Sprawdzian z systemów liczbowych

Grupa A

Informatyka: system dwójkowy, ósemkowy i szesnastkowy, kod U2 i zapis zmiennoprzecinkowy

Imię i nazwisko: _____

Klasa: _____

Data: _____

Punkty: _____ / 27

10 zadań, 27 pkt. Czytaj uważnie polecenia. W zadaniach otwartych zapisz obliczenia.

Zadanie 1. (0–1 pkt)

Jaka jest maksymalna wartość dziesiętna na 8 bitach w naturalnym kodzie binarnym (NKB, bez znaku)?

(A) 128
(B) 256
(C) 255
(D) 127

Zaznacz jedną poprawną odpowiedź.

Zadanie 2. (0–1 pkt)

Liczba binarna 101101_2 zapisana w systemie dziesiętnym jest równa:

(A) 43
(B) 44
(C) 46
(D) 45

Zaznacz jedną poprawną odpowiedź.

Zadanie 3. (0–3 pkt)

Oceń prawdziwość stwierdzeń o systemach liczbowych:

1. Każda liczba szesnastkowa zawiera wyłącznie cyfry 0–9.	**P**	**F**
2. Liczba 10_2 w systemie dziesiętnym to 2.	**P**	**F**
3. W systemie ósemkowym najwyższą cyfrą jest 8.	**P**	**F**

Przy każdym zdaniu zakreśl P (prawda) albo F (fałsz).

Zadanie 4. (0–4 pkt)

Dopasuj system lub kod liczbowy do jego kluczowej charakterystyki.

1. System oktalny	…	A. koduje jedną cyfrę dziesiętną na 4 bitach
2. System szesnastkowy	…	B. zakres wartości: 0 do 255, 8 bitów = 1 bajt
3. NKB 8-bitowy (bez znaku)	…	C. używa cyfr 0–9 i liter A–F
4. Kod BCD	…	D. używa cyfr 0–7 jako symboli

Przy każdym numerze wpisz literę pasującego elementu.

Zadanie 5. (0–4 pkt)

Dopasuj reprezentację do cechy:

1. NKB (bez znaku)	…	A. Zakres $[0, 2^n - 1]$, nie obsługuje ujemnych
2. BCD	…	B. Jedno zero, zakres $[-2^{n-1}, 2^{n-1} - 1]$
3. IEEE 754	…	C. Każda cyfra dziesiętna na 4 bitach
4. Kod U2	…	D. Mantysa, wykładnik, bit znaku — liczby rzeczywiste

Przy każdym numerze wpisz literę pasującego elementu.

Zadanie 6. (0–2 pkt)

Uzupełnij zdania dotyczące kodowania znaków: (1) Standard kodowania znaków, w którym każdemu podstawowemu znakowi łacińskiemu odpowiada 7- lub 8-bitowa wartość liczbowa (0-127 dla podstawowego zakresu), nazywa się (1) _____ . (2) Kod znaku wielkiej litery „A” w tym standardzie, zapisany w systemie dziesiętnym, wynosi (2) _____ .

Zadanie 7. (0–3 pkt)

Wykonaj dodawanie dwóch liczb zapisanych w systemie dwójkowym: $10110011_2 + 01001100_2$. Podaj wynik w systemie dwójkowym i dziesiętnym.

Zadanie 8. (0–3 pkt)

Wyjaśnij, na czym polega utrata precyzji przy zapisie liczb rzeczywistych w standardzie zmiennoprzecinkowym IEEE 754. Podaj przykład sytuacji, w której dodanie bardzo małej liczby do bardzo dużej nie zmienia wyniku.

Zadanie 9. (0–3 pkt)

Wyjaśnij, na czym polega reprezentacja liczb ujemnych w kodzie uzupełnień do dwóch (U2) i podaj, jak na 8 bitach zapisać liczbę -5.

Zadanie 10. (0–3 pkt)

Wyjaśnij, na czym polega zapis liczb ujemnych w systemie U2 (uzupełnienie do dwóch). Zapisz liczbę -5 w systemie U2 na 8 bitach i pokaż sposób obliczenia.

Rozwiąż ten sprawdzian online, z wyjaśnieniem każdej odpowiedzi: matury-online.pl/sprawdzian/informatyka/systemy-liczbowe

Sprawdzian z systemów liczbowych

Grupa B

Informatyka: system dwójkowy, ósemkowy i szesnastkowy, kod U2 i zapis zmiennoprzecinkowy

Imię i nazwisko: _____

Klasa: _____

Data: _____

Punkty: _____ / 27

10 zadań, 27 pkt. Czytaj uważnie polecenia. W zadaniach otwartych zapisz obliczenia.

