

Egzamin licencjacki/inżynierski — 04 lutego 2019

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 przeznaczona jest czas $3 \times 40 = 120$ 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

Funkcja $f : \mathbb{N} \rightarrow \mathbb{N}$ jest *prawie okresowa*, jeśli spełnia warunek

$$\exists n_0 \in \mathbb{N} \exists k \in \mathbb{N}. k > 0 \wedge \forall n > n_0 f(n+k) = f(n).$$

Czy dla dowolnych prawie okresowych funkcji f i g funkcja gf jest funkcją prawie okresową? Czy odpowiedź zmieni się, jeśli usuniemy założenie, że $f : \mathbb{N} \rightarrow \mathbb{N}$ jest prawie okresowa? Podaj dowody albo kontrprzykłady.

Uwaga: To jest zadanie z logiki. Przy ocenianiu zwrócimy szczególną uwagę na poprawność i klarowność rozumowania, odpowiednie sformułowanie i użycie wszystkich założeń, odpowiednie użycie kwantyfikatorów i nawiasów itp.

Programowanie

Za tę część egzaminu można otrzymać 20 punktów. Aby otrzymać 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 P :

$$A \rightarrow a, A \rightarrow aA, B \rightarrow b, B \rightarrow bB, C \rightarrow c, C \rightarrow cC, S \rightarrow ASA, S \rightarrow BSB, S \rightarrow CSC, S \rightarrow \varepsilon$$

Dla gramatyki G przez $L(G)$ rozumiemy język generowany przez G . Dla wyrażenia regularnego r przez $\mathcal{L}(r)$ rozumiemy język opisany przez wyrażenie r . Uwaga: w niektórych wersjach wyrażen regularnych zamiast znaku $+$ używa się znaku $|$.

- a) Czy $abbccbaa$ należy do $L(G_1)$? Odpowiedź uzasadnij. **(1)**
- b) Czy gramatyka G_2 jest jednoznaczna? Odpowiedź krótko uzasadnij. **(2)**
- c) Opisz zbiór $L(G_1) \cap \mathcal{L}(a + b)^*$. Czy jest on regularny? Odpowiedź uzasadnij. **(3)**
- d) Opisz jednym zdaniem, jakie słowa należą do zbioru:

$$A = L(G_1) \cap \mathcal{L}(abc(abc + cba)^*)$$

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

Zadanie 2. (5p) Przyjmiemy, że atom ma głębokość 0 ($\text{depth}(a) = 0$), a ponadto dla termu $f(t_1, \dots, t_n)$ głębokością jest $1 + \max(\text{depth}(t_1), \dots, \text{depth}(t_n))$.

Napisz predykat `generate(-E, +Depth, +Value)`, który generuje (na pozycji `E`, wszystkie wyrażenia zbudowane ze znaków `+`, `-` oraz `*` (i nawiasów) oraz z cyfr, które to wyrażenia mają głębokość `Depth` oraz wartość `Value`.

Możesz zdefiniować predykat(y) pomocnicze.

Zadanie 3. (5p) Napisz w Haskellu funkcję, która bierze listę liczb naturalnych i zwraca sumę cyfr występujących na tej liście. Przykładowo dla listy `[1,22,33,4004]` powinna zwrócić `1+2+2+3+3+4+4`, czyli 19.

Matematyka dyskretna

Ile jest liczb nieparzystych, nie większych niż n , które nie dzielą się przez 3, 5 lub 7?

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: długa podróż (4 punkty)

Wybierasz się w długą podróż. Na trasie podróży jest zlokalizowanych n hoteli w odległościach $a_1 < a_2 < \dots < a_n$ kilometrów od punktu początkowego. Możesz wybrać hotele, w których chcesz się zatrzymać. Musisz jednak zatrzymać się w ostatnim n -tym hotelu, który jest punktem docelowym Twojej podróży.

Masz dziennie przebyć 256 kilometrów. Niestety może to być niemożliwe, ze względu na odległości między hotelami. Jeżeli w ciągu dnia pokonasz x kilometrów, to kara za ten dzień wyniesie $(256 - x)^2$. Chcesz tak rozplanować podróż, aby zminimalizować sumę kar z poszczególnych dni podróży.

