

# Egzamin licencjacki/inżynierski

27 czerwca 2025

## **Informacja dla zdających egzamin na kierunku informatyka**

Z sześciu poniższych zestawów zadań (Matematyka I, Matematyka II, Metody programowania, Matematyka dyskretna, Algorytmy i struktury danych, Metody numeryczne) należy wybrać i przedstawić na osobnych kartkach rozwiązania trzech zestawów.

## **Informacja dla zdających egzamin na kierunku ISIM**

Z sześciu poniższych zestawów zadań (Matematyka I, Metody programowania, 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  $3 \times 40 + 30 = 150$  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

**Kontekst zadania:** Chcemy pokazać, że równoważność dualna do prawa algebry Boole’a jest prawem algebry Boole’a. W dowodzie przydadzą się pewne lematy.

**Oznaczenia:** Niech  $\mathcal{F}$  oznacza zbiór formuł rachunku zdań zbudowanych ze zmiennych zdaniowych z ustalonego zbioru  $V$  oraz spójników ze zbioru  $\{\wedge, \vee, \neg, \top, \perp\}$ . Przez *prawo algebry Boole’a* rozumiemy dowolną tautologię postaci  $\phi_1 \Leftrightarrow \phi_2$ , gdzie  $\phi_1, \phi_2 \in \mathcal{F}$ . Rozważmy odwzorowania  $d, n: \mathcal{F} \rightarrow \mathcal{F}$  zdefiniowane indukcyjnie w następujący sposób.

$$\begin{array}{ll} d(p) = p, & n(p) = \neg p, \quad \text{dla wszystkich zmiennych } p \in V \\ d(\perp) = \top, & n(\perp) = \perp, \\ d(\top) = \perp, & n(\top) = \top, \\ d(\neg\phi) = \neg d(\phi), & n(\neg\phi) = \neg n(\phi), \\ d(\phi \wedge \psi) = d(\phi) \vee d(\psi), & n(\phi \wedge \psi) = n(\phi) \wedge n(\psi), \\ d(\phi \vee \psi) = d(\phi) \wedge d(\psi), & n(\phi \vee \psi) = n(\phi) \vee n(\psi). \end{array}$$

Dla wartościowania  $\sigma: V \rightarrow \{\top, \text{F}\}$  przez  $\tilde{\sigma}$  oznaczmy wartościowanie zdefiniowane wzorem

$$\tilde{\sigma}(v) = \begin{cases} \top, & \text{jeśli } \sigma(v) = \text{F} \\ \text{F}, & \text{wpp.} \end{cases}$$

**Przykład:** Formuła

$$(p \vee \top) \wedge \neg q \Leftrightarrow \neg(q \vee \perp)$$

jest prawem algebry Boole’a. Dualnym do niego prawem jest

$$(p \wedge \perp) \vee \neg q \Leftrightarrow \neg(q \wedge \top).$$

**Właściwe zadanie:**

- (a) Sformułuj zasadę indukcji, z której skorzystasz w dalszych częściach zadania.
- (b) Udowodnij, że dla dowolnego wartościowania  $\sigma$  i dowolnej formuły  $\phi$  zachodzi

$$\sigma \models d(\phi) \text{ wtedy i tylko wtedy, gdy } \tilde{\sigma} \models \neg\phi.$$

- (c) *ta część zadania została anulowana*
- (d) Udowodnij, że dla dowolnych formuł  $\phi, \psi \in \mathcal{F}$ , jeśli formuła  $\phi \Leftrightarrow \psi$  jest prawem algebry Boole’a, to  $d(\phi) \Leftrightarrow d(\psi)$  jest prawem algebry Boole’a.

## Matematyka II — Algebra

Zadanie 1. (4 punkty)

W przestrzeni  $\Pi_2$  wielomianów stopnia  $\leq 2$  definiujemy iloczyn skalarny

$$\langle f, g \rangle = \int_0^2 f(x) g(x) dx$$

Proszę wykonać ortogonalizację bazy potęgowej tej przestrzeni.

Zadanie 2. (5 punktów)

**Znaleźć** całkowite liczby  $a, b$  takie że  $23a + 17b = 1$ . (Odgadnięcie to 1p.)

Zadanie 3. (6 punktów)

Użyć chińskiego twierdzenia o resztach w celu wyznaczenia całkowitego  $x$  takiego, że  $x \equiv_5 2$ ,  $x \equiv_7 6$ ,  $x \equiv_{13} 3$ .

Progi punktowe: 5, 7, 9, 11, 13 punktów.

## Metody Programowania

Poniższe zadania należy rozwiązać używając języka Plait lub OCaml.

W poniższych zadaniach rozważamy funkcyjny język programowania **egz**, którego składnia abstrakcyjna zawiera:

- zmienne,
- funkcje jednoargumentowe,
- aplikacje funkcji,
- konstruktor pary i selektory elementów pary.

Semantyka jest typowa, tzn. analogiczna do semantyki języków Plait i OCaml.

**Zadanie 1** Zdefiniuj typ danych reprezentujący składnię abstrakcyjną języka **egz**. Zmienne reprezentuj przy użyciu symboli (dla języka Plait) lub ciągów znaków (dla języka OCaml).

**Zadanie 2** Zdefiniuj dwa alternatywne typy danych reprezentujące wartości w języku **egz** z różnymi reprezentacjami funkcji, odpowiednimi dla, odpowiednio, podstawieniowego i środowiskowego modelu obliczeń.

