**rBajty**

Koľko bajtov spolu definujú nasledujúce direktívy?

*DB ´Chyba!$´*

*DW 5 dup (0)*

*DD 1.0*

**Odpoveď: 21**

Koľko bajtov spolu definujú nasledujúce direktívy?

*DB ´Chyba!´ , 0*

*DW 4 dup (0)*

*DD 10.0, 11.0, 12.0*

**Odpoveď: 27**

Koľko čísiel so znamienkom možno zobraziť v jednom bajte? (Uveďte v desiatkovej sústave.)

**Odpoveď: 256**

Aké najmenšie celé číslo so znamienkom možno zobraziť v jedno, bajte? (Uveďte

v desiatkovej sústave)

**Odpoveď: -128**

Aké najväčšie celé číslo so znamienkom možno zobraziť v jednom bajte? (Uveďte v desiatkovej

sústave)

**Odpoveď: 127**

Aký je najmenší počet bajtov, v ktorých môže byť uložené číslo v pohyblivej rádovej čiarke

podľa

normy IEEE?

**Odpoveď: 4**

**Adresy**

Koľko bajtov leží medzi adresami 4567:0100 a 4576:0010 (vrátane prvej a poslednej adresy)?

**Odpoveď: 1h**

Koľko bajtov leží medzi adresami 3010:0100 a 3120:0010 (vrátane prvej a poslednej

adresy)?

• A. 20h

**• B. 1011h**

• C. 21h

• D. F1h

• E. 1010h

Koľko bajtov leží medzi adresami 7010:0100 a 7100:0010 vrátane?

**E11h**

E1h

E10h

F1h

Ktorá z nasledujúcich adries v tvare segment: offset reprezentuje najvyššiu fyzickú adresu?

• A.74D7:0100

• B. 74D6:0200

**• C. 74D5:0300**

• D. 74D5:0100

Ktorá z nasledujúcich adries v tvare segment: offset reprezentuje najvyššiu fyzickú adresu?

• A. EEE5:0100

**• B. EED5:0300**

• C. EED6:0200

• D. EED7:0100

Ktorá z nasledujúcich adries v tvare segment: offset reprezentuje najvyššiu fyzickú adresu?

• A. ABC5:0200

**• B. ABC4:0300**

• D. ABD4:0100

• C. ABC6:0100

Bázová adresa segmentu je 4567h. Aký offset (v šestnáskovej sústave) v tomto segmente

zodpovedá

fyzickej adrese 45A00h?

**Odpoveď: 390**

Bázová adresa segmentu je 4567h. Aký offset (v šestnáskovej sústave) v tomto segmente

zodpovedá

fyzickej adrese 45A8Ah?

**Odpoveď: 41Ah**

Bázová adresa segmentu je 1267h. Aký offset (v šestnáskovej sústave) v tomto segmente

zodpovedá

fyzickej adrese 12A90h?

**Odpoveď: 420**

Bázová adresa segmentu je 1234h. Aký offset (v šestnáskovej sústave) v tomto segmente

zodpovedá

fyzickej adrese 12ABCh?

**Odpoveď: 077Ch //**12ABC – 1234 \* 10 = 077Ch

Bázová adresa segmentu je 6723h. Aký offset v tomto segmente zodpovedá fyzickej

adrese 67CECh?

**Odpoveď:**  0ABCh

**Offset**

Aký je najvyšší možný offset v segmente, ktorý má veľkosť 256 bajtov?

**Odpoveď: 255**

Aký bude obsah bajtu na offsete 200h v dátovom segmente po vykonaní nasledujúcich

inštrukcií?

*mov bx, 200h*

*mov ax, 32ABh*

*inc ax*

*mov [bx], ax*

**Odpoveď: Ach**

Aký bude obsah bajtu na ofsete 200h v dátovom segmente po vykonaní nasledujúcich

inštrukcií?

*mov bx, 200h*

*mov ax, 32ABh*

*dec ax*

*mov [bx], ax*

**Odpoveď: AAh**

Aký bude obsah bajtu na offsete 201h v dátovom segmente po vykonaní nasledujúcich

inštrukcií?