Zadanie 1. (0–1 pkt)

Które stwierdzenia dotyczące systemów liczbowych są prawdziwe?

A. Konwersja z binarnego na szesnastkowy polega na grupowaniu bitów po 4.

B. Liczba FF_{16} jest równa 11111110_2 .

C. Liczba 1000_2 jest równa 7_{10} .

D. System BCD koduje każdą cyfrę dziesiętną na 3 bitach.

E. Liczba 10_8 jest równa 9_{10} .

Zaznacz wszystkie poprawne odpowiedzi.

Zadanie 2. (0–1 pkt)

Liczba szesnastkowa $2F_{16}$ zapisana w systemie dziesiętnym jest równa:

(A) 47 (B) 45

(C) 31 (D) 62

Zaznacz jedną poprawną odpowiedź.

Zadanie 3. (0–3 pkt)

Oceń prawdziwość stwierdzeń o reprezentacji liczb w komputerze:

1. Operacja XOR liczby z samą sobą zawsze daje 0.	**P**	**F**
2. Liczba zmiennoprzecinkowa w standardzie IEEE 754 na 32 bitach (float) ma 23 bity mantysy.	**P**	**F**
3. W kodzie U2 na 8 bitach, operacja $127 + 1$ daje wynik 128.	**P**	**F**

Przy każdym zdaniu zakreśl P (prawda) albo F (fałsz).

Zadanie 4. (0–4 pkt)

Dopasuj pojęcie związane z reprezentacją liczb w komputerze do jego opisu.

1. Bit parzystości (parity bit)	...	A. Kod binarny, w którym sąsiednie liczby różnią się dokładnie na jednym bicie
2. Nadmiar (overflow)	...	B. Reprezentacja liczby rzeczywistej za pomocą mantysy i wykładnika
3. Kod Graya	...	C. Dodatkowy bit służący do wykrywania pojedynczych błędów podczas transmisji danych
4. Liczba zmiennoprzecinkowa (floating point)	...	D. Sytuacja, w której wynik działania arytmetycznego nie mieści się w dostępnej liczbie bitów

Przy każdym numerze wpisz literę pasującego elementu.

Zadanie 5. (0–4 pkt)

Dopasuj liczbę zapisaną w systemie binarnym, szesnastkowym lub ósemkowym do jej wartości w systemie dziesiętnym.

1. 1010_2	...	A. Liczba dziesiętna 10
2. $1F_{16}$	...	B. Liczba dziesiętna 24
3. 17_8	...	C. Liczba dziesiętna 31
4. 11000_2	...	D. Liczba dziesiętna 15

Przy każdym numerze wpisz literę pasującego elementu.

Zadanie 6. (0–2 pkt)

Uzupełnij konwersje: (1) Liczba 52_{10} w systemie ósemkowym to (1) _____$_8$. (2) Liczba 10101010_2 w systemie szesnastkowym to (2) _____$_{16}$.

Zadanie 7. (0–3 pkt)

Przedstaw liczbę dziesiętną 205 w systemie szesnastkowym (heksadecymalnym). Pokaż obliczenia.

Zadanie 8. (0–3 pkt)

Wyjaśnij metodę kodu uzupełnienia do dwóch (U2 / two's complement) i oblicz reprezentację liczby -13 w 8-bitowym kodzie U2. Podaj wszystkie kroki.

Zadanie 9. (0–3 pkt)

Wyjaśnij, na czym polega nadmiar (overflow) przy dodawaniu dwóch liczb w kodzie U2 (uzupełnienie do dwóch) na ograniczonej liczbie bitów. Podaj przykład dwóch liczb 4-bitowych w U2, których dodanie powoduje nadmiar.

Zadanie 10. (0–3 pkt)

Wyjaśnij, czym różni się kodowanie liczb całkowitych ze znakiem w systemie „znak-moduł" (sign-magnitude) od kodowania w systemie uzupełnień do dwóch (U2). Wskaż jedną wadę reprezentacji znak-moduł, której nie ma U2.

