


- 1.** Ktorá z uvedených rovností je korektným vyjadrením súčtu kladného a záporného čísla.

~~a)~~ Ak $|x| > M$ potom $[(x+y_i)] = [(x+y)_i]$

M - память (memory)

✓  Ak $|x| < M$ potom $[(x+y)_i] = [(x+y)_i]$

- 2.** Prenos z najvyššieho bitu výsledku aritmetickej operácie je indikovaný stavovým príznakom.

- a) SF

- b) ZF

- c) CF • Carry (Cy) – prenos z najvyššieho bitu výsledku operácie (1 – prenos nastal)

- d) PF

- 3.)** Po realizácii dvojice inštrukcii

mov AL,10101110b

xor AL, 11110000b

bude register AL obsahovať hodnotu:

- a) 00H

- b) AOH

- c) 5EH

XOR - 2 equn. $\rightarrow 0$
2 page. $\rightarrow 1$

$$\begin{array}{r} 10101110 \\ 11110000 \\ \hline 01011110 \\ 41842 \\ 5 \quad 4 \\ e \end{array}$$
 → для перевода в 16-ю систему
 группируем по 2 знака,
 переводим в 16-ю.

10 - а
 11 - в

$$\begin{array}{r} 01011110 \\ 21 \quad 42 \\ 1 \quad 3 \quad 6 \\ 126 \end{array}$$

4. Inštrukciu aritmetických posunov **sal** je možné využiť na:

- a) Násobenie čísel bez znamienka

- b) Delenie čísel bez znamienka

- c) Delenie čísel so znamienkom

- d) Násobenie čísel so znamienkom

- dva typy posunov

- logické (čísla bez znamienka) – shl, shr
- aritmetické (čísla so znamienkom) – sal, sar

- aritmetické (číslo so znamienkom) – s_{al} , s_{ar}

5. Aký stav je signalizovaný stavovým príkazom SF registra EFLAGS procesora Pentium?

- a) Výsledok aritmetickej operácie na bezznamienkových číslach

- b) Výsledkom poslednej aritmetickej operácie bola 0

- c) Výsledkom poslednej aritmetickej operácie bola záporná hodnota

- d) Výsledok aritmetickej operácie na číslach so znamienkom prekročil rozsah.

- 6. Prístup k parametrom v zásobníkovom rámci procedúry je realizovaný**

- a) Odpočítaním príslušnej vzdialenosti vzhľadom na obsah EBP (napr. EBP-8)

- b) Pripočítaním príslušnej vzdialenosti vzhľadom na obsah EBP (napr. EBP+8)

- prístup k obsahu zásobníkového rámca – EBP (frame pointer)

- ⊛ parametre (a, b - EBP+12, EBP+8)

- lokálne premenné (temp, N – EBP-4, EBP-8)

7. Ak výsledkom poslednej operácie (ovplyvňujúcej ZF) bola 0 príznak ZF bude:

a) Nastavený na 0, ZF = 0

Príznak nuly (zero flag)

b) Nastavený na 1, ZF = 1

• výsledkom poslednej operácie (ovplyvňujúcej ZF) bola 0 – ZF = 1, ináč ZF = 0

c) Nedefinovaný

8. Organizácia pamäti v reálnom režime procesorov x86 – môže viac ako jedna ... referovať na rovnakú fyzickú adresu?

a) Áno

b) Nie

- logická vs. fyzická adresa
 - pre každú logickú adresu existuje jedinečná fyzická
 - opačne to neplatí – viac ako jedna logická adresa môže referovať na rovnakú fyzickú adresu

9. Medzi podmienené skoky realizované podľa znamienkových porovnaní patria:

a) jl

b) jnge

c) je

d) jg

Skoky podľa výsledku znamienkových porovnaní

- porovnania =, ≠ pracujú rovnako na číslach so znamienkom i bez znamienka
- pre čísla so znamienkom relevantné SF, OF, ZF

mnemonika	je/jz	jne/jnz	jg/jnle	jge/jnl	jl/jnge	jle/jng
význam	equal/ zero	not equal/ not zero	greater/ not less or equal	greater or equal/ not less	less/ not greater or equal	less or equal/ not greater
podmienka	ZF = 1	ZF = 0	ZF = 0 AND SF = OF	SF = OF	SF ≠ OF	ZF = 1 OR SF ≠ OF

10. Inštrukcia AND realizujúca operáciu logického súčinu sa využíva na:

~~a)~~ Podpora zložených log. výrazov a bitová operácia log. súčinu HL jazykov

b) Nulovanie bitov

c) Izolácia bitov

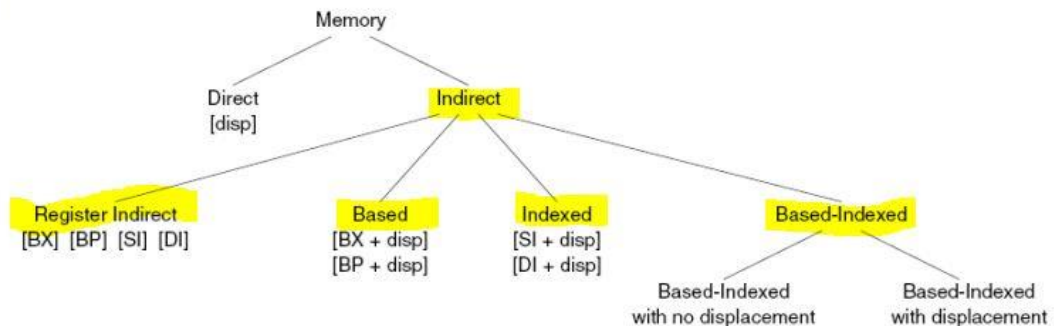
inštrukcia and

- podpora zložených log. výrazov a bitová operácia log. súčinu HL jazykov (neskôr)
- nulovanie bitov
- izolácia bitov

11. Medzi pamäťové nepriame režimy adresovania procesorov Pentium patria:

- a) Bázovo-indexové (Bassed-indexed)
- b) Bázové
- c) Indexové
- d) Nepriame registrové ??