*mov bx, 200h*

*mov ax, 1234h*

*dec ax*

*mov [ax], ax*

**Odpoveď: 12h**

Aký bude obsah bajtu na ofsete 200h v dátovom segmente po vykonaní nasledujúcich

inštrukcií?

*mov bx, 200h*

*mov ax, 1234h*

*inc ax*

*mov [bx], ax*

**Odpoveď: 35h**

Aký bude obsah bajtu na ofsete 201h v dátovom segmente po vykonaní nasledujúcich inštrukcií

mov bx, 200h

mov ax, 3456h

dec ax

bov [bx], ax

**Odpoveď: 34h**

**Prevody**

Akému číslu v desiatkovej sústave zodpovedá číslo 10101010b, ak ho považujete za číslo so

znamienkom?

**Odpoveď: -86**

Akému číslu v desiatkovej sústave zodpovedá číslo 00101010b, ak ho považujete za číslo so

znamienkom?

**Odpoveď: 42**

Akému číslu v desiatkovej sústave zodpovedá číslo 11000011b, ak ho považujete za číslo so

znamienkom?

**Odpoveď: -61**

Ako je uložené číslo -70 v jednom bajte?

**Odpoveď: 10111010**

Číslo -52 je v jednom bajte uložené ako

10110101b

10110100b

**11001100b**

11001011b

Číslo -53 je v jednom bajte uložené ako

**11001011b**

10110110b

10110101b

11001010b

Číslo -54 je v jednom bajte uložené ako

10110111b

11001001b

10110110b

**11001010b**

**Priznakove bity**

Ktorý z uvedených príznakových bitov sa nastaví na 1 po vykonaní nasledujúcej postupnosti

inštrukcií?

*mov al, 0Fh*

*add al, 1*

• A. ZF

• B. CF

• C. OF

**• D. žiadny**

Ktorý z uvedených príznakových bitov sa nastaví na 1 po vykonaní nasledujúcej postupnosti

inštrukcií?

*mov al, 0FFh*

*cmp al, 0F0h*

• A. SF

• B. CF

**• D. žiadny**

• C. ZF

Po vykonaní nasledujúcej postupnosti inštrukcií

*mov al, 0FFh*

*add al, 1*

bude len v jednom z nasledujúcich príznakových bitov nula. V ktorom?

Vyberte jednu odpoveď:

• A. AF

• B. ZF

• C. CF

**• D. OF**

Ktorý z uvedených príznakových bitov sa nastaví na 1 po vykonaní nasledujúcej postupnosti

inštrukcií?

*mov al, 0FFh*

*cmp al, 1*

• A. ZF

• B. CF

**• C. SF**

• D. OF

Ktorý z uvedených príznakových bitov sa nastaví na 1 po vykonaní nasledujúcej postupnosti

inštrukcií?

*mov al, 11001001b*

*ad dal, 00110001b*

• A. ZF

• B. CF

**• C. SF**

• D. OF

Ktorý z uvedených príznakových bitov sa nastaví na 1 po vykonaní nasledujúcej postupnosti

inštrukcií?

*mov al, 1100000b*

*ad dal, 0100000b*

• A. CF

• B. ZF

**• C. OF**

• D. AF

Ktorý z uvedených príznakových bitov sa nastaví na 1 po vykonaní nasledujúcej postupnosti

inštrukcií?

*mov al, 11000000b*

*cmp al, 0*

• A. ZF

**• B. SF**

• C. OF

• D. CF

**Instrukcie A**

Je deklarovaná premenná

*R DB ´abcd´*

Čo vypíše nasledujúca postupnosť inštrukcií?

*mov di, 1*

*mov dl, [R+di]*

*inc di*

*dec dl*

*mov ah, 2*

*int 21h*

**Odpoveď: a**

Je deklarovaná premenná:

*A DB ´abc´*

Čo vypíše nasledujúca postupnosť inštrukcií?

*mov dl, A*

*inc A*

*mov ah, 2*

*int 21h*

**Odpoveď: a**

Je deklarovaná premenná

*R DB ´abcd´*

Čo vypíše nasledujúca postupnosť inštrukcií?

*mov di, 1*

*mov dl, [R+di]*

*dec di*

*inc dl*

*mov ah, 2*

*int 21h*

**Odpoveď: c**

Je deklarovaná premenná:

*R DB ´abde´*

Čo vypíše nasledujúca postupnosť inštrukcií?