Rozwiąż ten sprawdzian online, z wyjaśnieniem każdej odpowiedzi: matury-online.pl/sprawdzian/informatyka/systemy-liczbowe

Sprawdzian z systemów liczbowych

Grupa A

1. (1 pkt) **C**

2. (1 pkt) **D**

3. (3 pkt) **1. F; 2. P; 3. F** — po 1 pkt za zdanie

4. (4 pkt) **1–D, 2–C, 3–B, 4–A** — po 1 pkt za parę

5. (4 pkt) **1–A, 2–C, 3–D, 4–B** — po 1 pkt za parę

6. (2 pkt) **(1) ASCII; (2) 65** — po 1 pkt za lukę

7. (3 pkt) Przykładowa odpowiedź: Zamiana na dziesiętne: $10110011_2 = 128 + 32 + 16 + 2 + 1 = 179$, $01001100_2 = 64 + 8 + 4 = 76$. Suma: $179 + 76 = 255 = 11111111_2$. Weryfikacja: $11111111_2 = 2^8 - 1 = 255$.
Punktacja: 1. Przeliczenie lub dodawanie kolumnowe z przeniesieniami (1 pkt). 2. Wynik dwójkowy: 11111111_2 (1 pkt). 3. Wynik dziesiętny: 255 (1 pkt).

8. (3 pkt) Przykładowa odpowiedź: Liczby zmiennoprzecinkowe IEEE 754 zapisują wartość jako iloczyn mantysy i potęgi dwójki, a mantysa ma ograniczoną, stałą liczbę bitów (np. 23 bity dla pojedynczej precyzji). Wiele ułamków dziesiętnych, np. $0,1$, nie da się zapisać dokładnie w systemie binarnym, więc są zaokrąglane do najbliższej reprezentowalnej wartości. Jeśli dodamy bardzo małą liczbę (np. 10^{-10}) do bardzo dużej (np. 10^{10}), różnica rzędów wielkości sprawia, że mała liczba nie mieści się w precyzji mantysy wyniku i zostaje zaokrąglona do zera względem dużej liczby — wynik pozostaje praktycznie niezmieniony.
Punktacja: Poprawne wyjaśnienie skończonej liczby bitów mantysy jako źródła utraty precyzji — 1pkt. Odniesienie do ułamków dziesiętnych niereprezentowalnych dokładnie binarnie (np. $0,1$) — 1pkt. Trafny przykład dodania bardzo małej liczby do bardzo dużej bez zmiany wyniku — 1pkt. Maks. 3 pkt.

9. (3 pkt) Przykładowa odpowiedź: W kodzie uzupełnień do dwóch (U2) najstarszy bit ma wagę ujemną — dla liczby n-bitowej wynosi -2^{n-1}, a pozostałe bity mają zwykłe, dodatnie wagi potęg dwójki. Aby zapisać liczbę ujemną, bierze się zapis binarny jej wartości bezwzględnej, neguje się wszystkie bity, a następnie dodaje się 1. Dla liczby 5 na 8 bitach: $5 = 00000101_2$. Negacja bitów daje 11111010. Dodanie 1 daje 11111011. Zatem -5 w kodzie U2 na 8 bitach to 11111011_2.
Punktacja: 1 pkt za wyjaśnienie idei U2 (najstarszy bit ma ujemną wagę, lub opis przez negację bitową i dodanie 1). 1 pkt za poprawny opis algorytmu zamiany liczby dodatniej na jej reprezentację ujemną w U2. 1 pkt za poprawny wynik dla -5 na 8 bitach. Maks. 3 pkt.

10. (3 pkt) Przykładowa odpowiedź: W systemie U2 liczbę ujemną otrzymuje się, zapisując najpierw odpowiadającą jej liczbę dodatnią w postaci binarnej, następnie negując wszystkie bity (zamieniając 0 na 1 i odwrotnie), a na koniec dodając 1 do wyniku. Dla -5 na 8 bitach: $5_{10} = 00000101_2$. Negacja bitowa: 11111010_2. Dodanie 1: $11111010_2 + 1 = 11111011_2$. Zatem -5 w systemie U2 na 8 bitach to 11111011_2.
Punktacja: Poprawne wyjaśnienie idei U2 — negacja bitowa liczby dodatniej i dodanie 1 (1 pkt) + poprawny zapis liczby 5 na 8 bitach jako punkt wyjścia (1 pkt) + poprawny końcowy wynik dla -5 wraz z pokazaniem obliczenia (1 pkt). Maksymalnie 3 pkt.

Grupa B

1. (1 pkt) **A**

2. (1 pkt) **A**

3. (3 pkt) **1. P; 2. P; 3. F** — po 1 pkt za zdanie

4. (4 pkt) **1–C, 2–D, 3–A, 4–B** — po 1 pkt za parę

5. (4 pkt) **1–A, 2–C, 3–D, 4–B** — po 1 pkt za parę

6. (2 pkt) **(1) 64; (2) AA** — po 1 pkt za lukę

7. (3 pkt) Przykładowa odpowiedź: 205 : 16 = 12, reszta 13. 12 : 16 = 0, reszta 12. Odczytując reszty od dołu do góry: 12, 13, czyli w systemie szesnastkowym cyfry C (12) i D (13). Wynik: 205 = CD_{16}.
Punktacja: Poprawnie wykonane kolejne dzielenia przez 16 z zapisem reszt (1 pkt) + poprawne odczytanie cyfr wyniku od dołu, w tym zamiana 12→C i 13→D (1 pkt) + poprawny wynik końcowy: CD_{16} (1 pkt). Razem 3 pkt.