16-bit adresy



12. Aký režim adresovania procesorov Pentium je použitý v rámci ... **add**

DX,[class_marks+EBX+ESI*2]

- a) Bázové adresovanie
- b) Bázovo-indexové adresovanie
- c) Indexové
- d) Nepriame registrové

13. Nulový výsledok aritmetickej operácie je indikovaný stavovým ...

14. Sémantika inštrukcie **pop dst32** je vyjadrená:

- a) $Dst32 \leftarrow [SS:ESP], ESP \leftarrow ESP-4$
- b) $Dst32 \leftarrow [SS:ESP], ESP \leftarrow ESP+4$
- c) $Dst32 \leftarrow [SS:EBP], ESP \leftarrow ESP-4$
- d) $Dst32 \leftarrow [SS:EBP], ESP \leftarrow ESP+4$

◦ sémantika

push src16	$ESP \leftarrow ESP - 2, [SS:ESP] \leftarrow src16$
push src32	$ESP \leftarrow ESP - 4, [SS:ESP] \leftarrow src32$
pop dst16	$dst16 \leftarrow [SS:ESP], ESP \leftarrow ESP+2$
pop dst32	$dst32 \leftarrow [SS:ESP], ESP \leftarrow ESP+4$

15. Základné typy operandov v jazyku procesorov pentium zahŕňajú

- a) Implicitný Adresovanie operandov (metóda sprístupnenia operandov)
- b) Registrový
 - **implicitné** (implied) – dané funkciou inštrukcie (STC, DAA)
- c) Pamäťový
 - **registrové** (register) – operand v registri, druhý implicitný operand – akumulátor (CMP, ADD), RP
- d) Bezprostredný
 - **bezprostredné** (immediate) – údaj časťou inštrukcie (za operačným kódom, ADI, CPI, MVI, LXI)
 - **priame** (direct) – priama adresa (16-bitov) súčasťou inštrukcie (JMP, LDA, STA)
 - **nepriame** (register indirect) – pamäťová referencia v RP (LDAX, STAX)

16. Vzdialenosť cieľa podmienených skokov (short) môže nadobúdať hodnoty z rozsahu:

- a) -32768/+32767 B
- b) 0/255 B
- c) -128/+127 B
- d) 0/65535 B

Vzdialenosť cieľa podmienených skokov

- podmienené skoky – SHORT/NEAR (najefektívnejšie, ak sú kódované ako 2B inštrukcie – SHORT)
- rozsah -128/127 B (SHORT)

17. FPU procesorov Pentium podporuje formáty čísel s pohyblivou rádovou čiarkou s veľkosťou:

- a) 64 bitov
 - b) 80bitov
 - c) 32 bitov
 - d) 16bitov
 - e) 128 bitov
- FPU procesora Pentium podporuje 3 formáty čísel:
 - čísla s jednoduchou presnosťou (32-bit, float v C)
 - čísla s dvojitou presnosťou (64-bit, double v C)
 - rozšírený formát (80-bit, interný)

18. Prístup k lokálnym premenným v zásobníkovom rámci procedúry je realizovaný:

- a) Pripočítaním príslušnej vzdialenosti vzhľadom na obsah EBP (napr. EBP+8)
- b) Odpočítaním príslušnej vzdialenosti vzhľadom na obsah EBP (napr. EBP-8)

- prístup k obsahu zásobníkového rámca – EBP (frame pointer)
 - parametre (a, b – EBP+12, EBP+8)
 - lokálne premenné (temp, N – EBP-4, EBP-8)

19. Smer spracovania reťazcov pomocou reťazových inštrukcii jazyka procesora Pentium je určený hodnotou príznaku:

- a) CF
 - b) TF
 - c) OF
 - d) DF
- smer spracovania – vpred/vzad (direction flag, DF)

20. Aký indexový register pri pamäťových režimoch adresovania procesorov Pentium možno v 16-bit režime použiť:

- a) EDI
 - b) ESI
 - c) BX
 - d) IP
 - e) BP
- indexové (index registers) – ESI, EDI (aj ich 16-bitové subregistre – SI, DI; reťazcové operácie)

add	+
sub	-
mul	*
div	/

21. Inštrukcia loop využíva pri svojej činnosti register:

- a) ECX
- b) EBP
- c) EBX
- d) ESI

22. Syntax zápisu inštrukcii používaná prekladačom gcc pre in-line (vložený) assembler je:

- a) AT&T
- b) IBM
- c) Intel

In-line (vložený) assembler

- asm-priказы vložené v C-kóde, použitie konštrukcie `asm`
- problém: syntax používaná prekladačom gcc (AT&T) je odlišná od syntaxe v NASM (Intel)

23. Pre násobenie čísel bez znamienka, je v jazyku procesora Pentium určená inštrukcia:

- a) idiv
 - b) imul
 - c) div
 - d) mul
- násobenie čísel bez znamienka (`mul`)
 - syntax
- ```
mul src (src - 8, 16, 32-bit GPR, pamäť)
```

24. Inštrukciu logických posunov `shl` je možné využiť na:

- a) Delenie čísel so znamienkom
  - b) Delenie čísel bez znamienka
  - c) Násobenie čísel so znamienkom
  - d) Násobenie čísel bez znamienka
- dva typy posunov
    - logické (čísla bez znamienka) – `shl, shr`
    - aritmetické (čísla so znamienkom) – `sar, sar`

25. Koľko prvkov obsahuje abeceda ( $V_r$ ) pre generovanie numerálov pozičných sústav so základom  $r$ ?

### **Reprezentácie čísel**

- a)  $r+1$
  - b)  $r$
  - c)  $r-1$
- použitie pozičných sústav s rôznym základom  $r = 2, 8, 10, 16$  ( $r$ -numerály)
  - abeceda:  $V_r = \{0, 1, \dots, (r-1)\}$

26. Ako bude nastavený príznak SF po realizácii uvedeného fragmentu kódu?

**mov EAX,15**

**sub EAX,97**

```
mov EAX, 15
sub EAX, 97 (15 - 97 = -82, SF = 1)
```

- a) SF = 0
- b) SF = 1

27. Uvedený fragment kódu je jadrom procedúry:

```
mov edi, src
```

```
mov ecx, 0FFFFFFFh
```

```
xor al,al
```

```
cld
```

```
repnz scasb
```

- a) `_asm_strlen` (určenie dĺžky reťazca)
- b) `_asm_find` (hľadanie zadaného bajtu v pamäti)
- c) `_asm_copy` (kopírovanie bloku pamäti)

28. Pre systémové volania OS Linux je vyhradený vektor prerušenia:

- a) 0x21 (21H)
- b) 0x16 (16H)
- c) 0x80 (80H)
- d) 0x20 (20H)

- parametrizácia (napr. systémové volania Linuxu int 0x80, cca. 180 volaní (v závislosti od verzie), číslo volania v EAX)

