

Czas: 24h. Używanie kalkulatorów: zabronione. Do notatek służy druga strona kartki. Przejrzysty zapis obliczeń ułatwia mi rozstrzyganie przypadków niejednoznacznych. Życzę Wam powodzenia – Piotr Patronik

Numer indeksu: $h_5h_4h_3h_2h_1h_0$ Data i godzina
wypełnienia: Niech h_i oznaczać cyfry szesnastkowe numeru indeksu.

1. (4p) Jest dany procesor o 4-bitowym słowie rozkazowym i poniższym kodowaniu rozkazów, w którym czas wykonania każdej mikrooperacji wynosi 250ps. Zapisać program w postaci mnemoników. Zobrazować i podać czas wykonania 6 rozkazów (zapisanych szesnastkowo): h_1 , h_3 , 0xFE, h_0 , h_4 dwóch przypadkach: (i) procesor jest w pełni sekwencyjny, (ii) mikrooperacje (F, W), (F, E), (D, W) mogą być wykonane równocześnie w potoku.

Kod	Zapis	Operacja	Mikrooperacje
$ii\ v\ 1$	ld \$v, %ri	$ri \leftarrow v$	FDW
$ii\ j\ 0$	add %ri, %rj	$rj \leftarrow ri+rj$	FDREW

2. (5p) Jest dana liczba binarna $(1+bbbb\cdot2^{-23}+3\cdot2^{-23})\cdot2^{-2d}$ gdzie $bbbb$ to 4 najmłodsze bity, zaś d to najmłodsza cyfra dziesiętna numeru indeksu. Zapisać tę liczbę w postaci zgodnej z normą IEEE 754, a następnie obliczyć (i zapisać jw.) sześćcian tej liczby korzystając z przybliżenia $(1\pm x)^{1/2}\approx 1\pm 1/2x$ dla $x\approx 1$. Omówić (2 zd.) użyty schemat zaokrąglania.

3. (3p) Omówić różnice pomiędzy sumatorem CLA a PPA.

4. (5p) Jest dany fragment kodu pewnego procesu w systemie z segmentacją stronicowaną, z modelem programowym x86. Niech rozkaz rand (będący rozszerzeniem modelu) ładuje rejestr wartością zmiennej losowej wg rozkładu jednostajnego z przedziału od 0 do wartości z rejestru. Jaki powinien być minimalny rozmiar segmentu danych tego procesu?

```
mov $(h0*15+2), %eax
mov $0x10000, %ebx
mul %ebx
mov %eax, %ebx
mov $-1, %ecx

begin:
    mov %ebx, %eax
    rand %eax
    mov %eax, (%eax)
loop begin
```

5. (4p) Wymienić z przykładami tryby adresowania architektury x86

```
mov ..... - ..... ADDR = .....
mov ..... - ..... ADDR = .....
mov ..... - ..... ADDR = .....
mov ..... - ..... ADDR = .....
mov ..... - ..... ADDR = .....
mov ..... - ..... ADDR = .....
```

6. (4p) Przedstawić: $h_0 \bmod 2=0$: zasady ochrony zasobów procesu, $h_0 \bmod 2=1$: metody wspomagania ochrony na poziomie architektury.