Algorytmika
Lista zadań

Jacek Cichoń
WIT, PWr, 2023/24 (semestr letni)

1 Podstawowy model obliczeń

Zadanie 1

Pokaż, że następujący problem:

> Mamy dwa programy P i Q obliczające funkcje z liczb naturalnych w liczby naturalne. Czy dla każdego naturalnego n P(n)=Q(n)?

jest nierozstrzygalny.
Wskazówka: Zredukuj problem $(\forall n)(P(n) = 0)$ do powyższego problemu.

Zadanie 2

Zbiór $A \subseteq \mathbb{N}^k$ nazywamy rekurencyjnie przeliczalnym (RE) jeśli istnieje całkowita funkcja obliczalna $f : \mathbb{N} \to \mathbb{N}^k$ taka, że $A = \{f(n) : n \in \mathbb{N}\}$.

1. Pokaż, że jeśli $A \subseteq \mathbb{N}$ jest taki, że $A \in RE$ oraz $\mathbb{N} \setminus A \in RE$ to A jest zbiorem rekurencyjnym.
2. Pokaż, że jeśli A jest rekurencyjny, to A jest rekurencyjnie przeliczalny.
3. Uzasadnij tezę $STOP \in RE$.

Zadanie 3

Dla ciągów $x, y \in \Sigma^*$ określamy $x \sqsubseteq y \longleftrightarrow (\exists z \in \Sigma^*)(y = xz)$ oraz $x \sqsupseteq y \longleftrightarrow (\exists z \in \Sigma^*)(y = zx)$.

1. Pokaż, że $\sqsubseteq$ jest częściowym porządkiem na Σ^*.
2. Wyraź relację $\sqsupseteq$ za pomocą relacji $\sqsubseteq$ oraz funkcji `reverse` odwracania ciągów.

Zadanie 4

Ustalmy skończony alfabet Σ oraz wzorzec $P[1..n] \in \Sigma^*$. Niech p_n oznacza prawdopodobieństwo tego, że losowy ciąg długości n elementów Σ jest zgodny (w jakimś miejscu) ze wzorcem P. Pokaż, że $\lim_{n\to\infty} p_n = 1$.

Zadanie 5

Ustalmy skończony alfabet Σ. Ustalmy ciąg $A[1..k]$. Niech p_n oznacza prawdopodobieństwo tego, że losowy ciąg X długości n elementów Σ^* zawiera podciąg A, czyli, że istnieją $1 \leqslant j_1 < j_2 < \ldots < j_k \leqslant n$ takie, że $X[i_j] = A[j]$ dla wszystkich $j = 1, \ldots, k$.

1. Wyznacz dokładny wzór na p_k.
2. Wyznacz asymptotykę ciągu $(p_k)_k$

Zadanie 6

Załóżmy, że wzorzec P składa się z różnych znaków. Pokaż, że problem dopasowania wzorca P do ciągu długości n można rozwiązać w czasie $O(n)$ (niezależnym od długości wzorca).

Ćwiczenie 1

Napisz w języku C program, który nie ma operacji wejściowych i generuje swoją własną kopię.

Ćwiczenie 2

Napisz w języku `C` funkcje które sprawdzają, czy dany łańcuch jest prefixem bądź postfixem drugiego. Jakie funkcje języka `Python` służą do tego celu?

Ćwiczenie 3

Napisz w języku `C` oraz `Python` funkcję która dla danych dwóch łańcuchów `x[1:n]` i `y[1:m]` zwraca największą liczbę k taką, że `x[1:k] = y[m-k+1:m]`.

Ćwiczenie 4

Zaimplementuj możliwie optymalnie (skorzystaj z funkcji `memcmp`) naiwny algorytm zgodności wzorca z tekstem w języku `C`.

Ćwiczenie 5

Mamy dany wzorzec P.

1. Napisz procedurę, która wyznacza prefixowy automat skończony służący do wykrywania obecności wzorca P w dowolnym łańcuchu.
2. Wykorzystaj powyższą procedurę do napisania procedury służącej do wykrywania obecności wzorca P w dowolnym łańcuchu.

Ćwiczenie 6

Zaimplementuj metodę Hornera wyznaczania wartości wielomianu i dokładnie zbadaj jej złożoność obliczeniową.

Ćwiczenie 7

Zaimplementuj algorytm Rabina-Karpa wykrywania obecności wzorca w tekście.

2 Model postawowy + kostka losowa

Zadanie 7

Mamy fałszywą monetę, która zwraca orła z prawdopodobieństwem p oraz reszkę z prawdopodobieństwem $q = 1-p$. Wiemy tylko, że $0 < p < 1$. Jak możesz użyć tej monety do wygerowania uczciwej monety (tzn. takiej aby prawdopodobieństwo orła i reszki było równe $\frac{1}{2}$?
Wskazówka: Użyj (co najmniej) dwóch rzutów fałszywą monetą.

Zadanie 8

Załóżmy, że mamy metodę generowania liczb losowych ze zbioru $\{0, 1, 2, 3, 4\}$ zgodną z jednostajnym rozkładem. Użyj tej metody do zbudowania generatora losowych liczb ze zbioru $\{0, 1, 2, 3, 4, 5, 6\}$ zgodnego z rozkładem jednostajnym.
Wskazówka: Wygeneruj najpierw losową liczbę ze zbioru $\{, 1, \ldots, 24\}$ i zastosuj metodę odrzucania.

Zadanie 9

Rozważamy metodę Monte-Carlo do obliczenia pola koła $\{(x, y) \in \mathbb{R}^2 : x^2 + y^2 \leqslant 1\}$.

1. Jaka jest wariancja tej metody?
2. Skorzystaj z nierówności Czebyszewa do oszacowania liczby iteracji do osiągnięcia dokładność 0.01 z prawdopodobieństwem 0.99?
3. Skorzystaj z następującego wariantu nierówności Chernoffa

   Jeśli $X_1, \ldots, X_n$ są niezależne takimi, że $0 \leqslant X_i \leqslant 1$ dla każdego $i = 1, \ldots, n$, $\mu = E(X_1 + \ldots + X_n)$ oraz $\varepsilon > 0$, to

$$Pr(|X - \mu| > \varepsilon\mu) \leqslant 2 \exp\left(-\frac{\varepsilon^2}{2 + \varepsilon}\mu\right)$$

do lepszego oszacowania liczby n potrzebnych prób.

Ćwiczenie 8

Zastosuj metodę Monte-Carlo do wyznaczenia aproksymacji $\int_0^\pi \sin(x)dx$.

Ćwiczenie 9

Zaimplementuj zrandomizowany min-cut Kargera. Jeżeli będziesz go implementował w języku Python, to do modelowania grafów możesz posłużyć się słownikami, np.

```
graph = {
  'a' : ['b','c']
  'b' : ['a','d'],
  ...
}
```

Przetestuj jego działanie na kilku prostych grafach.

Zadanie 10

Mamy fałszywą monetę, która zwraca orła z prawdopodobieństwem p oraz reszkę z prawdopodobieństwem $q = 1 - p$. Wiemy tylko, że $0 < p < 1$. Jak możesz użyć tej monety do wygerowania uczciwej monety (tzn. takiej aby prawdopodobieństwo orła i reszki było równe $\frac{1}{2}$?
Wskazówka: Użyj dwóch rzutów fałszywą monetą.

Fischer - Yeats, Freivald matrix multiplication verification, Karger min-cut

c.d.n.
Powodzenia,
Jacek Cichoń