Podaj efektywny algorytm wyznaczania optymalnego ciągu hoteli, w których powinien się zatrzymywać. Opisz ideę algorytmu, uzasadnij jego poprawność i oszacuj złożoność obliczeniową (czasową i pamięciową).

Zadanie 2: zastosowanie kopca binarnego w sortowaniu (5 punktów)

Opisz budowę *kopca binarnego*:

- Jakie operacje są wykonywane na takich kopcach binarnych?
- Jaką strukturę ma kopiec binarny (w sensie drzewa)? Jak taki kopiec umieścić w tablicy?
- Jaki porządek musi być zachowany w kopcu?
- Napisz w pseudokodzie implementację operacji przywracania porządku w kopcu binarnym *sieve-up* i *sieve-down*.
- Jak wykorzystać te operacje do implementacji innych operacji na kopcu?
- Przeanalizuj złożoność czasową wszystkich operacji wykonywanych na kopcu binarnym?

Następnie opisz *algorytm sortowania kopcowego*:

- Opisz ideę algorytmu.
- Zapisz go w pseudokodzie.
- Przeanalizuj złożoność obliczeniową tego algorytmu.

Metody numeryczne

Za rozwiązanie zadania 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. **4 punkty** Opisz metodę Newtona przybliżonego rozwiązywania równania nieliniowego $f(x) = 0$.
2. **3 punkty** Podaj definicję naturalnej interpolacyjnej funkcji sklejanej trzeciego stopnia (NIFS3). Znajdź NIFS3 dla danych

$$\begin{array}{c|c|c|c} x_k & -1 & 0 & 1 \\ \hline y_k & 0 & 2 & 4 \end{array}.$$

3. **5 punktów** Wykorzystaj schemat Hornera do zaproponowania algorytmu obliczania punktu $P_n(t)$ ($0 \leq t \leq 1$) leżącego na krzywej Béziera stopnia n ,

$$P_n(t) := \sum_{k=0}^n W_k B_k^n(t) \quad (W_k \in \mathbb{R}^d),$$

i działającego w czasie $O(n)$.

Matematyka II — Algebra

Zadanie 1. (7 punktów)

Znaleźć najmniejszą liczbę naturalną x taką, że:

$$\begin{cases} x \equiv 1 \pmod{3} \\ x \equiv 2 \pmod{5} \\ x \equiv 3 \pmod{7} \end{cases}$$

Zadanie 2. (7 punktów)

Macierz T przekształcenia liniowego ma postać:

$$T = \begin{bmatrix} 1 & 3 & 1 & 4 \\ 2 & 7 & 3 & 9 \\ 1 & 5 & 3 & 1 \\ 1 & 2 & 0 & 8 \end{bmatrix}$$

- a) Wskazać dowolną bazę jądra tego przekształcenia;
- b) Wskazać dowolną bazę obrazu przekształcenia T .

Progi punktowe: 4, 6, 8, 10, 12 punktów

Języki formalne i złożoność obliczeniowa

Słowo nieskończone nad alfabetem Σ to dowolny, nieskończony ciąg elementów Σ . Deterministyczny automat skończony działa na słowach nieskończonych tak samo, jak na słowach skończonych, tyle, że nigdy się nie zatrzymuje. Powiemy, że automat akceptuje dane słowo nieskończone, jeśli w czasie czytania tego słowa choć raz osiągnie stan akceptujący. Język słów nieskończonych takiego automatu to zbiór tych słów, które automat akceptuje.

1. (1 punkt) Czy klasa języków słów nieskończonych rozpoznawanych przez deterministyczne automaty skończone jest zamknięta na sumę?
2. (2 punkty) Czy klasa języków słów nieskończonych rozpoznawanych przez deterministyczne automaty skończone jest zamknięta na dopełnienie?
3. (2 punkty) Jaka jest złożoność obliczeniowa następującego problemu: dla danych dwóch deterministycznych automatów skończonych, czy języki słów nieskończonych tych automatów są takie same?

Zaliczenie od 3 punktów, ocena to liczba zdobytych punktów.