29. Sémantika inštrukcie iterácii loop je vyjadrená:

- a) `ECX=ECX-1, IF (ECX ≠ 0 AND ZF = 0) skok na cieľ`
- b) `ECX=ECX-1, IF (ECX ≠ 0 AND ZF = 1) skok na cieľ`
- c) `ECX=ECX-1, IF (ECX = 0) skok na cieľ`
- d) `ECX=ECX-1, IF (ECX ≠ 0) skok na cieľ`

```
ECX = ECX - 1
if ECX ≠ 0 then EIP ← label
```

30. Direktíva EXTERN prekladača NASM:

- a) Sprístupní návestie iným modulom programu
- b) Informuje assembler, že návestie nie je definované v aktuálnom module

#### Direktíva EXTERN

- informuje assembler, že návestie nie je definované v aktuálnom module (definované v inom)

31. Pri programovaní procesora i8080 možno využiť tieto registrové páry:

- a) BE
  - b) AL
  - c) HL
  - d) BC
- registrové páry (BC, DE, HL, PSW)

32. Sémantika inštrukcie nepodmieneného skoku `jmp label` je vyjadrená prenosom:

- a) `ESI <- label`
  - b) `EIP <- label`
  - c) `EBX <- label`
  - d) `EBP <- label`
- nepodmienený skok – JMP
- odovzdanie riadenia na návestie „label“

```
jmp label (EIP ← label)
```

stack  
reg na číslach



33. Ako bazový register pri pamäťových nepriamych režimoch adresovania procesorov Pentium možno v 32-bit režime použiť:

- a) EBP
- b) EAX ??
- c) **EBX**
- d) EIP
- e) ESI

|               | 16-bit addressing | 32-bit addressing                               |
|---------------|-------------------|-------------------------------------------------|
| Base register | BX<br>BP          | <b>EAX, EBX, ECX, EDX</b><br>ESI, EDI, EBP, ESP |

34. Aký bude obsah FPU zásobníka po vykonaní inštrukcie **fadd** nad zásobníkom s hodnotami 5.0, 2.0, 3.0, 8.0 (vrchol zásobníka predpokladáme vľavo)?

- a) ~~7.0, 2.0, 3.0, 8.0~~
- b) 5.0, 7.0, 3.0, 8.0
- c) **7.0, 3.0, 8.0**
- d) 3.0, 7.0, 8.0

`fadd` (výber ST0 a ST1, sčítanie a uloženie sumy späť na FPU zásobník)

35. Segmentový deskriptor používaný v chránenom režime (protected mode) procesora Pentium obsahuje tieto informácie:

- Segmentový deskriptor**
- a) **Veľkosť segmentu (segment limit)**
  - b) **Stavové a radiace informácie**
  - c) **Bázová adresa (base address)**
- poskytuje atribúty segmentu:
  - bázová adresa (32-bit) – začiatok segmentu vo fyzickom adresnom priestore
  - veľkosť segmentu (20-bit), dve interpretácie
    - 1B – 1MB ( $2^{20}$ B), krok 1B ( $G = 0$ )
    - 4KB – 4GB, krok 4KB ( $G = 1$ )
  - stavové a radiace informácie

36. Pod pojmom pamäť chápeme usporiadanú postupnosť registrov (buniek) veľkosti:

- a) 2 bajty
- b) 8 bajtov
- c) 4 bajty
- d) **1 bajt**

Pamäť

- usporiadaná postupnosť registrov (buniek) veľkosti 1 bajt (Byte)

37. Programátorovi sú v jazyku procesora i8080 priamo dostupné (napr. pre účely vetvenia programu) tieto stavové príznaky:

- a) ZF
- b) PF
- c) ~~AF~~
- d) **SF**
- e) CF

- 5 príznakov (S, Z, Ac, P, Cy), programovo dostupné 4 z nich

38. Natívnym režimom procesora Pentium s podporou segmentácie i stránkovania pamäti je:

- a) Reálny režim (real mode)
- b) **Chránený režim (protected mode)**

- chránený režim (protected mode) – **natívny** režim, podpora segmentácie i stránkovania

39. Určte súčet dvoch záporných čísel  $(x+y)$  v doplnkovej reprezentácii ak  $x=-542$ ,  $y=-781$ ,  $r=10$ ,  $n=3$ .

- a) 677
- b) 8676
- c) 8677
- d) 676

40. Klasifikácia prerušení procesorov Pentium:

- a) Systémové volania
  - b) Softvérové
  - c) Hardvérové
  - d) Výnimky
- HW a SW prerušenia, výnimky, systémové volania (Linux)

41. Aký bude obsah FPU zásobníka inštrukcie **fmul** nad zásobníkom 5.0, 2.0, 3.0, 8.0

- a) 10.0, 2.0, 3.0, 8.0
- b) 5.0, 10.0, 3.0, 8.0
- c) 3.0, 8.0, 10.0

**fmul** (výber ST0 a ST1, výpočet ST0 \* ST1, uloženie späť)

42. Pri násobení dvoch  $n$ -bitových čísel bez znamienka, rozmer výsledku môže ...

- a)  $n+2$  bitov
  - b)  $2n$  bitov
  - c)  $2n+1$  bitov
  - d)  $n+1$  bitov
- rozmer výsledku ( $2n$  bitov pri násobení dvoch  $n$ -bit. čísel)

43. Na akej adrese bude uložený bajt (CD) hodnoty ABCDH (údaj veľkosti...) ...byť údaj uložený od adresy 100?

- a) 101
- b) 100

44. Pri výskyte prerušenia v chránenom režime procesorov Pentium, do zásobníka ...

- a) EFLAGS, DS, ESI
- b) IDTR, ES, EDI
- c) IDTR, DS, ESI

~~EFlags, CS, EIP~~

45. Ktorá z uvedených rovností je korektným vyjadrením súčtu čísel v doplnkovej reprezentácii..

- a)  $(x_c + y_c) \bmod r^{n+1} + 1 = (x+y)_r$
- b)  $(x_c + y_c) \bmod r^{n+1} = (x+y)_{r-1}$

$(x_c + y_c) \bmod r^{n+1} = (x+y)_r$

46. Koľko bajtov pamäti bude vyhradených použitím direktívy: **buffer resw 100**

- a) 200
- b) 400
- c) 800
- d) 100

Neinicializované údaje:

- direktívy alokácie (bez inicializácie)
  - RESB (reserve byte, 1B)
  - RESW (2B)
  - RESD (4B)
  - RESQ (8B)
  - REST (10B)



47. Organizácia pamäti v reálnom režime procesorov x86 – môže viac ako jedna logická adresa referovať na rovnakú fyzickú adresu?