**Zadanie 3** Zaimplementuj ewaluator dla języka **egz** wykorzystujący podstawieniowy model obliczeń.

**Zadanie 4** Zaimplementuj ewaluator dla języka **egz** wykorzystujący środowiskowy model obliczeń.

## Matematyka dyskretna

Napisz funkcję tworzącą ciąg liczb harmoniczných  $H_n = \sum_{i=1}^n 1/n$ .

## 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: najdłuższy wspólny podciąg bez identycznych liter sąsiadujących ze sobą (5 punktów)

Dane są dwa  $n$ -elementowe ciągi znaków  $X = (x_1, x_2, \dots, x_n)$  i  $Y = (y_1, y_2, \dots, y_n)$ , zbudowane nad  $k$ -literowym alfabetem  $\Sigma = \{a_1, a_2, \dots, a_k\}$ , gdzie  $x_i, y_i \in \Sigma$  dla  $i = 1 \dots n$ , oraz  $n > 2$  i  $k \geq 2$ .

Należy wyznaczyć długość najdłuższego wspólnego podciągu dla zadanych ciągów  $X$  i  $Y$  zbudowanych nad alfabetem  $\Sigma$ , ale w podciągu tym nie mogą sąsiadować ze sobą dwie takie same litery.

- Najpierw przypomnij klasyczny dynamiczny algorytm rozwiązujący problem LCS.
- Następnie precyzyjnie opisz adaptację tego algorytmu na potrzeby opisanego problemu (w podciągu nie mogą sąsiadować ze sobą dwie takie same litery) i uzasadnij poprawność opisanego algorytmu. Zapisz opisany algorytm w pseudokodzie i oszacuj jego złożoność obliczeniową (czasową i pamięciową). Jak odtworzyć najdłuższy wspólny podciąg o wyznaczonej długości?
- Jak zmieni się rozwiązanie tego zadania, gdy alfabet będzie tylko 2-literowy.

### Zadanie 2: słownik z operacją *less-than* (4 punkty)

Rozważmy słownik  $S$ , którego elementy pochodzą ze zbioru z porządkiem liniowym. Do standardowych operacji słownikowych *insert*( $x$ ), *delete*( $x$ ) i *search*( $x$ ) dodajemy jeszcze operację *less-than*( $x$ ) określającą ile jest w słowniku elementów mniejszych niż  $x$ :

$$S.\text{less-than}(x) = |\{a \in S : a < x\}|$$

Zaprojektuj słownik w oparciu o jakąś inną znaną implementację takiej struktury danych, w którym każda z operacji słownikowych oraz dodatkowo operacja *less-than* będzie miała pesymistyczną złożoność czasową  $O(\log n)$ , gdzie  $n = |S|$ . Procedurę *less-than*( $x$ ) zapisz w pseudokodzie i opisz dokładnie jej działanie. Krótko ale precyzyjnie opisz, jak zmodyfikowałeś pozostałe procedury słownikowe.

## 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+ 8.5 pkt., a dla bdb – 10 pkt.

1. **4 punkty** Opisz zjawisko *utraty cyfr znaczących*. Jakie ma ono znaczenie w kontekście wykonywania **obliczeń zmiennopozycyjnych**? Czy zjawiska tego **można uniknąć**? Jeśli tak, to w jaki sposób — **podaj odpowiedni przykład**. Jeśli nie, to **napisz dlaczego**.
2. **4 punkty** Przy pomocy komputera chcemy dla zadanego przedziału  $[a, b]$  ( $a < b$ ) narysować przybliżony wykres wielomianu  $w$  stopnia  $n \leq 16$ . Z tym, że **nie znamy** jawnego wzoru dla tego wielomianu, a **jedynie jego wartości** w parami różnych argumentach  $x_0, x_1, \dots, x_n$  **leżących poza** przedziałem  $[a, b]$ . Zaproponuj sposób rozwiązania tego zadania, który będzie **efektywny** nie tylko **numerycznie**, ale i pod względem **złożoności obliczeniowej**.
3. **4 punkty** Dana jest macierz  $A$  stopnia  $n \in \mathbb{N}$ , której **wszystkie minory główne** są różne od zera. Zaproponuj sposób obliczania jej wyznacznika. W rozwiązaniu **uwzględnij aspekty numeryczne**, jak i te związane ze **złożonością obliczeniową**.

## Języki formalne i złożoność obliczeniowa

Lemat o pompowaniu mówi, że jeśli język  $\mathcal{L}$  jest regularny, to istnieje stała  $p \in \mathbb{N}$  taka, że dla każdego słowa  $w \in \mathcal{L}$  spełniającego  $|w| \geq p$  istnieje podział  $w = xyz$  taki, że: (a)  $|y| > 0$ , (b)  $|xy| \leq p$  oraz (c) dla każdego  $i \in \mathbb{N}$  zachodzi  $xy^iz \in \mathcal{L}$ .

Jaka jest złożoność następującego problemu:

- **Dane:** deterministyczny automat skończony  $\mathcal{A}$  oraz  $k > 0$
- **Pytanie:** czy istnieje stała z lematu o pompowaniu dla  $\mathcal{L}(\mathcal{A})$  o wartości co najwyżej  $k$ ?