

1. Procesory Core 2 posiadają współczynnik IPC(Instruction Per Cycle) równy:
  - a) 4
  - b) 3,5**
  - c) 2,5
  - d) 2
  - e) 3
  
2. Wykonując instrukcję push byte ptr[eax] procesor używa segmentów
  - a) CS i DS ?**
  - b) ES i DS
  - c) DS i SS
  - d) SS i GS
  - e) FS i CS

Takie mamy rejestry w procesorach x86:

- f) CS – 16-bitowy rejestr segmentu kodu programu
  - g) DS – 16-bitowy rejestr segmentu danych
  - h) SS – 16-bitowy rejestr segmentu [stosu](#)
  - i) ES, FS, GS – 16-bitowe rejestry pomocnicze dla danych
  
3. Pośredni bazowy tryb adresowania występuje w instrukcji:
  - a) inc ecx
  - b) mov edi, offset zmienna
  - c) or edx, [ebx+tab]**
  - d) mov [edi\*8+tablica],edx
  - e) mov zmienna, edx
  
4. Która instrukcja nie jest poprawna:
  - a) Fcmovne
  - b) Fcmovnbe
  - c) Fcmovnu
  - d) Fcmovnb
  - e) Fcmovl**
  
5. Która z instrukcji nie zmienia flagi Z:
  - a) Ficom flagi C3 C2 C0**
  - b) Neg flagi OSZAPC
  - c) Dec flagi OSZAP
  - d) Cmp flagi OSZAPC
  - e) Xadd flagi OSZAPC

**W teście było zaznaczone neg!?**

6. Po wykonaniu instrukcji:

Mov ecx, 10023004h

Mov eax, 87654321h

Shr eax, cl -> przesuwam wszystkie bity A w prawo o ilość bitów zdefiniowaną przez B

Bsr edx, eax -> Skanuje src w poszukiwaniu pierwszego bitu równego 1. Ustawia Zero Flag (ZF=1) jeżeli znajdzie taki bit oraz zapisuje w dest nr pierwszego znalezione bitu. Czyści ZF (ZF=0) jeżeli żaden bit nie jest ustawiony. BSF skanuje od najmniej znaczącego bitu (0-n), podczas gdy BSR skanuje odwrotnie(n-0).

Rejestr edx będzie zawierał?

- a) 0
- b) 1
- c) 15
- d) 27**
- e) 32

Wyjaśnienie <http://wklej.org/hash/a701950baaa/>

7. Która z instrukcji nie zmienia flagi CF ?

- a) Cmc zanegowanie CF, flagi **C**
- b) Add dodawanie, flagi **OSZAPC**
- c) Stc ustawienie CF, flagi **C**
- d) Clc zerowanie CF, flagi **C**
- e) Inc flagi OSZAP**

8. Która z instrukcji nie umożliwia pomnożenia zawartości rejestru przez 2?

- a) Add
- b) Imul
- c) Sal
- d) Ucięte zdjęcie ☹

9. Kopiując tekst użyjemy instrukcji:

- a) Copysb
- b) Movsb**
- c) Lodsb
- d) Cpsb
- e) Stosb

10. Do przeszukiwania bitów wstecz służy instrukcja:

- a) bsr** Przeszukiwanie bitów wstecz
- b) btc Testowanie bitu z negacją
- c) test Porównanie logiczne
- d) lea Ładowanie adresu efektywnego
- e) bsf Przeszukiwanie bitów w przód
- f)

11. Jednostkę zarządzania pamięcią Intel wprowadził po raz pierwszy w procesorze:

- a) Pentium 4
- b) Pentium III
- c) 80486 DX
- d) 80386**
- e) 80286

12. Ile rejestrów segmentowych dołożono w trybie EMT64T procesorów Intel

- a) 2
- b) 4
- c) 6
- d) 8
- e) Żadne z powyższych**

**Nie ma architektury EMT64T!**

13. Ile rejestrów XMM posiadają w trybie EMT64T procesory Intel

- a) 8
- b) 16
- c) 32
- d) 64
- e) Nie ma takich rejestrów**

**Nie ma architektury EMT64T!**

14. Która instrukcja dla liczb bez znaku zmienia bajt na podwójne słowo:

- a) Xlatb rozkaz zwraca wartość elementu tablicy którego indeks znajduje się w al
- b) Movzx sample: movzx edx, al;**
- c) Cbw konwertuje bajt na słowo
- d) Cbdw konwertuje słowo na podwójne słowo
- e) Bswap zamienia bajty