- a) **Áno** • logická vs. fyzická adresa
- b) Nie
  - pre každú logickú adresu existuje jedinečná fyzická
  - opačne to neplatí – viac ako jedna logická adresa môže referovať na rovnakú fyzickú adresu

48. Určte inverznú reprezentáciu čísla X, ak  $X = -752$ ,  $r = 10$ ,  $n = 3$

Odpoveď: **0100001111**

49. Medzi indexové registre procesora Pentium patria:

- a) EBX
- b) **EDI**
- c) EBP ◦ *indexové* (index registers) – ESI, EDI (aj ich 16-bitové subregistre – SI, DI; reťazcové operácie)
- d) **ESI**

50. V rámci štandardnej C-konvencie, pre odovzdanie celočíselnej návratovej hodnoty funkcie s pohyblivou rádovou čiarkou je využívaný:

- a) Register EBX
- b) Zásobník
- c) **Register EAX**
- d) Register ESI

51. Inštrukcia **test** nedeštruktívne (bez zmeny cieľového operandu) realizuje operáciu:

- a) Logického súčtu
- b) Logickej negácie bitov • inštrukcia **TEST**
- c) **Logického súčinu** ◦ ak je problémom modifikácia operandu
- d) Logickej nonekvivalencie ◦ vykonáva bitovú operáciu AND (ako inštrukcia **and**),

52. Aký indexový register pri pamäťových nepriamych režimoch adresovania procesorov Pentium možno v 32 bitovom režime použiť:

- a) **EDI**
- b) EBP
- c) **ESI**
- d) EIP
- e) EBX

53. Exponent vyjadrujúci hodnotu 32 (20H) bude (v rámci jednotky FPU) na 8-mich bitoch v kóde s posunutou nulou reprezentovaný hodnotou:

- a) 5FH
- b) 7FH
- c) **9FH**
- d) **3FH**

54. Dvojica inštrukcii

**shr EDX,1**

**rcr EAX,1**

realizuje:

- a) rotáciu EDX:EAX o 1 bit vpravo
- b) nulovanie dolných 32-bitov EDX:EAX
- c) nulovanie horných 32 bitov EDX:EAX
- d) posun EDX:EAX o 1 bit vpravo

55. Ako bude nastavený príznak SF po realizácii uvedeného fragmentu kódu?

**mov EAX,15**

mov EAX, 15

**add EAX,97**

add EAX, 97 (15 + 97 = 112, SF = 0)

- a) SF = 0
- b) SF = 1

56. Medzi reťazcové inštrukcie v jazyku procesora Pentium patria:

- a) SCASB
- b) ADDSB
- c) CMPSD
- d) MODSW
- e) LODSB
- f) STOSW

| Mnemonic | Meaning         | Operand(s) required  |
|----------|-----------------|----------------------|
| LODS     | LOaD String     | source               |
| STOS     | STORe String    | destination          |
| MOVS     | MOVE String     | source & destination |
| CMPS     | CoMPare Strings | source & destination |
| SCAS     | SCAn String     | destination          |

57. Obsahy koľkých registrov procesora Pentium uloží na zásobník inštrukcia **pushad**?

- a) 4
- b) 6
- c) 8 pushad (EAX, ECX, EDX, EBX, ESP, EBP, ESI, EDI) 2
- d) 10

58. Ktorá časť (časti) symbolického systému je špecifikovaná pomocou generujúceho systému?

- a) Syntax Generujúce systémy (gramatika)
- b) Abeceda • konečná množina symbolov (abeceda – V)
- c) Sémantika • konečná množina pravidiel ( $p: u\alpha v \rightarrow u\beta v$ )
  - $u, v, \alpha, \beta$  – reťazce nad abecedou V

59. Radič zariadenia (IO controller) v úlohe rozhrania počítač/zariadenie zabezpečuje prispôsobenie:

- a) Rýchlosti
- b) Úrovni el. signálov
- c) Súboru inštrukcii procesora

- radič zariadenia (I/O controller) v úlohe rozhrania počítač/zariadenie (rýchlosť, prispôsobenie úrovni el. signálov)

60. Pri spracovaní prerušení v chránenom režime procesorov Pentium, číslo prerušenia predstavuje index do tabuľky:

- a) GDT • číslo (typ) prerušenia (vector, 0-255) – index do tabuľky s adresami ISR
- b) LDT • tabuľka IDT (Interrupt Descriptor Table), číslo prerušenia \* 8 → index do tabuľky (položka – descriptor, 8B)
- c) IDT
- d) FDT

61. Pri volaní (knižničných) funkcií jazyka C z assembleru, ich parametre je potrebné uložiť do:

- a) ~~Registra ESI~~
- b) Registra EBX
- c) Registra EAX
- d)  Zásobníka

62. Poradie v akom sú parametre C-funkcii ukladané na zásobník je:

- a) Zľava doprava • **Odovzdávanie parametrov**
- b) ~~Sprava doľava~~ • odovzdávanie parametrov (v akom poradí sú parametre ukladané na zásobník)
  - zľava doprava (left-pusher languages, väčšina HL jazykov)
  - sprava doľava (right-pusher, napr. jazyk C – predmet nášho záujmu)
- c) Podľa typu param...

63. Bit D segmentového deskriptora CS určuje adresový režim (16/32-bit), ktorý má CPU použiť. Je možné túto implicitnú voľbu explicitne zmeniť?

- a) ~~Áno~~ • bit D segmentového deskriptora CS (D = 1: 32-bit default)
- b) Nie • možnosť implicitnú voľbu explicitne zmeniť (size override prefix)

64. Sémantika inštrukcie LDAX B jazyka i8080 je vyjadrená nasledujúcim prenosom:

- a) ~~A<-[BC]~~
- b) [HL]<-B
- c) B<-[HL]
- d) [BC]<-A

65. Aké budú obsahy registrov EAX, EBX po vykonaní fragment kódu

**Push EAX**

**Push EBX**

**Pop EAX**

**Pop EBX**

Ak pred jeho vykonaním platilo EAX = 1, EBX = 2?

