

Rozwiązania egzaminów teoretycznych z AKO

Emilian Zawrotny

14.02.2018

Zad. 1

```
push 41650000h
fld dword PTR [esp]
add esp, 4
```

Zad. 2

Wyrównanie naturalne jest to wyrównanie danych w taki sposób, by adres ich początku był podzielny przez rozmiar tych danych wyrażony w bajtach. Np. tablica intów wyrównana naturalnie zacznie się w adresie podzielnym przez 4

Wady braku wyrównania naturalnego: - Gorsza wydajność (procesor musi wykonywać 2 odczyty zamiast 1 w celu pobrania jednej danej)

Zad. 3

Procesor ustala to poprzez xor wartości dwóch ostatnich pożyczek

Zad. 4

```
// Zakładamy, że dana jest już zainicjowana
char* k = dana;
for (int i = 0; i<10; ++i) {
    ++k;
    if (*k == '0') {
        break;
    }
}
```

Zad. 5

Należy zamaskować wyjątek (choć ten domyślnie już taki jest) dzielenia przez 0, poprzez ustawienie bitu **ZM=1**.

```
sub esp, 4
fstcw word PTR [esp]
bts word PTR [esp], 2
fldcw word PTR [esp]
add esp, 4
```

Zad. 6

Jest to architektura Harvardzka. Rozwiązanie to umożliwia jednoczesne pobieranie rozkazu i odczyt/zapis danych lub wykonywanie rozkazu podczas wykonania potokowego.

Zad. 7

4 kanały $\cdot 2^4$ linii $\cdot (2+8)$ B = 640 B

Zad. 8

Chuj wie? Dzięki przerwaniom procesor nie marnuje energii na aktywne odpytywanie (sprawdzanie czy urządzenie lub dana są gotowe) a wykorzystuje ją na pożyteczne obliczenia.

Zad. 9

Procesor podczas wykonywania kodu napotkał na instrukcję, której nie rozumie. W praktyce błąd ten powstaje podczas uruchomienia aplikacji korzystającego z najnowyszch instrukcji na starszym procesorze (np.)

Zad. 10

Ściśle powiązane - wszystkie procesory współdzielą pamięć między sobą Luźno powiązane - każdy procesor posiada prywatną przestrzeń adresową

Zad. 11

ABI (Application Binary Interface) to zbiór reguł współpracy modułów oprogramowania na poziomie kodu binarnego, ustala on następujące zasady: - Jak przekazywane są parametry do procedur? - Kto usuwa przekazane parametry ze stosu w wypadku przekazywania ich tą drogą - Jakie rejestry muszą być zachowane po wykonaniu procedury? - W jaki sposób zwracany jest wynik?

API (Application Programming Interface) to zbiór reguł współpracy modułów oprogramowania na poziomie kodu źródłowego, ustalający np: - Ilość, typy i znaczenie przyjmowanych parametrów - Zwracana wartość - Rzucane wyjątki

Zad. 12

Przesunięcie logiczne to po prostu przesunięcie bitów, przesunięcie arytmetyczne to przesunięcie bitów w taki sposób, by najstarszy bit (bit znaku) pozostał bez zmian.

05.02.2020 (1 termin)

Zad. 1

R10: - R10B (najmłodszy bajt) - R10W (najmłodsze słowo = 2 bajty) - R10D (najmłodszy DWORD = 4 bajty)

```
push r10
mov eax, dword PTR [rsp+4]
pop r10
```

Zad. 2

BMP to punkty o kodach z zakresu $<0;FFFF>$, stąd potrzeba 16 bitów do zakodowania wszystkich.

Do zakodowania tych wartości w formacie UTF-8 potrzebujemy 3 bajtów.

Zad. 3

$\text{st}(1) - \text{st}(0) = 2.0$

ABCDEF: 02

ABCDF0: 00

ABCDF1: 00

ABCDF2: 00

Zad. 4

Skoro obydwa rozkazy zajmują 5 bajtów, to \$ wskazuje na $\text{alfa}+5$, stąd $\$-5 = \text{alfa}$.