15. Która z instrukcji dla liczb ze znakiem przesyła dla warunku mniejszość?

- a) Cmpbe
- b) Movl
- c) Movae
- d) Cmovnge – ani większe ani równe – czyli mniejsze**
- e) Cmovnae

16. Ile instrukcji na słowach może wykonać jedna instrukcja MMX

- a) 4**
  - b) 8
  - c) 16
  - d) 32
  - e) 64
- //2 – na podwójnych słowach  
//1 – na poczwórnych słowach

17. Napisz program obliczający sumę n cyfrowych niespakowanych liczb BCD

```
asm
{
    movecx, N;
    movedi, a;
    add edi, ecx;
    decedi;
    movesi, b;
    add esi, ecx;
    decesi;

    mov ax, 0;
    add al,al;
SumBCD:
    mov al, [esi];
    adc al, ds:[edi];
    mov ah, 0;
    aaa;
    mov ds:[edi], al;
    decesi;
    decedi;
    dececx;
    jnzSumBCD;

}
```

18. Oblicz  $y=e^x$

```
fld x ;          x
fldl2e ;         log2e; x
fmul ;          x*log2e
fld st(0) ;      x*log2e; x*log2e
frndint ;        round(x*log2e); x*log2e
fsub st(1), st ; round(x*log2e); x*log2e - round(x*log2e)
fxch st(1) ;     x*log2e - round(x*log2e); round(x*log2e)
f2xm1 ;          2^(x*log2e - round(x*log2e))-1; round(x*log2e)
fld1 ;           1,2^(x*log2e - round(x*log2e))-1; round(x*log2e)
fadd ;           2^(x*log2e - round(x*log2e)); round(x*log2e)
fscale ;         2^(x*log2e - round(x*log2e))*2^round(x*log2e)
                2^( x*log2e - round(x*log2e)+round(x*log2e))= 2^( x*log2e)
fstp y
```

19. Głównym konstruktorem procesorów 8086 był

- a) Steve Balmer
- b) Steve Jobs
- c) Steven Morse**
- d) Bill Gates
- e) Żadne z powyższych

20. Ile rejestrów XMM występuje w trybie EM64T procesorów Intel?

- a) 8
- b) 16**
- c) 64
- d) 256
- e) Żadne z powyższych

//MMX zawsze 8! XMM w arch. EM64T - 16, w IA32 - 8;

21. Natychmiastowy tryb adresowania występuje w instrukcji

- a) Inc ecx
- b) Mov edi, offset tabela**
- c) Mov Zmienna, edx
- d) Or Edx, [ebx]
- e) Mov [Edi\*8+tabela], edx

22. Po wykonaniu instrukcji

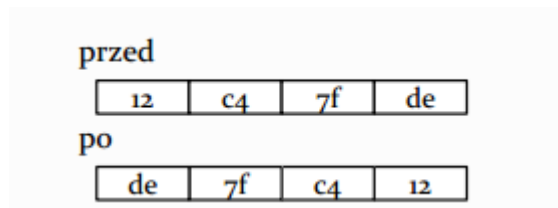
Mov edx, 0f459a01h

Bswap edx

Rejestr edx będzie zawierał

- a) 0f459a01h
- b) 9a010f45h
- c) 10a954f0h
- d) 019a450fh**
- e) Żadne z powyższych

Przykład użycia bswap:



23. Instrukcja Dec zmienia flagi:

- a) OZAPC
- b) OSAPC
- c) OSZPC
- d) OSZAC
- e) OSZAP**

24. Po wykonaniu instrukcji

```
Mov eax, 0f0f0f0fh
Mov edx, 12345678h
Not ax
Inc dx
And edx,eax
```

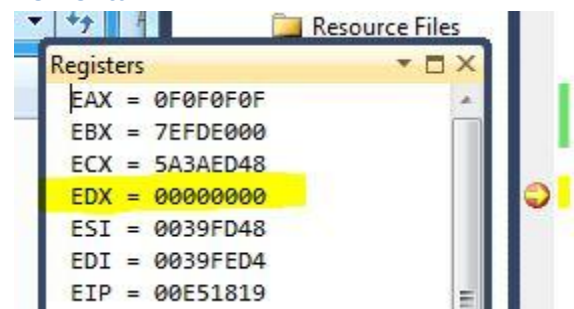
Rejestr edx będzie zawierał:

- a) **02045070h**
- b) 12345678h
- c) 10300608h
- d) 02040608h
- e) 02040609h

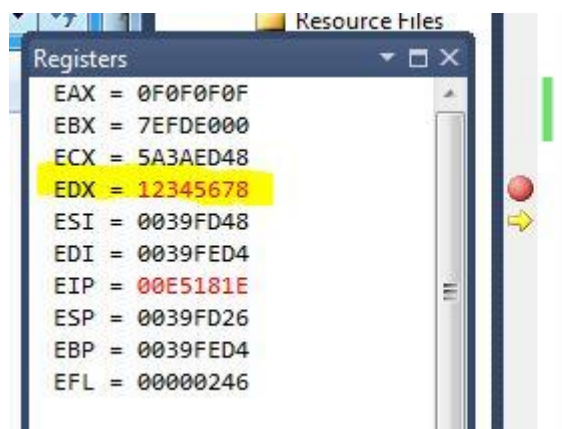
Program napisany przez Adriana wszystko wam wyjaśni (Trzeba sobie skompilować!)

<http://wklej.org/hash/359ede80ce6/>

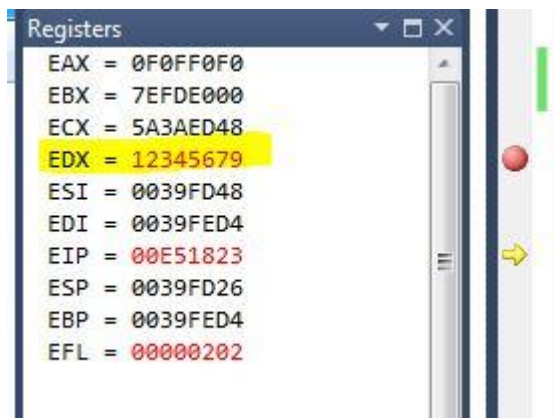
Komentarz:



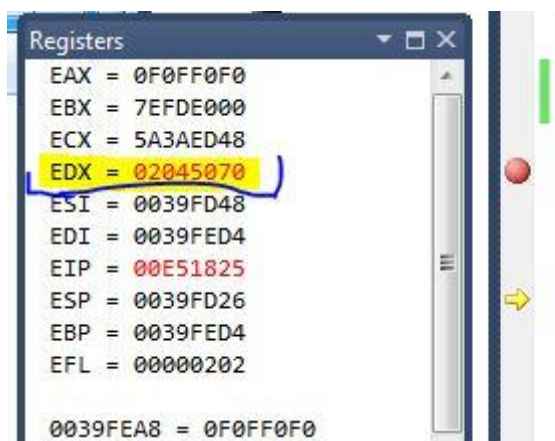
```
{
    pushf;
    pushad;
    xor edx, edx;
    Mov eax, 0f0f0f0fh;
    Mov edx, 12345678h;
    not ax;
    inc dx;
    and edx, eax;
}
```



```
{
    pushf;
    pushad;
    xor edx, edx;
    Mov eax, 0f0f0f0fh;
    Mov edx, 12345678h;
    not ax;
    inc dx;
    and edx, eax;
    mov rejestr_EAX, eax;
    mov rejestr_EDX, edx;
    popad;
    popf;
}
```



```
pushf;
pushad;
xor edx, edx;
Mov eax, 0f0f0f0fh;
Mov edx, 12345678h;
not ax;
inc dx;
and edx, eax;
mov rejestr_EAX, eax;
mov rejestr_EDX, edx;
popad;
popf;
```



```
pushf;
pushad;
xor edx, edx;
Mov eax, 0f0f0f0fh;
Mov edx, 12345678h;
not ax;
inc dx;
and edx, eax;
mov rejestr_EAX, eax;
mov rejestr_EDX, edx;
popad;
popf;
```

