

Architektury systemów komputerowych

Lista zadań nr 0

Na zajęcia 27 lutego 2020

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 rozwiązań należy zdefiniować i wyjaśnić pojęcia, które zostały oznaczone **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:

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

Podpowiedź: Użyj tabelki dodawania dla liczb w systemie szesnastkowym.

Zadanie 3. Napisz fragment kodu, który dla zmiennych x i k wykona poniższe operacje:

- wyzeruje k -ty bit zmiennej x ,
- ustawi k -ty bit zmiennej x ,
- zaneguje k -ty bit zmiennej x .

Zadanie 4. Napisz fragment kodu, który dla zmiennych x i y obliczy poniższe wyrażenia:

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

Zadanie 5. Napisz fragment kodu, który stwierdza czy dana liczba x nie jest potęgą dwójki.

Zadanie 6. Napisz fragment kodu, który skonwertuje zmienną x z formatu **little-endian** do formatu **big-endian**. Należy użyć jak najmniejszej liczby operacji bitowych.

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

Zadanie 8. 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€!