8. (3 pkt) Przykładowa odpowiedź: Kod U2 pozwala reprezentować ujemne liczby całkowite: (1) zapisujemy wartość bezwzględną binarnie: $13_{10} = 00001101_2$; (2) negujemy wszystkie bity (NOT): 11110010_2; (3) dodajemy 1: $11110010_2 + 1 = 11110011_2$. Wynik: $-13_{10} = 11110011_{U2}$. Zakres 8-bitowego U2: od -128 do $+127$.
Punktacja: Definicja U2 (metoda reprezentacji ujemnych liczb całkowitych w binarnym systemie komputerowym) — 1 pkt. Poprawne obliczenie: $13_{10} = 00001101_2$; negacja bitowa → 11110010_2; +1 → 11110011_2 — 1 pkt. Poprawny wynik końcowy $-13_{10} = 11110011_2$ (U2) — 1 pkt.

9. (3 pkt) Przykładowa odpowiedź: Nadmiar (overflow) w kodzie U2 na n bitach występuje, gdy wynik dodawania dwóch liczb wykracza poza zakres reprezentowalny na tych n bitach (dla 4 bitów w U2 zakres to $[-8, 7]$). Nadmiar sygnalizuje sytuacja, w której dodajemy dwie liczby o tym samym znaku (obie dodatnie lub obie ujemne), a otrzymany wynik ma znak przeciwny — to znak, że „nie zmieścił się" w dostępnej liczbie bitów. Przykład: na 4 bitach liczba 5 to `0101`, a liczba 4 to `0100`. Ich suma binarna `0101+0100=1001`, co w U2 na 4 bitach odczytuje się jako -7, mimo że matematycznie $5 + 4 = 9$ — wystąpił nadmiar, bo wynik 9 przekracza maksymalną wartość 7 możliwą do zapisania na 4 bitach w U2.
Punktacja: Poprawne wyjaśnienie, że nadmiar występuje, gdy matematyczny wynik dodawania nie mieści się w zakresie reprezentowalnym przez daną liczbę bitów w U2 (1 pkt) + wskazanie, że sygnałem nadmiaru jest dodanie dwóch liczb tego samego znaku i otrzymanie wyniku o przeciwnym znaku (1 pkt) + poprawny przykład liczbowy (1 pkt). 3 pkt.

10. (3 pkt) Przykładowa odpowiedź: W reprezentacji znak-moduł najstarszy bit określa znak liczby (0 — dodatnia, 1 — ujemna), a pozostałe bity zapisują wartość bezwzględną wprost w systemie binarnym. W kodzie U2 nie wyodrębnia się osobnego bitu znaku — najstarszy bit ma po prostu ujemną wagę (np. dla 8 bitów wagę -128), dzięki czemu dodawanie liczb dodatnich i ujemnych realizuje ten sam obwód sumatora bez rozróżniania przypadków. Wadą znak-moduł jest istnienie dwóch różnych zapisów zera ($+0$ i -0), czego U2 nie ma — tam zero ma tylko jedną reprezentację.
Punktacja: Poprawny opis znak-moduł (najstarszy bit = znak, pozostałe bity = wartość bezwzględna) — 1pkt. Poprawny opis U2 (najstarszy bit ma ujemną wagę, brak osobnego bitu znaku) — 1pkt. Trafna wada znak-moduł (np. dwie reprezentacje zera, bardziej złożone dodawanie) — 1pkt. Maks. 3 pkt.

Skala ocen (orientacyjna, 27 pkt w każdej grupie)

bardzo dobry (5)	25–27 pkt
dobry (4)	21–24 pkt
dostateczny (3)	14–20 pkt
dopuszczający (2)	9–13 pkt
niedostateczny (1)	0–8 pkt

Progi: 90%, 75%, 50%, 30%. Progi w Twojej szkole mogą być inne. W zadaniach otwartych uznaj każdą odpowiedź poprawną merytorycznie, także sformułowaną inaczej niż przykład.

Ten sprawdzian online, z wyjaśnieniem każdej odpowiedzi: **matury-online.pl/sprawdzian/informatyka/systemy-liczbowe**. Nauczyciele i korepetytorzy ułożą własną kartę z banku ponad 20 tys. zadań: **matury-online.pl/dla-korepetytorow**.