25. Napisz program obliczający  $\ln x$ . Wskaźnik do  $x$  znajduje się w `eax`, wynik pozostaw w rejestrze `ST(0)`.

```
double wynik2;
__asm
{
    fldl2e;           //log2(e)
    fld1;             //1, log2(e)
    fld x;            //x, 1, log2(e)
    fyl2x;            //log2(x), log2(e);
    fxch;             //log2(e), log2(x);
    fdiv;             //log2(x)/log2(e);lnx
    fstp wynik2;
}
cout<<"ln z 3: "<< wynik2 <<endl;
```

26. Napisz, przy użyciu instrukcji łańcuchowych program przesyłający 777 bajtów z tablicy `tab` do tablicy `tab2`.

```
Mov ecx, 777;
Mov esi, tab1;
Mov edi, tab2;
Rep movsb;
```

27. Napisz program umieszczający w eax zaokrągloną średnią eax i edx

28. Współczesne procesory Intel i7 zbudowane są z około

- a) 100 tys, tranzystorów
- b) 1 mln tranzystorów
- c) 10 mln tranzystorów
- d) 100 mln tranzystorów
- e) **Żadne z powyższych**

Zbudowane są z około **770mln** tranzystorów

29. Do odwołania się do zmiennych lokalnych stosuje się rejestr:

- a) Eax
- b) Ebx
- c) **Ebp**
- d) Esi
- e) Ds

30. Która z instrukcji nie jest poprawna:

- a. Fmul
- b. fadd ST, ST(1)
- c. **faddp ST, ST(1)**
- d. fsubr ST(1),ST
- e. fsubrp ST(1),ST

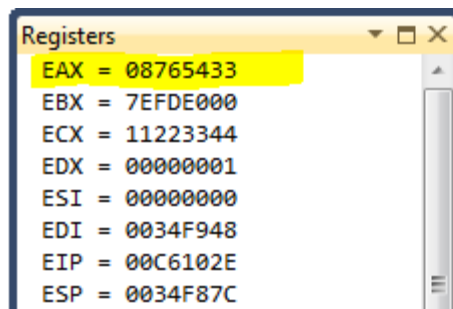
31. Która z instrukcji zmienia flagę C:

- a) Bswap
- b) Dec
- c) Ewd
- d) Fmul
- e) **Fcomi**

32. Po wykonaniu instrukcji

```
mov ecx, 11223344h;  
mov eax, 87654321h;  
shr eax, cl;  
bts eax, 0;
```

- a) 08765433h
- b) 08765432h
- c) 043b2a190h
- d) 0e3b2a190h
- e) 021436587h



asm

```
{  
    mov ecx, 11223344h  
    mov eax, 87654321h  
    shr eax, cl;  
    bts eax, 0  
}
```



33. Która z instrukcji wpisuje 0 do flagi CF?

- a) cmc
- b) eld
- c) stc
- d) clc**
- e) vld

34. Która z instrukcji umożliwia dodanie trzech wartości?

Strasznie słabo czytelne ☹

Poprawna jest chyba lea

35. Która z instrukcji pozwala na poszukiwanie podanego znaku w tekście?

- a) Scasb**
- b) Movsd
- c) Lodsw
- d) Cmpsb
- e) Stosw

36. Prefix LOCK może odnosić się do instrukcji

- a) Nop
- b) Xchg**
- c) Cmp
- d) Mov
- e) Fdiv

37. Instrukcje AVX Intel wprowadził po raz pierwszy w procesorze:

- a) Pentium II
- b) Pentium 4
- c) Pentium D
- d) Sandy Bridge**
- e) Ivy Bridge

