Egzamin licencjacki — 11 wrzesnia 2009

Z zestawu sześciu zadań (Matematyka I, Matematyka II, Programowanie, Matematyka dyskretna, Algorytmy i struktury danych i Metody numeryczne) poniżej należy wybrać i przedstawić na osobnych kartkach rozwiązania trzech zadań. Za brakujące (do trzech) zadania zostanie wystawiona ocena nieostateczna z urzędu. Egzamin uważa się za zaliczony, jeśli student rozwiąże z oceną dostateczną co najmniej 2 zadania. Wtedy ocena z egzaminu jest średnią arytmetyczną ocen z trzech wybranych zadań. Na rozwiązanie zadań przeznacza się czas 3x40=120 minut. Po wyjściu z sali egzaminacyjnej w czasie egzaminu nie ma możliwości powrotu do tej sali i kontynuowania pisania egzaminu.

Matematyka I

1. Udowodnij, że dla wszystkich dodatnich liczb naturalnych n każdy n-wierzchołkowy spójny graf prosty ma co najmniej $n-1$ krawędzi.

Matematyka II

Za każdy podpunkt można otrzymać 2 punkty. Skala ocen: 5 punktów – *dostateczny*, 6 punktów – *+dostateczny*, 7 punktów – *dobry*, 8 punktów – *+dobry*, 9 punktów – *bardzo dobry.*

1. Dana jest permutacja $f = (1, 2, 3, 4, 5, 6, 7, 8)(3, 10)(4, 6)(4, 6, 3, 11) \in S_{11}$.
   (a) Przedstaw f jako złożenie cykli rozłącznych.
   (b) Znajdź permutację odwrotną do f.
   (c) Czy f jest parzysta?
   (d) Jaki jest rząd f? Czy w S_{11} istnieje permutacja o większym rzędzie?
   (e) Czy w S_{11} istnieje podgrupa izomorficzna z grupą symetrii kwadratu?

Programowanie

Za zadanie można otrzymać 20 punktów. Aby otrzymac ocenę dostateczną, należy zdobyć 7 punktów, próg dla dst+ to 9p, dla db – 11p, dla db+ 13p, dla bdb – 15p.

Część 1. Gramatyka G_1 z symbolem startowym S nad alfabetem $\{a, b\}$ dana jest za pomocą następującego zbioru produkcji:

$$\{S \to aSb\ ,\ S \to bSa,\ \ S \to \varepsilon\}$$

Gramatyka G_2 z symbolem startowym S nad alfabetem $\{a, b\}$ dana jest za pomocą następującego zbioru produkcji:

$$\{S \to abS,\ S \to aS\ ,\ S \to aSb,\ S \to aaSb,\ S \to SS,\ S \to \varepsilon\}$$

a) Czy gramatyki G_1 i G_2 są jednoznaczne (odpowiedź uzasadnij)? **(3p)**

b) Niech $A_1 = L(G_1) \cap L(G_2) \cap \mathcal{L}(a^*bbbb)$. Czy A_1 jest regularny? Przedstaw gramatykę bezkontekstową lub wyrażenie regularne generujące A_1 **(3p)**

c) Niech $A_2 = (L(G_1) \cup L(G_2)) \cap \mathcal{L}(a^*b^*)$. Przedstaw gramatykę generującą A_2. Uzasadnij, że istotnie generuje ona ten język. **(3p)**.

d) Czy A_2 jest regularny? **(1p)**

Część 2. Będziemy rozważać algorytm sortowania *quicksort*, w którym, aby posortować listę wybieramy jakiś element (na przykład pierwszy), dzielimy listę na dwie podlisty: większe oraz niewiększe od tego elementu. Podlisty sortujemy rekurencyjnie i sklejamy w wynikową, posortowaną listę. Zadanie ma dwa warianty: funkcjonalny oraz logiczny, powinieneś rozwiązać tylko jeden z nich.

