

# Architektury systemów komputerowych

## Lista zadań nr 0

Na zajęcia od 24 do 28 lutego 2025

W zadaniach odnoszących się do języka C wolno używać **wyłącznie** instrukcji przypisania, operatorów bitowych, dodawania i odejmowania, przesunięć bitowych i stałych! Pętle, rozgałęzienia, operatory mnożenia, dzielenia i reszty z dzielenia są **niedozwolone**! Zakładamy, że liczby są typu «uint32\_t» – tj. nie posiadają znaku i mają szerokość 32 bitów. Należy wytłumaczyć czemu rozwiązanie działa!

**UWAGA!** W trakcie prezentacji należy być gotowym do zdefiniowania pojęć oznaczonych **wytłuszczoną** czcionką.

**Zadanie 1.** Przekształć każdą z podanych liczb z systemu ósemkowego na system binarny, szesnastkowy i dziesiętny:  $42_8$ ,  $255_8$ ,  $3047_8$  i  $140336_8$ .

**Zadanie 2.** Wykonaj poniższe operacje bez konwersji liczb do systemu dziesiętnego lub binarnego:

- $22_{16} + 8_{16}$
- $73_{16} + 2C_{16}$
- $7F_{16} + 7F_{16}$
- $C2_{16} + A4_{16}$

**Zadanie 3.** Napisz instrukcje w języku C, które dla zmiennych  $x$  i  $k$  wykonają poniższe obliczenia:

- wyzeruj  $k$ -ty bit zmiennej  $x$ ,
- zapal<sup>1</sup>  $k$ -ty bit zmiennej  $x$ ,
- zaneguj  $k$ -ty bit zmiennej  $x$ .

**Zadanie 4.** Napisz wyrażenia w języku C, które dla zmiennych  $x$  i  $y$  wykonają poniższe obliczenia:

- $x \cdot 2^y$ ,
- $\lfloor x/2^y \rfloor$ ,
- $x \bmod 2^y$ ,
- $\lceil x/2^y \rceil$ .

**Uwaga!** W ostatnim wyrażeniu nie wolno dopuścić do **przepełnienia** (ang. *overflow*) co może wystąpić, jeśli  $x$  jest bardzo duże.

**Zadanie 5.** Napisz wyrażenie w języku C, które wyznaczy liczbę przeciwną do liczby przechowywanej w zmiennej  $x$  typu «int32\_t». W wyrażeniu nie wolno użyć unarnego lub binarnego operatora minus!

**Uwaga!** Przyjmujemy tutaj, że przepełnienie na liczbach ze znakiem zachowuje się dokładnie tak, jak na liczbach bez znaku.

**Zadanie 6.** Napisz ciąg instrukcji w języku C, który zmieni miejscami zmienne  $x$  i  $y$ . Nie należy używać zmiennych tymczasowych.

**Zadanie 7.** Napisz wyrażenie w języku C, które oblicza się do 0 wtedy i tylko wtedy gdy liczba  $x$  jest potęgą dwójki.

**Uwaga!** Pamiętaj, że 0 nie jest potęgą dwójki.

**Zadanie 8.** Napisz ciąg instrukcji w języku C, który skonwertuje zmienną  $x$  z formatu **little-endian** do formatu **big-endian**. Należy użyć jak najmniejszej liczby operacji bitowych.

**Zadanie 9.** Jaką rolę pełnią **kody sterujące** standardu **ASCII** o numerach 0, 4, 7, 10 i 12?

**Wskazówka:** Opisy kodów sterujących można znaleźć w artykule [ASCII control code chart](https://en.wikipedia.org/wiki/ASCII#Control_code_chart)<sup>2</sup>.

---

<sup>1</sup>Stosowane zamiennie z „ustaw”. Te słowa zawsze oznaczają ustalenie wartości bitu na 1.

<sup>2</sup>[https://en.wikipedia.org/wiki/ASCII#Control\\_code\\_chart](https://en.wikipedia.org/wiki/ASCII#Control_code_chart)

**Zadanie 10.** Jakie ograniczenia standardu ASCII przyczyniły się do powstania **UTF-8**? Wyjaśnij zasadę kodowania znaków do postaci binarnej UTF-8 i zapisz poniższy ciąg znaków w systemie szesnastkowym:

*Proszę zapłacić 5€!*