Egzamin licencjacki/inżynierski — 08 września 2017

Informacja dla zdających egzamin na kierunku informatyka:
Z sześciu poniższych zestawów zadań (Matematyka I, Programowanie, Matematyka dyskretna, Algorytmy i struktury danych, Metody numeryczne, Matematyka II) należy wybrać i przedstawić na osobnych kartkach rozwiązania trzech zestawów.

Informacja dla zdających egzamin na kierunku indywidualne studia informatyczno-matematyczne:
Z sześciu poniższych zestawów zadań (Matematyka I, Programowanie, Matematyka dyskretna, Algorytmy i struktury danych, Metody numeryczne, Języki formalne i złożoność obliczeniowa) należy wybrać i przedstawić na osobnych kartkach rozwiązania trzech zestawów.

Informacja dla wszystkich zdających:
Za brakujące (do trzech) zestawy zostanie wystawiona ocena niedostateczna z urzędu. Egzamin uważa się za zaliczony, jeśli student rozwiąże z oceną dostateczną co najmniej 2 zestawy. Wtedy ocena z egzaminu jest średnią arytmetyczną ocen z trzech wybranych zestawów. Na rozwiązanie 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 — Logika dla informatyków

Udowodnij, że istnieje dokładnie continuum różnych przechodnich relacji na zbiorze liczb naturalnych, które są funkcjami.

Wskazówka: Ciągów zero-jedynkowych też jest continuum.
Uwaga: To jest zadanie z logiki. Przy ocenianiu zwrócimy szczególną uwagę na poprawność i klarowność rozumowania, w szczególności na odpowiednie sformułowanie i użycie wszystkich założeń, odpowiednie użycie kwantyfikatorów i nawiasów itp.

Programowanie

Za tę część egzaminu 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.

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

$$S \to aaSbbb,\ S \to bbbSaa,\ S \to SS,\ S \to cS,\ S \to \varepsilon$$

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

$$S \to XY, X \to aXb,\ Y \to bY,\ X \to \varepsilon, Y \to \varepsilon,$$

Dla gramatyki G przez $L(G)$ rozumieć będziemy język generowany przez G. Dla wyrażenia regularnego r przez $\mathcal{L}(r)$ rozumiemy język opisany przez wyrażenie r.

a) Czy $aaaaaacccbbbbbb$ należy do $L(G_1)$? Odpowiedź uzasadnij. **(1)**

b) Czy gramatyka G_1 jest jednoznaczna? Odpowiedź krótko uzasadnij. **(2)**

c) Czy da się tak dobrać $f_1, f_2, f_3 : \mathbb{N} \to \mathbb{N}$, żeby było spełnione:

$$L(G_1) \cap L(G_2) = \{b^{f_1(k)} a^{f_2(k)} b^{f_3(k)} \text{ dla } k \in 0, 1, 2, \dots\}$$

Odpowiedź krótko uzasadnij. **(3)**

d) Napisz w języku imperatywnym funkcję, która bierze jako wejście napis i zwraca wartość logiczną, równą True wtedy i tylko wtedy, gdy ten napis należy do zbioru $L(G_1)$.

Możesz używać języka wybranego z następującej listy: C, C++, Java, C#, Python, Ruby, Go, AWK, Rust. **(4)**

Zadanie 2. (7p) Będziemy rozważać następujące listę równań postaci `term1 = term2`, w której termy składają się z binarnego symbolu funkcyjnego `f`, stałych `a` oraz `b`, oraz zmiennych (prologowych, czyli pisanych wielką literą). Równania te interpretujemy jako równania na termach (czyli takie, jakie rozwiązuje prologowy algorytm unifikacji). Napisz predykat `has_solution_with_term(+Term, +EquationList)`, który sprawdza, czy lista równań ma rozwiązanie, w którym któraś zmienna jest zunifikowana z termem `Term` (możesz przyjąć, że `Term` jest stały). W kolejnych nawrotach predykat powinien generować[1] wszystkie rozwiązania (niekoniecznie każde jednokrotnie).

Powinieneś zdefiniować odpowiednie predykaty pomocnicze (i dla każdego przedstawić krótki komentarz odnośnie jego działania). Skoncentruj się na czytelności, w szczególności unikaj rekursji ogonowej, jeżeli jej wprowadzenie łączy się z dodaniem dodatkowych argumentów do predykatu.

Zadanie 3. (3p) Napisz w Haskellu funkcję `sort`, która bierze listę liczb i zwraca jej posortowaną permutację. Algorytm jest dowolny, ale definicja powinna korzystać z mechanizmu list składanych (list comprehension).

[1] Generowanie rozwiązania oznacza zakończenie działania predykatu sukcesem i przypisanie odpowiednich wartości zmiennym z `EquationList`.

Matematyka dyskretna

Ile permutacji 26 liter angielskiego alfabetu nie zawiera żadnego z ciągów liter: *fish, rat, bird*? Napisz wzór — nie musisz wyliczać konkretnej liczby.

Algorytmy i struktury danych

Za rozwiązanie obydwu zadań z tej części można otrzymać w sumie do 9 punktów. Skala ocen: poniżej 3 punktów — ocena niedostateczna (egzamin niezdany), 3 punkty dają ocenę dostateczną, 4 — dostateczną z plusem, 5 — dobrą, 6 — dobrą z plusem, 7 albo więcej punktów daje ocenę bardzo dobrą.