- a) EAX = 2, EBX = 2
- b) EAX = 1, EBX = 1
- c) EAX = 1, EBX = 2
- d) **EAX = 2, EBX = 1**

66. Po realizácii dvojice inštrukcii jazyka i8080 bude register A obsahovať hodnotu:

**mvi A,10101110b**

**ani 11110000b**

- a) 10H
- b) 0AH
- c) 00H
- d) A0H

ani 11 1  
or 1 1  
xor 11 00 0

67. Ktoré z uvedených dvojíc inštrukcii jazyka procesora Pentium majú rovnakú sémantiku?

- a) jl, jng
- b) jg, jnle
- c) jle, jnge
- d) jge, jnl

| mnemonika | je/jz          | jne/jnz                | jg/jnle                       | jge/jnl                       | jl/jnge                       | jle/jng                       |
|-----------|----------------|------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|
| význam    | equal/<br>zero | not equal/<br>not zero | greater/<br>not less or equal | greater or equal/<br>not less | less/<br>not greater or equal | less or equal/<br>not greater |

68. Exponent vyjadrujúci hodnotu 48 (30H) bude (v rámci jednotky FPU) na 8-mich bitoch v kóde s posunutou nulou reprezentovaný hodnotou:

- a) 9FH
- b) 7FH
- c) 0AFH
- d) 0CFH

69. Klasifikácia výnimiek procesorov Pentium:

- a) Trap (pasca)
- b) Fault (porucha)
- c) Nemaskovateľné (non-maskable)
- d) Abort (zlyhanie)
- e) Maskovateľné (maskable)

#### Výnimky

- klasifikované do 3 skupín (faults, traps, aborts)

70. Medzi podmienené skoky realizované podľa **bezznamienkových** porovnaní v jazyku procesora Pentium patria:

- a) jnae
- b) jle
- c) jae
- d) jge
- e) jnb
- f) jb

| mnemonika | je/jz          | jne/jnz                | ja/jnbe                      | jae/jnb                      | jb/jnae                      | jbe/jna                      |
|-----------|----------------|------------------------|------------------------------|------------------------------|------------------------------|------------------------------|
| význam    | equal/<br>zero | not equal/<br>not zero | above/<br>not below or equal | above or equal/<br>not below | below/<br>not above or equal | below or equal/<br>not above |
| podmienka | ZF = 1         | ZF = 0                 | CF = 0 AND ZF = 0            | CF = 0                       | CF = 1                       | CF = 1 OR ZF = 1             |

71. Aký bude obsah FPU zásobníka po vykonaní inštrukcie **fstp ST1** nad zásobníkom s hodnotami 5.0, 2.0, 3.0, 8.0?

- a) 7.0, 3.0, 8.0     `fstp     ST1`     ; odstráni ST1, ST0 ponechá na vrchole
- b) 5.0, 5.0, 3.0, 8.0
- c) 2.0, 2.0, 3.0, 8.0
- d) 5.0, 3.0, 8.0

72. Reštart (výnimku generujúcej) inštrukcie je v architektúre Pentium realizovaný v prípade výnimky typu:

- a) Abort
- b) Trap
- c) Fault

◦ **fault** (porucha) – výnimočný stav, ktorý všeobecne možno napraviť v rámci obsluhy

- v prípade úspešnej nápravy program reštartovaný bez straty kontinuity
- obnova stavu do bodu pred vykonaním (výnimku generujúcej) inštrukcie (uchované CS:EIP ukazujúce na danú inštrukciu)

73. Pre delenie čísel so znamienkom, je v jazyku procesora Pentium určená inštrukcia:

- a) **Idiv**
- b) `Mul     div     src     (bezznamienkové, src – 8, 16, 32-bit GPR, pamäť)`
- c) `Duv     idiv     src     (znamienkové)`
- d) `Imul`

74. Znamienko výsledku aritmetickej operácie je indikované stavovým príznakom procesora i8080:

- a) PF
- b) CF
- c) ZF
- d) **SF**

75. Inštrukcia pre prácu s reťazcami `cmps` v jazyku procesora Pentium:

- a) **Porovná bajty na adresách DS:ESI a ES:EDI, nastaví príznaky, aktualizuje hodnoty registrov**
- b) Porovná bajty na adresách CS:ESI a ES:EDI, nastaví príznaky, aktualizuje hodnoty registrov
- c) Porovná bajty na adresách DS:ESI a SS:EDI, nastaví príznaky, aktualizuje hodnoty registrov
- d) Porovná bajty na adresách DS:EBP a ES:EBX, nastaví príznaky, aktualizuje hodnoty registrov

#### **Porovnanie reťazcov (`cmps`)**

- porovná bajty (slová, dvoj-slová) na DS:ESI a ES:EDI a nastaví príznaky (ako `cmp`)
- aktualizuje hodnoty ESI, EDI (podľa hodnoty DF a veľkosti operandov)

76. Direktíva `GLOBAL` prekladača `NASM`:

- a) Informuje assembler, že návestie nie je definované v aktuálnom module
- b) **Sprístupní návestie iným modulom programu**

#### **Direktíva `GLOBAL`**

- sprístupní návestie iným modulom programu (mená procedúr, premenných, ...)

77. Aký najvyšší počet segmentov môže byť súčasne aktívnych vo viac segmentovom režime (multisegment model) procesora Pentium?

- a) 8
- b) 6 • viac-segmentový (multisegment model) – súčasne aktívnych max. 6 segmentov
- c) 4
- d) 10

78. Ktoré z uvedených dvojíc inštrukcii jazyka procesora Pentium majú rovnakú sémantiku?

- a) ja, jnbe
- b) ja, jnb
- c) jbe, jna
- d) jb, jna

| mnemonika | je/jz          | jne/jnz                | ja/jnbe                      | jae/jnb                      | jb/jnae                      | jbe/jna                      |
|-----------|----------------|------------------------|------------------------------|------------------------------|------------------------------|------------------------------|
| význam    | equal/<br>zero | not equal/<br>not zero | above/<br>not below or equal | above or equal/<br>not below | below/<br>not above or equal | below or equal/<br>not above |

79. Syntax zápisu inštrukcii používaná prekladačom gcc pre in-line (vložený) assembler využíva pre uvedenie mena registra prefix:

- a) \* **AT&T syntax**
- b) =
- c) % • mená registrov – prefix % (napr. %eax)
- d) \$

80. Sémantika inštrukcie **xchg dst, src** je vyjadrená výrazom:

- a) Dst<-src • inštrukcia XCHG (exchange)
- b) Ani jeden z uvedených ○ výmena operandov, syntax
- c) Dst<->src
- d) Src<-dst xchg dst, src (dst ↔ src)