*mov di, 2*

*mov dl, [R+di]*

*inc di*

*dec dl*

*mov ah, 2*

*int 21h*

**Odpoveď: c**

Je deklarovaná premenná:

*A DB 0*

o vypíše nasledujúca Č postupnosť inštrukcií?

*add A, ´a´*

*inc A*

*mov dl, A*

*mov ah, 2*

*int 21h*

**Odpoveď: b**

Je deklarovaná premenná:

*R DB ´abcd´*

Čo vypíše nasledujúca postupnosť inštrukcií?

*mov di, 1*

*inc di*

*mov dl, [R+di]*

*dec dl*

*mov ah, 2*

*int 21h*

**Odpoveď: b**

Je deklarovaná premenná:

*R DB ´abcd´*

Čo vypíše nasledujúca postupnosť inštrukcií?

*mov bx, offset R*

*mov dl, [bx+2]*

*dec bx*

*inc dl*

*mov ah, 2*

*int 21h*

**Odpoveď: d**

Je deklarovaná premenná:

*A DB ´abc´*

Čo vypíše nasledujúca postupnosť inštrukcií?

*mov dl, A*

*inc dl*

*mov ah, 2*

*int 21h*

**Odpoveď: b**

Je deklarovaná premenná:

*R DB ´abcd$´*

Čo vypíše nasledujúca postupnosť inštrukcií?

*mov dx, offset R+2*

*mov ah, 9*

*int 21h*

**Odpoveď: cd**

Je deklarovaná premenná:

*A DB ´0´*

Čo vypíše nasledujúca postupnosť inštrukcií?

*add A, 5*

*mov dl, A*

*mov ah, 2*

*int 21h*

**Odpoveď: 5**

Je deklarovaná premenná

*A DB ´123´*

Čo vypíše nasledujúca postupnosť inštrukci?

*mov dl, A*

*inc A*

*mov ah, 2*

*int 21h*

**Odpoveď: 1**

Aká bude deklarácia premennej R, ak nasledujúca postupnosť inštrukcií vypíše na obrazovku

text

´Ahoj´?

*mov dx, offset R*

*mov ah, 9*

*int 21h*

**Odpoveď: R DB 'Ahoj$**'

Je deklarovaná premenná

Retaz DB ‘Pondelok’

Čo vypíše nasledujúci program?

mov edx,offset Retaz

mov ecx,5

mov edi,5

Cyklus: mov al, [ecx+edi]

dec edi

loop Cyklus

call WriteChar

**Odpoveď: o**

**Instrukcie B**

Sú deklarované premenné

*DataA DB 56h*

*DataB DW -123*

*DataC DB ´Welcome´*

Ktorý z nasledujúcich príkazov je správny?

**• A. add DataC, al**

• B. add DataC, ax

• C. add DataB, al

• D. add ax, DataA

Sú deklarované premenné:

*DataA DB 0*

*DataB DW -123*

*DataC DB ´Welcome´*

Ktorý z nasledujúcich príkazov je správny?

• A. add DataB, al

• B. add DataC, ax

**• C. add DataB, offset DataA**

• D. add DataA, ax

Sú deklarované premenné

A DB 56h

B DW -123

C DB ‘$’

Ktorý z nasledujúcich príkazov je správny

Add C, ax

**Add C, al**

Add B, al

Add ax,A

**Instrukcie C**

V nasledujúcom programe chceme vynulovať 10 prvých prvkov Postupnosti. Doplňte

Chýbajúcu inštrukciu

*mov ax, 0*

*mov bx, offset Postupnost*

*mov cx, 10*

*mov di, 0*

*Cyklus: mov [bx+di], ax*

*....*

*loop Cyklus*

**Odpoveď: inc di**

Premenná postupnosť je deklarovaná takto

Postupnost DW 10 dup (0FFFFh)

V nasledujúcom programe chceme vynulovať všetky prvky Postupnosti. Doplňte chýbajúcu inštrukciu.

mov ax,0

mov ebx, ofset Postupnost

mov ecx,10

mov edi,0

Cyklus: mov [ebx+2\*edi],ax

......