Zadanie 1: skojarzenie doskonałe w drzewie (5 punktów)

Zaprojektuj algorytm, który dla zadanego drzewa (spójny graf acykliczny) rozstrzygmie, czy zawiera ono skojarzenie doskonałe. Skojarzenie doskonałe (ang. *perfect matching*) to taki podzbiór krawędzi, że każdy wierzchołek jest końcem dokładnie jednej krawędzi w tym podzbiorze.

Przedstaw efektywną metodę rozwiązującą to zadanie. Precyzyjnie opisz swój algorytm. Uzasadnij jego poprawność. Oszacuj złożoność obliczeniową opisanego rozwiązania.

Zadanie 2: podział drzewa BST (4 punkty)

Niech T będzie drzewem BST, w którym każdy węzeł przechowuje jedynie wartość liczbową oraz dwa wskaźniki na lewego i prawego syna. Zaprojektuj algorytm, który dla zadanej wartości v rozdzieli drzewo T na dwa rozłączne drzewa BST T_1 i T_2 takie, że drzewo T_1 będzie zawierało wszystkie elementy $< v$ a drzewo T_2 wszystkie elementy $\geq v$. Zadbaj o to, by wysokości drzew T_1 i T_2 były nie większe niż wysokość drzewa T.

Opisz ideę rozwiązania. Zapisz swój algorytm w pseudokodzie wraz z komentarzami. Oszacuj złożoność obliczeniową opisanej metody.

Metody numeryczne

Za rozwiązanie zadań można otrzymać łącznie 12 punktów. Otrzymanie 4 pkt. gwarantuje ocenę dostateczną, próg dla dst+ to 5.5 pkt., dla db – 7 pkt., dla db+ 9 pkt., a dla bdb – 11 pkt.

1. $\boxed{\textbf{4 punkty}}$ Niech dane będą: liczba dodatnia a oraz liczba naturalna m. Zaproponuj i krótko uzasadnij skuteczny algorytm znajdowania wartości $\sqrt[m]{a}$ z dużą dokładnością, przy założeniu, że wolno używać **jedynie** operacji arytmetycznych.

2. $\boxed{\textbf{4 punkty}}$ Wartości funkcji f znane są w parami różnych punktach $x_0, x_1, \ldots, x_n$. Ile i jakich operacji arytmetycznych należy wykonać, aby wyznaczyć ilorazy różnicowe $f[x_0, x_1, \ldots, x_k]$ dla $k = 0, 1, \ldots, n$? Jakie jest zastosowanie ilorazów różnicowych w interpolacji wielomianowej?

3. $\boxed{\textbf{4 punkty}}$ Podaj definicję naturalnej funkcji sklejanej trzeciego stopnia interpolującej funkcję f w węzłach $x_0 < x_1 < \cdots < x_n$. Sprawdź czy funkcja

$$s(x) = \begin{cases} 3x^3 + 9x^2 - 2x - 4 & \text{dla } -1 \le x \le 0, \\ -3x^3 + 9x^2 - 2x - 4 & \text{dla } 0 \le x \le 2, \\ 3x^3 - 27x^2 + 70x - 52 & \text{dla } 2 \le x \le 3 \end{cases}$$

jest naturalną funkcją sklejana trzeciego stopnia.

Matematyka II — Algebra

Za zadania można otrzymać 13 punktów. Aby otrzymać ocenę dostateczną, należy zdobyć 3 punkty, próg dla dst+ to 5p, dla db – 7p, dla db+ 9p, dla bdb – 11p.

Zadanie 1. (4 punkty)
Niech G będzie grupą. $C(x)$ oznacza centralizator elementu $x \in G$, to znaczy

$$C(x) = \{g \in G \mid gx = xg\}.$$

Udowodnić, że centralizator (dowolnego elementu x) jest podgrupą grupy G.

Zadanie 2. (6 punktów)
Niech T będzie przekształceniem liniowym z przestrzeni V w przestrzeń W ($T : V \to W$). Załóżmy dodatkowo, że T jest bijekcją. Wykazać, że przekształcenie odwrotne do T — nazwijmy je S — jest też przekształceniem liniowym.

Zadanie 3. (3 punkty)
Sprawdzić, czy wektory $w_1 = [1\ 2\ 2]$, $w_2 = [2\ 1\ 2]$, $w_3 = [2\ 1\ 1]$ są niezależne w przestrzeni liniowej $\mathbb{Z}_3^3$ (przestrzeni nad ciałem dodawania i mnożenia *modulo* 3.

Języki formalne i złożoność obliczeniowa

Niech $L = \{43^m2^n1^m0^n \mid n, m \in \mathbb{N}\}$ (zakładamy, że 0 jest naturalne), a D będzie zbiorem tych liczb, których dziesiętna reprezentacja należy do L.

Zadanie 1. (2 punkty) Czy L jest językiem bezkontekstowym?

Zadanie 2. (3 punkty) Czy istnieje nierozstrzygalny zbiór X taki, że $D \cap X$ jest rozstrzygalny?

Ocena to liczba zdobytych punktów.