli $\{v, w\} \notin E$. Podzbiór $S \subseteq V$ wierzchołków nazywamy niezależnym, jeśli wszystkie jego elementy są parami niezależne. Wzorując się na algorytmie Maximal Matching podanym na wykładzie zaprojektuj, zaimplementuj i przetestuj samo-stabilizujący algorytm znajdujący maksymalny zbiór niezależny w nieskierowanym grafie spójnym[†]. Podaj przekonywujące uzasadnienie poprawności algorytmu (formalny dowód zrobimy na ćwiczeniach). **(15p.)**

Procesy stochastyczne

Zadanie 9 — (**do 17 V**) Dla danego grafu skierowanego o n wierzchołkach definiujemy macierz PageRank jako

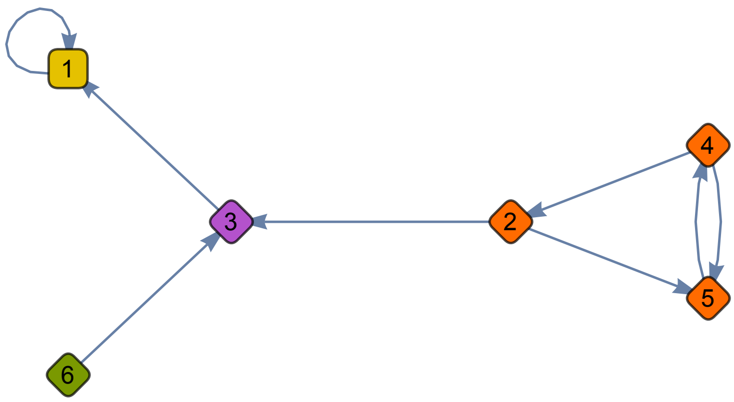
$$P = (1 - \alpha)A + \alpha\frac{1}{n}B,$$

gdzie $\alpha \in [0, 1]$ to tzw. współczynnik tłumienia, A to macierz $n \times n$ przedstawiająca prawdopodobieństwa przejścia między wierzchołkami grafu[‡], a B to macierz $n \times n$ złożona z samych jedynek. Poniższy graf przedstawia schemat połączeń między sześcioma stronami internetowymi. Wyznacz PageRank (czyli rozkład stacjonarny dla łańcucha Markowa związanego z macierzą P) dla $\alpha = 0, 0.15, 0.5, 1$. Co się zmieni jeśli zlikwidujemy połączenie między stanem 2 i 3? Jaki jest cel wprowadzania macierzy B? **(10p.)**

[*] Gdzie N oznacza rozmiar największego pierścienia, który udało Ci się zweryfikować, a funkcja f zostanie ustalona przez prowadzącego w stosownym czasie.

[†] Zobacz: Maximal Independent Set. Algorytmy znajdowania Maximal Independent Set mają wiele zastosowań, np. w algorytmach przydziału częstotliwości w sieciach bezprzewodowych.

[‡] Przyjmujemy, że dla danego wierzchołka prawdopodobieństwo przejścia do każdego z sąsiadów w grafie skierowanym jest takie samo



Zadanie 10 — (do 17 V) Rozważ łańcuch Markowa o zbiorze stanów $\{0, 1, 2, 3\}$ i macierzy przejścia

$$P = \begin{pmatrix} 0 & 3/10 & 1/10 & 3/5 \\ 1/10 & 1/10 & 7/10 & 1/10 \\ 1/10 & 7/10 & 1/10 & 1/10 \\ 9/10 & 1/10 & 0 & 0 \end{pmatrix}.$$

a) Znajdź rozkład stacjonarny $(\pi_0, \pi_1, \pi_2, \pi_3)$ dla tego łańcucha Markowa.

b) Znajdź prawdopodobieństwo znalezienia się w stanie 3 po 32 krokach, jeśli zaczynamy w stanie 0.

c) Znajdź prawdopodobieństwo znalezienia się w stanie 3 po 128 krokach, jeśli łańcuch zaczyna się w stanie wybranym losowo w sposób jednostajny ze wszystkich czterech stanów.

d) Przypuśćmy, że zaczynamy w stanie 0. Jaka jest najmniejsza wartość liczby kroków t, dla której

$$\max_s |P^t_{0,s} - \pi_s| \leq \varepsilon$$

dla $\varepsilon = 1/10, 1/100, 1/1000$.

Kiedy przydaje się możliwość tego rodzaju aproksymacji rozkładu stacjonarnego? **(15p.)**

Zadanie 11 — (do 24 V) Zaimplementuj symulator prostej kolejki z czasem dyskretnym i bez ograniczenia rozmiaru (model z wykładu) dla różnych wartości parametrów λ i μ, takich, że $\lambda > \mu$, $\lambda < \mu$ oraz $\lambda = \mu$. Dla dużej liczby kroków sprawdź, jak często łańcuch przebywa w danym stanie i jaka jest średnia liczba kroków do chwili powrotu do danego stanu. Porównaj swoje wyniki z wynikami teoretycznymi. **(10p.)**

Zadanie 12 — (do 7 VI) Zaimplementuj symulator opisany w Zadaniu 8.26 w książce "Metody probabilistyczne i obliczenia"(Mitzenmacher, Upfal) i wykonaj opisane w tym zdaniu polecenia. Można łatwo uzyskać dostęp do angielskojęzycznej wersji książki: "Probability and Computing: Randomized Algorithms and Probabilistic Analysis". W razie problemów z dostępem chętnie użyczę swój egzemplarz (proszę o maila). Treść Sekcji 8.4.2 (łączenie i rozdzielanie procesów Poissona) omówię jeszcze szybko na najbliższym wykładzie. **(25p.)**

Zadanie 13 — (**do 21 VI**) Wykorzystując funkcje tworzące wyznacz liczbę wywołań linii 6 dla danego n

```
1:  f(int n) {
2:  int s = 0;
3:  if ( n == 0 ) then return 1;
4:  else
5:      for int i = 0; i < n; i++  do
6:          s += f(i);
7:      end for
8:      return s;
9:  end if
10: }
```

(10p.)

Zadanie 14 — (**do 21 VI**) Algorytm otrzymuje na wejściu tablicę długości $n \geq 0$. Jeśli $n \geq 2$ dla każdego $k \in \{1, 2, 3, \ldots, n\}$ algorytm z prawdopodobieństwem 1/2 wywołuje się rekurencyjnie na pewnej losowej „podtablicy” długości k. Wykorzystując funkcje tworzące wyznacz średnią liczbę wywołań algorytmu dla danego n. Zweryfikuj odpowiedź eksperymentalnie. **(10p.)**