

Architektury systemów komputerowych

Lista zadań nr 3

Na zajęcia 18 – 19 marca 2019

Jeśli nie stwierdzono inaczej, rozwiązania zadań muszą się trzymać następujących wytycznych:

- Założenia:
 - liczby całkowite są w reprezentacji uzupełnień do dwóch,
 - wartość logiczna prawdy i fałszu odpowiada kolejno wartościom całkowitoliczbowym 1 i 0,
 - przesunięcie w prawo na liczbach ze znakiem jest przesunięciem arytmetycznym,
 - dane typu `int` mają N bitów długości; rozwiązanie musi działać dla dowolnego N będącego wielokrotnością 8.
- Zabronione:
 - wyrażenia warunkowe (`?:`) i wszystkie instrukcje poza przypisaniem,
 - operacja mnożenia, dzielenia i reszty z dzielenia,
 - operacje logiczne (`&&`, `||`, `^^`),
 - porównania (`<`, `>`, `<=` i `>=`).
- Dozwolone:
 - operacje bitowe,
 - przesunięcie w lewo i prawo z argumentem w przedziale $0 \dots N - 1$,
 - dodawanie i odejmowanie,
 - test równości (`==`) i nierówności (`!=`),
 - stała N , stałe własne oraz zdefiniowane w pliku nagłówkowym `<limits.h>`.

Zadanie 1. Zastąp instrukcję dzielenia całkowitoliczbowego zmiennej `n` typu `int32_t` przez stałą 3 przy pomocy operacji mnożenia liczb typu `int64_t`. Skorzystaj z faktu, że $\frac{x}{k} \equiv x \cdot \frac{1}{k}$. Przedstaw dowód poprawności swojego rozwiązania. Instrukcja dzielenia działa zgodnie z wzorem podanym na wykładzie, tj.:

$$\text{div3}(n) = \begin{cases} \lfloor \frac{n}{3} \rfloor & \text{dla } n \geq 0 \\ \lceil \frac{n}{3} \rceil & \text{dla } n < 0 \end{cases}$$

Wskazówka: Spróbuj rozwiązać zadanie samodzielnie, a następnie przeczytaj §10.3 książki „Uczta programistów”.

Zadanie 2. Standard IEEE 754-2008 definiuje liczby zmiennopozycyjne o szerokości 16-bitów. Zapisz ciąg bitów reprezentujący liczbę $1.5625 \cdot 10^{-1}$. Porównaj zakres liczbowy i dokładność w stosunku do liczb zmiennopozycyjnych pojedynczej precyzji (`float`).

Zadanie 3. Oblicz ręcznie $3.984375 \cdot 10^{-1} + 3.4375 \cdot 10^{-1} + 1.771 \cdot 10^3$ używając liczb w formacie z poprzedniego zadania. Zapisz wynik binarnie i dziesiętnie. Czy wynik się zmieni jeśli najpierw wykonamy drugie dodawanie?

UWAGA! Domyślną metodą zaokrąglania w obliczeniach zmiennoprzecinkowych jest *round-to-even*.

Zadanie 4. Mamy zmienne «x», «y» i «z» typu «int32_t» ustawione na wybrane przez nas wartości. Konwertujemy je do liczb typu «double» zapisanych w zmiennych «dx», «dy» i «dz». Rozważmy poniższe wyrażenia z języka C. Wskaż, które z nich mogą obliczyć się do fałszu. Podaj kontrprzykład – konkretne wartości naszych zmiennych całkowitoliczbowych.

1. `(float)x == (float)dx`
2. `dx - dy == (double)(x - y)`
3. `(dx + dy) + dz == dx + (dy + dz)`
4. `(dx * dy) * dz == dx * (dy * dz)`
5. `dx / dx == dz / dz`

Zadanie 5. Reprezentacje binarne liczb zmiennoprzecinkowych f i g typu «float» zostały załadowane odpowiednio do zmiennych «x» i «y» typu «uint32_t». Podaj wyrażenie, które:

1. zmieni znak liczby «x»,
2. obliczy wartość $\lfloor \log_2|x| \rfloor$ typu «int» dla f w postaci znormalizowanej,
3. zwróci wartość logiczną operacji «x == y»,
4. zwróci wartość logiczną operacji «x < y».

Pamiętaj, że dla liczb zmiennopozycyjnych w standardzie IEEE 754 zachodzi $-0 \equiv +0$. Można pominąć rozważanie wartości NaN.

Wskazówka: Spróbuj rozwiązać zadanie samodzielnie, a następnie przeczytaj §15.2 książki „Uczta programistów”.

Zadanie 6. Reprezentacja binarna liczby zmiennoprzecinkowej f typu «float» została załadowana do zmiennej «x» typu «uint32_t». Podaj algorytm obliczający $f \cdot 2^i$ wykonujący obliczenia na zmiennej «x» używając wyłącznie operacji na liczbach całkowitych. Osobno rozważ $i \geq 0$ i $i < 0$. Zakładamy, że liczba f jest znormalizowana, ale wynik operacji może dać wartość $\pm\infty$, ± 0 lub liczbę zdenormalizowaną.

UWAGA! Należy podać algorytm, zatem dozwolona jest cała składnia języka C bez ograniczeń z nagłówka listy zadań. Jednakże należy używać wyłącznie operacji na typie «int32_t».

Zadanie 7. Uzupełnij ciało funkcji zadeklarowanej następująco:

```
/* Skonwertuj reprezentację liczby float do wartości int32_t. */
int32_t float2int(int32_t f);
```

Zaokrąglij liczbę w kierunku zera. Jeśli konwersja spowoduje nadmiar lub f ma wartość NaN, zwróć 0x80000000. Dla czytelności napisz najpierw rozwiązanie z instrukcjami warunkowymi. Potem przepisz je, by zachować zgodność z wytycznymi z nagłówka listy.

Wskazówka: Postaraj się znaleźć jak najkrótsze rozwiązanie. Wzorcówka ma około 10 linii kodu!

Zadanie 8. Na podstawie artykułów [0x5f3759df](http://h14s.p5r.org/2012/09/0x5f3759df.html)¹ oraz [0x5f3759df \(appendix\)](http://h14s.p5r.org/2012/09/0x5f3759df-appendix.html)² zreferuj działanie algorytmu szybkiego przybliżania odwrotności pierwiastka kwadratowego z liczby typu «float». Należy wyjaśnić podstawy obliczeń na binarnej reprezentacji liczby «x» i pochodzenie stałej 0x5f3759df.

¹<http://h14s.p5r.org/2012/09/0x5f3759df.html>

²<http://h14s.p5r.org/2012/09/0x5f3759df-appendix.html>