81. Inštrukcia **xor** realizujúca operáciu logickej nonekvivalencie sa využíva na:

- a) Nastavenie bitov (na hodnotu 1) **inštrukcia xor**
- b) Preklopenie hodnoty bitov (na opačnú)
- c) Izoláciu bitov • podpora zložených log. výrazov HL jazykov (neskôr)
- d) Nulovanie bitov • preklopenie (toggle) hodnoty bitu/bitov
- inicializácia registrov (0)

82. Po realizácii dvojice inštrukcii

**Mov AL, 10101110b**

**Or AL, 11110000b**

Bude register AL obsahovať hodnotu:

- a) EFH
- b) 00H
- c) FEH
- d) A0H



83. Nasledujúci fragment kódu predstavuje volanie služby systému:

**Mov AH,0**

**Int 16H**      Základné služby systému BIOS pre prácu s klávesnicou:

- a) Windows      • dostupné pomocou int 16H
- b) BIOS
- c) Linux
- d) MS DOS

84. Sémantika inštrukcie **loopne** v jazyku procesora Pentium je vyjadrená:

- a)  $ECX = ECX - 1$ , IF ( $ECX \neq 0$  AND  $ZF = 0$ ) skok na cieľ
- b)  $ECX = ECX - 1$ , IF ( $ECX \neq 0$  AND  $ZF = 1$ ) skok na cieľ
- c)  $ECX = ECX - 1$ , IF ( $ECX = 0$ ) skok na cieľ
- d)  $ECX = ECX - 1$ , IF ( $ECX \neq 0$ ) skok na cieľ

| loopne/loopnz                                                    |
|------------------------------------------------------------------|
| loop while not equal/<br>loop while not zero                     |
| $ECX = ECX - 1$<br>IF ( $ECX \neq 0$ AND $ZF = 0$ ) skok na cieľ |

85. Medzi stavové príznaky registra EFLAGS procesora Pentium patria:

- a) DF
- b) **OF**
- c) IF
- d) **CF**

86. Ako budú nastavené príznaky **Cy** a **Z** po vykonaní fragmentu kódu v jazyku i8080?

**Mvi A,0FH**

0000 1111

**Mvi B,0F1H**

**Add B**

- a)  $Cy = 0$ ,  $Z = 1$
- b)  $Cy = 1$ ,  $Z = 1$
- c)  $Cy = 0$ ,  $Z = 0$
- d)  **$Cy = 1$ ,  $Z = 0$**

S      1 - ak záporný výsledok  
Z      1 - nula  
P      1 - párny (s-ito rovnosť)  
Cy      1 - prenos nastal

87. Inštrukcie rotácii rcl, rcr pracujú:

- a) **S využitím CF v procese rotácie**
- b) Bez využitia CF v procese rotácie

88. Zásobník v systémoch s procesorom Pentium pri vkladaní údajov (push) narastá smerom:

- a) **K nižším adresám**      • zásobník narastá smerom k nižším adresám
- b) V závislosti od hodnoty bitu B segmentového deskriptora
- c) K vyšším adresám
- d) V závislosti od hodnoty bitu G segmentového deskriptora

89. Aký bude obsah FPU zásobníka po vykonaní inštrukcie fstp ST1 nad zásobníkom s hodnotami 5.0, 2.0, 3.0, 8.0 (vrchol zásobníka predpokladáme vľavo)?

- a) 5.0, 3.0, 8.0
- b) 5.0, 5.0, 3.0, 8.0
- c) 2.0, 2.0, 3.0, 8.0
- d) 7.0, 3.0, 8.0

`fstp ST1`



odstráni ST1, ST0 ponechá na vrchole

90. Normálna forma čísla +1101.101E11010 v zobrazení s pohyblivou rádovou čiarkou je:

- a) +1101.101E11101
- b) +1.101101E11100
- c) +1.101101E11101
- d) +1101.101E11010

91) Po realizácii ktorých z uvedených inštrukcií jazyka procesora Pentium bude platiť, že CF=1, ak obsah registra EAX je 15?

Označte jednu alebo viac odpovedí

a) `bt EAX, 1`

b) `bt EAX, 5`

c) `bt EAX, 7`

d) `bt EAX, 3`

```
mov EAX, 7
bt EAX, 1 ; CF = 1
bt EAX, 3 ; CF = 0
```

92) Inštrukcie logických operácií and, or, xor, not pracujú

Označte jednu alebo viac odpovedí

a) Bez operandov

b) S dvoma operandmi

c) S jedným operandom

d) S tromi operandmi

93) Ktoré z uvedených inštrukcií jazyka x86 majú priradené správne výsledky, ak AL=10101110 a BL=11110000b?

Označte jednu alebo viac odpovedí

a) `xor AL, BL` (AL=01010111b)

b) `xor AL, BL` (AL=10101110b)

c) `or AL, BL` (AL=11111110b)

d) `and AL, BL` (AL=10100000b) ????

94) Ktoré z uvedených príznakov budú nastavené (na hodnotu 1) po realizácii uvedeného fragmentu kódu v jazyku i8080?

```
Mvi A, 0FH
Mvi B, 0F1H
add B
```

Označte jednu alebo viac odpovedí

- ☒ CY  
b) ☐ S  
c) ☐ Z  
☒ P

95) Identifikácia registra na vrchole zásobníka (TOS) jednotky FPU procesorov Pentium sa nachádza v registri

Označte jednu alebo viac odpovedí

- a) FPU Status Register  
b) FPU Control Register  
c) Register ST0 ?????  
d) Tag Register

96) Voliteľný operand n inštrukcie ret predstavuje:

Označte jednu alebo viac odpovedí

- a) počet parametrov procedúry  
b) počet slov pre uvoľnenie zo zásobníka  
c) počet bajtov pre uvoľnenie zo zásobníka  
d) návratovú hodnotu

o parameter ret (počet bajtov pre uvoľnenie zo zásobníka, voliteľné)

```
ret n
```

97) Ktorá z uvedených inštrukcií jazyka i8080 realizuje operáciu vyjadrenú prenosom  $A \leftarrow A + R + Cy$ ?