$\text{EDX} = \text{EAX} = 0\text{xCAFEBABE}$

Zad. 5

```
mov ax, 0
mov es, ax
cli
mov es:[36], 1000h
mov es:[38], 2345h
sti
```

Zad. 6

Klawiatura wysyła kod wciśnięcia (make code) lub odcisnięcia (break code), a kontroler klawiatury przekształca tę informację na kod pozycji (scan code) wciśniętego lub odcisniętego przycisku i umieszcza go w porcie 0x60.

W przypadku odcisnięcia przycisku scan code jest powiększony o +128.

Zad. 7

Zwykły procesor ZAWSZE rzuci wyjątkiem i przerwie wykonywanie programu, natomiast zachowanie koprocesora jest zależne od jego konfiguracji (tj. wartości bitu ZM w rejestrze sterującym)

Zad. 8

Podczas wykonywania operacji blokowych, gdy chcemy by wykonywały się w pętli i jednocześnie zabezpieczyć przed sytuacją, gdzie przed wykonaniem pętli jej licznik jest = 0 (wtedy zwykły rep wykonałby operację $2^{32} - 1$ razy).

Zad. 9

Czas między kolejnymi cyklami zegara. Większość parametrów pamięci (takich jak czas między podaniem kolumny a odczytem czy częstotliwość odświeżania) podaje się właśnie w cyklach, żeby “skonwertować” te cykle na (mili/nano)sekundy należy przemnożyć je przez właśnie długość tego cyklu.

Zad. 10

Miss występuje co 16. iteracje pętli ($i=0,16,32,\dots,4080$)

Ilość missów: $\frac{4096}{16} = \frac{2^{13}}{2^4} = 2^9$

Hit rate: $\frac{2^{13}-2^9}{2^{13}} = 1 - \frac{2^9}{2^{13}} = 1 - \frac{1}{16} = \frac{15}{16} = 93,75\%$

Zad. 11

BTB przechowuje historię skoków, na podstawie której branch predictor przewiduje, czy procesor skoczy i odpowiednio ustawia potok wykonania.

Zad. 12

Hazard zasobów to sytuacja, w której jedna instrukcja potrzebuje danych, które są wynikiem innej, jeszcze niewykonanej instrukcji. np. klasyczny potok wykonania dla

```
mov eax, 2137h
add ebx, eax
```

Problem ten minimalizuje się poprzez odpowiednie szeregowanie: - Metoda statyczna wykonywana jest przez kompilator, ustawia wynikowe rozkazy w taki sposób, by ten nie nastąpił. (Okolo 70% skuteczności)

- Metoda dynamiczna wykonywana jest przez sam procesor podczas wykonywania kodu (tzw. Out-of-Order Execution)

5.02.2018

Zad. 1

0xFFFD

Zad. 2

Rozkaz wykonuje się w pętli, gdzie w każdej iteracji rejestr ECX jest dekrementowany. Pętla przestaje się wykonywać gdy ECX=0.

Jest to równoważne z:

```
et: rozkaz
loop et
```

Zad. 3

```
.data jakis_long_long dq ?
```

```
.code
...
add eax, [data]
adc edx, [data+4]
...
```

W ten sposób dodamy do EDX:EAX liczbę zawartą w `jakis_long_long`

Zad. 4

Przez prefix 0x66 występujący przed rozkazem działającym na 16-bitowych operandach.

Zad. 6

Assembler nie jest w stanie sprecyzować rozmiaru operandów. Rozwiązaniem jest użycie dyrektywy PTR z odpowiednim rozmiarem.

Zad. 7

FST zapisuje wartość z ST(0) w postaci zmiennoprzecinkowej, natomiast FIST skonwertowaną (poprzez zaokrąglenie) do postaci całkowitej w kodzie U2

Zad. 8

$$2^{32-12-4} = 2^{16} \text{ linii}$$

Zad.9

Szerokość magistrali jest taka sama w przypadku DDR2 i DDR3. Zatem przepustowość będzie taka sama (z pominięciem sytuacji typu jedna konfiguracja w single-channel, druga w dual-channel)

Zad. 10

Procesor z góry zakłada wynik skoków (np. zawsze się wykona, nigdy się nie wykona) lub w zależności od rodzaju porównania. Ta 3 metoda wykorzystywana z badaniami statystycznymi jest w stanie przynieść całkiem dobre rezultaty.

Zad. 11

Problemy związane z równoległością (hazardy) biorą przewagę nad zyskami wynikającymi z przetwarzania rozproszonego.