Wariant funkcjonalny

Możesz używać Haskella albo SML-a. W specyfikacji zadania używamy typów Haskellowych.

a) Zaimplementuj `quicksort` w Haskellu. Przy ocenie liczy się również elegancja rozwiązania. **(6p)**

b) Napisz funkcję, która oblicza część wspólną dwóch zbiorów liczb całkowitych, pamiętanych jako listy elementów (bez powtórzeń). Istotna jest efektywność rozwiązania. **(4p)**

Wariant logiczny

Powinieneś używac Prologa.

a) Zaimplementuj `quicksort` w Prologu. Przy ocenie liczy się również elegancja rozwiązania. **(6p)**

b) Napisz predykat, która oblicza część wspólną dwóch zbiorów liczb całkowitych, pamiętanych jako listy elementów (bez powtórzeń). Istotna jest efektywność rozwiązania. **(4p)**

Matematyka dyskretna

1. Ile cykli Hamiltona ma graf $K_{n,n}$?

Algorytmy i struktury danych

Za poprawne rozwiązanie całego zadania można otrzymać 9 punktów. 3 punkty dają ocenę dostateczną, 4 — dostateczną z plusem, 5 — dobrą, 6 — dobrą z plusem, 7 — ocenę bardzo dobrą.

Część 1: sortowanie n liczb całkowitych (3 punkty)

Do n–elementowej tablicy $T[0 \dots n-1]$ wpisano liczby całkowite ze zbioru $\{0, 1, \dots, n^2 - 1\}$. Opracuj i opisz efektywny algorytm sortowania takich danych. Przeanalizuj złożoność obliczeniową twojego rozwiązania. Czy twój algorytm jest stabilny? Czy działa on w miejscu? Uzasadnij swoje odpowiedzi.

Część 2: zbiory rozłączne (3 punkty)

Opisz drzewiastą strukturę danych dla zbiorów rozłącznych:

- (1.5 pkt.) Opisz budowę drzewiastej struktury danych dla zbiorów rozłącznych. Jakie zadania realizują operacje *union* i *find* w tej strukturze? Na czym polega łączenie według rozmiaru/rangi? Na czym polega kompresja ścieżki podczas wyszukiwania?
- (1.0 pkt.) Napisz w pseudokodzie implementację operacji *union* z uwzględnieniem rozmiaru/rangi oraz *find* (może być wersja rekurencyjna) z kompresją ścieżki. Jaka jest pesymistyczna złożoność czasowa każdej z tych operacji?
- (0.5 pkt.) Jaka jest złożoność czasowa wykonania ciągu n operacji *union*/*find* na zbiorach rozłącznych z łączeniem według rozmiarów/rang i kompresją ścieżek? Czy ta złożoność się zmieni, jeśli założymy, że w tym ciągu wszystkie operacje *union* występują przed operacjami *find*?

Część 3: problem plecakowy (3 punkty)

Opisz ciągły problem plecakowy:

- (1.0 pkt.) Opisz dokładnie na czym polega ciągły problem plecakowy (co jest dane i jakich oczekujemy wyników). Jaką techniką rozwiązuje się to zadanie? Napisz w pseudokodzie algorytm rozwiązujący to zadanie.
- (0.5 pkt.) Jaka jest złożoność obliczeniowa tego algorytmu?
- (1.0 pkt.) Udowodnij (niewprost) poprawność przedstawionego algorytmu.
- (0.5 pkt.) Na czym polega różnica między ciągłym a dyskretnym problemem plecakowym? Wykaż, że przedstawiony algorytm nie będzie działał dla dyskretnego problemu plecakowego.

Metody numeryczne

1. Wielomian $w \in \Pi_n$ można zapisać w różnych postaciach, np.

   - w postaci naturalnej:
   $$w(x) = \sum_{k=0}^{n} a_k x^k,$$
   - w postaci Newtona:
   $$w(x) := \sum_{k=0}^{n} b_k p_k(x),$$
   gdzie $p_0(x) := 1$, $p_k(x) := (x - x_0)(x - x_1) \cdots (x - x_{k-1})$ $(k \geq 1)$, a liczby $x_0, x_1, \ldots, x_{n-1}$ są ustalone,
   - czy w postaci kombinacji liniowej wielomianów Czebyszewa:
   $$w(x) = {\sum_{k=0}^{n}}' c_k T_k(x).$$

(a) Podaj efektywny algorytm obliczania wartości wielomianu w w punkcie x dla każdej z podanych wyżej postaci.

(b) Dany jest napis złożony ze znaków "0", "1",. . . , "9" (np. "24071977"). Jak szybko wyznaczyć wartość liczby naturalnej odpowiadającej temu napisowi?