Označte jednu alebo viac odpovedí

- a) SBB R  
b) ADC R  
c) ADD R  
d) SUB R

|       |          |   |   |                  |            |                           |
|-------|----------|---|---|------------------|------------|---------------------------|
| ADC R | 10001SSS | - | - | add reg. w.carry | S Z AcP Cy | $A \leftarrow A + R + Cy$ |
|-------|----------|---|---|------------------|------------|---------------------------|

98) Medzi podmienené skoky realizované podľa hodnoty jedného príznaku v jazyku procesora Pentium patria:

Označte jednu alebo viac odpovedí

- a) jge
- b) jnc
- c) je
- d) jns
- e) jo
- f) ja

99) Násobenie číslíc inštrukciou `mul`, je v jazyku procesora Pentium možné priamo realizovať na operandoch veľkosti

Označte jednu alebo viac odpovedí

- a) 32 bitov
- b) 16 bitov
- c) 64 bitov
- d) 8 bitov

`mul src (src - 8, 16, 32-bit GPR, pamäť)`

110) Inštrukcia ***or*** realizujúca operáciu logického súčtu sa využíva na:

Označte jednu alebo viac odpovedí

- a) Nastavenie bitov (na hodnotu 1)
- b) Preklopenie hodnoty bitov (na opačnú)
- c) Izoláciu bitov
- d) Nulovanie bitov

#### inštrukcia ***or***

- podpora zložených log. výrazov a bitová operácia log. súčtu HL jazykov (neskôr)
- nastavenie bitov (ak bit masky = 0 → výstup: kópia druhého vstupného bitu, ak bit masky = 1 → výstup: 1)

111) Ako bude nastavený príznak SF po realizácii uvedeného fragmentu kódu?

**`mov AL,72H`**  
**`add AL, 0EH`**

- a) OF = 1
  - b) OF = 0
- `mov AL, 72H`  
`add AL, 0EH` (114 + 14 = 128, OF = 1)

112) Do zásobníkového rámca (stack frame) sú údaje spojené s procedúrou vkladané v tomto poradí:

Označte jednu alebo viac odpovedí

- a) Starý EBP, parameter, návratová adresa, lokálne premenné
- b) návratová adresa, parametre, Starý EBP, lokálne premenné
- c) lokálne premenné, parametre, návratová adresa, Starý EBP
- d) parametre, návratová adresa, Starý EBP, lokálne premenné

- informácia v zásobníku (parametre, návratová adresa, starý EBP, príp. lokálne premenné) – zásobníkový rámec (stack frame)

113) Po realizácii nasledujúcej skupiny inštrukcií v jazyku procesora Pentium bude obsah registra AL:

Mov AL, 15

Mov CL, 4

Ror AL, CL

Označte jednu alebo viac odpovedí

- a) AL = 0FFH
- b) AL = 00H
- c) AL = 0FH
- d) AL = 0F0H

114) Adresu nasledujúcej inštrukcie procesora i8080 obsahuje register:

- a) Pc
- b) SP
  - SP (adresa poslednej vlozenej položky)
- c) RI
  - PC (adresa nasledujúcej inštrukcie)
- d) HL

115) Ktoré z uvedených dvojíc inštrukcií jazyka procesora Pentium majú rovnakú sémantiku?

- a) jnp, jnpe
- b) jno, jns
- c) jnp, jpo
- d) jnz, jne

116) Inštrukcia and realizujúca operáciu logického súčinu sa využíva na:

- a) izoláciu bitov
- b) nastavenie bitov (na hodnotu 1)
- c) preklopenie bitov (na opačnú)
- d) nulovanie bitov

**inštrukcia and**

- podpora zložených log. výrazov a bitová operácia log. súčinu HL jazykov (neskôr)
- nulovanie bitov
- izolácia bitov

**Otázka 1**

Ešte

nezodpovedané

Max. hodnotenie

1,00

Označiť otázku

Ako bude nastavený príznak PF po realizácii uvedeného fragmentu kódu?

```
mov AL,0FH
add AL,0F0H
```

Označte jednu alebo viac odpovedí:

- ☒ a. PF = 0
- ☐ b. PF = 1

**Otázka 1**

Ešte

nezodpovedané

Max. hodnotenie

1,00

Označiť otázku

Aký stav je signalizovaný stavovým príznakom CF registra EFLAGS procesora Pentium?

Označte jednu alebo viac odpovedí:

- ☒ a. výsledok aritmetickej operácie na bezznamienkových číslach prekročil rozsah cieľa
- ☐ b. výsledkom poslednej aritmetickej operácie bola 0
- ☐ c. výsledkom poslednej aritmetickej operácie bola záporná hodnota
- ☐ d. výsledok aritmetickej operácie na číslach so znamienkom prekročil rozsah cieľa

**Otázka 3**

Ešte

nezodpovedané

Max. hodnotenie

1,00

Označiť otázku

Inštrukciu logických posunov shr v jazyku procesora Pentium je možné využiť na:

Označte jednu alebo viac odpovedí:

- ☒ a. delenie čísel bez znamienka
- ☐ b. delenie čísel so znamienkom
- ☐ c. násobenie čísel bez znamienka
- ☐ d. násobenie čísel so znamienkom

**Otázka 3**

Ešte

nezodpovedané

Max. hodnotenie

1,00

Označiť otázku

Význam inštrukcie enter XY, 0 je vyjadrený:

Označte jednu alebo viac odpovedí:

- ☐ a. pop EBP; mov ESP, EBP; sub ESP, XY
- ☒ b. push EBP; mov EBP, ESP; sub ESP, XY
- ☐ c. mov ESP, EBP; pop EBP; sub ESP, XY
- ☐ d. mov EBP, ESP; push EBP; sub ESP, XY

**Otázka 3**

Ešte

nezodpovedané

Max. hodnotenie

1,00

Označiť otázku

Inštrukcia pre prácu s reťazcami cmpsb v jazyku procesora Pentium:

Označte jednu alebo viac odpovedí:

- ☐ a. porovná bajty na adresách DS:EBP a ES:EBX, nastaví príznaky, aktualizuje hodnoty registrov
- ☐ b. porovná bajty na adresách DS:ESI a SS:EDI, nastaví príznaky, aktualizuje hodnoty registrov
- ☐ c. porovná bajty na adresách CS:ESI a ES:EDI, nastaví príznaky, aktualizuje hodnoty registrov
- ☒ d. porovná bajty na adresách DS:ESI a ES:EDI, nastaví príznaky, aktualizuje hodnoty registrov



### Otázka 1

Ešte

nezodpovedané

Max. hodnotenie

1,00

Označiť otázku

Ktoré z uvedených inštrukcií jazyka procesora Pentium realizujú znamienkové rozšírenie?