38. Ile rejestrów XMM posiadają trybie EM64T procesory Intel

- a) 8
- b) 16
- c) 32
- d) 64
- e) Nie ma takich rejestrów**

Taka architektura nie istnieje, jest **EM64T** wtedy poprawną odpowiedzią byłaby 16

39. Do prostego szyfrowania danych może służyć instrukcja

- a) Movsd
- b) Btr
- c) Cwde
- d) Xlatb**
- e) Les

40. Która z instrukcji dla liczb ze znakiem przesyła dla warunku większości ?

- a) Movg
- b) Movae
- c) Cmovnle**
- d) Cmoka
- e) Cmpa

41. Do zmiany kolejności słów w rejestrze MMX służy instrukcja:

- a) Ldmxesr
- b) Pmovmskb
- c) Cmovnge
- d) Psadbw
- e) Pushfw**

42. Napisz program obliczający średnią arytmetyczną z zawartości rejestrów eax i edx

43. Oblicz  $y = x^a$  dla  $y, x, a$  typu double

```
fld a; // a
fld x; // x, a
fyl2x; // y*log2x, x, a
fld st; // y*log2x, y*log2x, x, a
frndint; // round(y*log2x), y*log2x, x, a
fsub st(1), st; // round(y*log2x), y*log2x-round(y*log2x)
fxch; // y*log2x-round(y*log2x), round(y*log2x)
f2xm1; // 2^y*log2x-round(y*logwx)-1, round(y*logw2x)
fld1; // 1, 2^y*log2x-round(y*logwx)-1, round(y*logw2x)
fadd; // 2^y*log2x-round(y*logwx), round(y*logw2x)
fscale; // x^a, cx+d
fstp y
```

44. Wykonanie instrukcji

Push [edx]

Pop [ebx]

Odpowiada instrukcji:

- a) Xor [edx], [ebx]
- b) Xchg [edx],[ebx]
- c) Xchg edx,ebx
- d) Fxch [edx],[ebx]
- e) Żadne z powyższych**

45. Instrukcja włączająca przerwania to:

- a) Sti - set interrupt**
- b) Cmc
- c) Cmp
- d) Stp
- e) Cli

46. Po wykonaniu instrukcji

Mov edx, 0f459101h

Bswap edx

Rejestr edx będzie zawierał:

- a) 0f459a01h
- b) 9a010f45h
- c) 10a95410h
- d) 0191450fh**
- e) Żadne z powyższych

47. Instrukcja Dec zmienia flagi

- a) OSZAPC
- b) OSAPC
- c) OSZPC
- d) OSZAC
- e) OSZAP**

48. Po wykonaniu której instrukcji zawartość AL. Zmieni się 0c9h na 92h?

- a) Or al,92h
- b) Neg al.
- c) Rol al,1
- d) Shl al,1**
- e) Mov 92h, al

C9 – 11001001

92 – 10010010 <- doszło zero, rol dodałoby jedynkę z początku

<http://wklej.org/hash/360e57da98c/> - przykłady tych instrukcji na bitach

49. Podaj instrukcję MMX pakującą z nasyceniem słowa bez znaku do bajtów

**Pckuswb**

50. Napisz program obliczający  $\log_a x$ . Wskaźniki do a oraz x znajdują się w eax i edx wynik pozostaw w rejestrze ST(0).

```
Fld1;  
Fld Word ptr[edx];  
Fyl2x;  
Fld1;  
Fld Word ptr[eax];  
Fyl2x;  
Fdiv;
```

51. Napisz przy użyciu instrukcji łańcuchowych program przesyłający 777 bajtów z tablicy tab1 do tablicy tab2

```
Mov ecx, 777;  
Mov esi, tab1;  
Mov edi, tab2;  
Rep movsb;
```

52. W którym procesorze Intel po raz pierwszy zastosował tryb chroniony

- a) 8086
- b) 80286**
- c) 80386
- d) Pentium Pro
- e) Żadne z powyższych