Označte jednu alebo viac odpovedí:

- ☒ a. `cwd`
- ☒ b. `cdq`
- ☐ c. `bsf`
- ☐ d. `btc`

### Otázka 2

Ešte

nezodpovedané

Max. hodnotenie

1,00

Označiť otázku

Sémantika inštrukcie iterácií `loop` v jazyku procesora Pentium je vyjadrená:

Označte jednu alebo viac odpovedí:

- ☐ a. `ECX = ECX - 1, IF (ECX ≠ 0) skok na cieľ`
- ☐ b. `ECX = ECX - 1, IF (ECX ≠ 0 AND ZF = 0) skok na cieľ`
- ☒ c. `ECX = ECX - 1, IF (ECX ≠ 0 AND ZF = 1) skok na cieľ`
- ☐ d. `ECX = ECX - 1, IF (ECX = 0) skok na cieľ`

### Otázka 1

Ešte

nezodpovedané

Max. hodnotenie

1,00

Označiť otázku

Ako bude nastavený príznač ZF po realizácii uvedeného fragmentu kódu?

```
mov AL, 0FH
add AL, 0F1H
```

Označte jednu alebo viac odpovedí:

- ☒ a. ZF = 1
- ☒ b. ZF = 0

### Otázka 3

Ešte

nezodpovedané

Max. hodnotenie

1,00

Označiť otázku

Ktoré z uvedených mnemonik sú inštrukciami jazyka procesora Pentium?

Označte jednu alebo viac odpovedí:

- ☒ a. `btr`
- ☒ b. `btc`
- ☒ c. `bsr`
- ☒ d. `bts`
- ☒ e. `bsf`

### Otázka 1

Ešte

nezodpovedané

Max. hodnotenie

1,00

Označiť otázku

Ktoré z uvedených príznačov (FLAGS) ovplyvňuje inštrukcia `inc` jazyka procesora Pentium?

Označte jednu alebo viac odpovedí:

- ☒ a. ZF
- ☒ b. OF
- ☒ c. PF
- ☒ d. SF
- ☒ e. CF

**Otázka 2**

Ešte

nezodpovedané

Max. hodnotenie

1,00

Označiť otázku

Pri realizácii inštrukcie tzv.vzdialeného (far) skoku `jmp` v jazyku procesora Pentium sú modifikované registre:

Označte jednu alebo viac odpovedí:

- ☒ a. SS
- ☐ b. ESP
- ☐ c. CS
- ☐ d. EIP

**Otázka 2**

Ešte

nezodpovedané

Max. hodnotenie

1,00

Označiť otázku

Hodnoty priradené ktorými z uvedených direktív možno ďalej v texte programu meniť?

Označte jednu alebo viac odpovedí:

- ☒ a. `%define`
- ☒ b. `%assign`
- ☐ c. `EQU`



Vymenuj registrové páry jazyka i8080  
abecedne podľa prvého písmena  
vzor: AA,PA,...,XA , bez medzier oddelené čiarkou.

Answer:

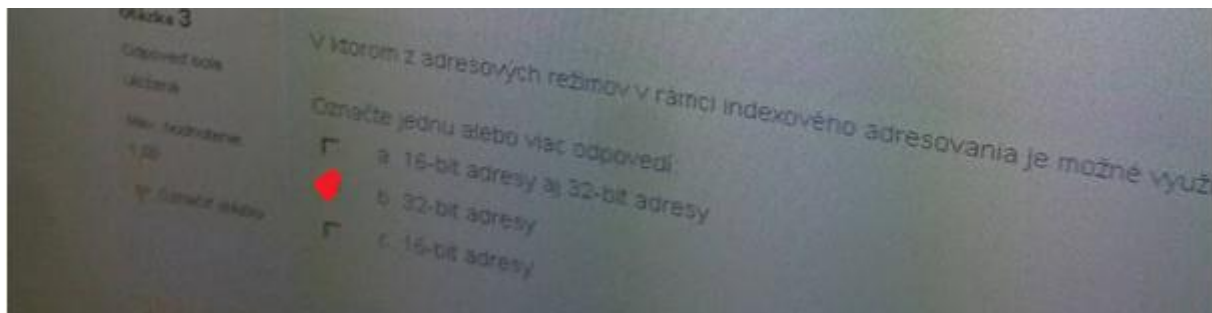
Syntax zápisu inštrukcií používaná prekladačom NASM pre in-line assembler je

- ☒ INTEL
- ☐ IBN
- ☐ AT&T

Organizácia pamäti v realnom režime procesora x86 predpokladá veľkosť segmentu

Označte jednu alebo viac odpovedí:

- ☐ a. 64MB
- ☒ b. 64KB
- ☐ c. 4GB
- ☐ d. 1MB



## Indexové adresovanie

- výpočet efektívnej adresy

$(\text{Index} * \text{scale}) + \text{disp}$  ; posunutie so znamienkom

- prístup k prvkom poľa
  - začiatok poľa (disp), prvok v poli (index register)
  - veľkosť prvkov (scale – 2, 4, 8 – iba 32-bit režim)

Ktorou z uvedených instrukcií procesora Pentium je možné zakázať obsluhu maskovateľných prerušení?

sti, cli

### Výskyt prerušení (prípád bez zmeny privilégii)

- EFLAGS → zásobník
- IF = 0 a TF = 0 (zakázanie ďalších prerušení; možné opätovné povolenie v ISR (sti, cli), pokiaľ nie je dôvod ich zakazovať)
- CS, EIP → zásobník
- CS ← 16b segment selector (interrupt gate)
- EIP ← 32b offset (interrupt gate)

**Otázka 2**  
Ešte nezodpovedané  
Max. hodnotenie: 1,00  
Označiť otázkou

Pri realizácii instrukcie shl. AL, 1 je najvyšší bit vystupujúci z registra AL zachytený v.

Označte jednu alebo viac odpovedí:

☐ a. OF

☒ b. CF

☐ c. PF

☐ d. SF

- stavové príznaky
  - AF – nedefinovaný
  - ZF, PF – podľa výsledku operácie, CF – posledný bit vysunutý z operandu
  - OF – nedefinovaný pre viac-bitové posuny
    - posuny o 1 bit – OF = 1, pri zmene bitu znamienka, ináč OF = 0

**Otázka 4**  
Ešte nezodpovedané  
Max. hodnotenie: 1,00  
Označiť otázkou

Ako indexový register pri pamäťových nepriamych režimoch adresovania procesorov Pentium možno v 16-bit režime použiť:

Označte jednu alebo viac odpovedí:

☒ a. DI

☐ b. BP

☐ c. BX

☒ d. SI