Loop Cyklus

**Odpoveď: inc edi**

**Instrukcie D**

Ktorý príkaz je správny?

• A. mov dl, ax

• B. mov dl, ´abc´

• C. mov dl, al+1

**• D. mov dl, 0FFh**

Ktorý príkaz **nie je** správny

**• A. mov dx, ax+1**

• B. mov dx, [bx+1]

• C. mov dx, ´ax´+1

• D. mov dx, offset Text + 1

V procedúre chceme zachovať obsahy registrov AX a BX z volajúceho programu. Na začiatku

procedúry

sa vykonajú inštrukcie

*push ax*

*push bx*

V akom poradí budú na konci procedúry príkazy pop a ret?

a.)

*pop ax*

*pop bx*

*ret*

**b.)**

***pop bx***

***pop ax***

***ret***

c.)

*ret*

*pop ax*

*pop bx*

d.)

*ret*

*pop bx*

*pop ax*

*Nech procedúra Vypis má jeden parameter typu word odovzdávané cez zásobník. Čo je po vykonaní inštrukcie call na vrchole zásobníka?*

**Odpoveď: návratová adresa**

Nech procedúra Vypis má dva parametre typu word odovzdávané cez zásobník. Napíšte

správny tvar inštrukcie, ktorá vykoná návrat do volajúceho programu a zároveň odstráni tieto parametre zo zásobníka. **Odpoveď: ret 4**

Nech procedúra Vypis má jeden parameter typu word odovzdávaný cez zásobník. Čo je po vykonaní inštrukcie call na vrchole zásobníka?

**Odpoveď: ret 2**

Je deklarovaná premenná:

*R DB ´abcd$´*

Akú inštrukciu musíme napísať pred nasledujúce inštrukcie, ak chceme na obrazovku vypísať reťazec

abcd?

*mov ah, 9*

*int 21h*

**Odpoveď: mov dx, offset R**

**And**

Predpokladajte, že v registri AL je hodnota 0Fh, čo bude v registri AL po vykonaní inštrukcie AND al,10101010b

**Odpoveď** 0Ah

V uvedenom zápise doplňte taký binárny operand inštrukcie and aby po jej vykonaní mali všetky bity registra BH hodnotu 0

and BH,…b

**Odpoveď 00000000**

**Instrukcie Array**

Je deklarovaná premenná

*Array DB 1, 2, 3, 4, 5*

Čo bude v AH po vykonaní nasledujúcich inštrukcií?

*mov di, 3*

*mov ah, Array[di]*

**Odpoveď: 4**

Je deklarovaná premenná

*Array DB 1, 2, 3, 4, 5*

Čo bude v AH po vykonaní nasledujúcej inštrukcie?

*mov ah, [Array+2]*

**Odpoveď: 3**

Je deklarovaná premenná

Array DB 10 dup(?)

V registri BX je jej offset. Napíšte inštrukciu, ktorá vynuluje posledný prvok poľa Array.

**Odpoveď: mov [bx+9], 0**

Je deklarovaná premenná

Array DW 6 dup(?)

Čo musí byť v registri SI, aby príkaz mov [Array+SI], 0 aby vynuloval posledný prvok poľa Array?

**Odpoveď: 10?**

**Instrukcie + registre**

Sú deklarované premenné:

*.DATA*

*Data1 DW 2*

*Data2 DW 4*

*Data3 DW 6*

Čo bude v BX po vykonaní nasledujúcich inštrukcií?

*mov bx, offset Data2*

*add bx, 2*

**Odpoveď: 4**

Sú deklarované premenné:

*.DATA*

*Data1 DB 0*

*Data2 DB 2*

*Data3 DB 4*

Čo bude v BX po vykonaní nasledujúcich inštrukcií?

*mov al, 2*

*mov bx, offset Data1*

*mov [bx], al*

**Odpoveď: 0**

Sú deklarované premenné:

*.DATA*

*Data1 DW 2*

*Data2 DW 4*

*Data3 DW 6*

Čo bude v BX po vykonaní nasledujúcich inštrukcií?

*mov bx, offset Data1*

*add bx, 2*

**Odpoveď:** 2

Sú deklarované premenné:

*.DATA*

*Data1 DB 2*

*Data2 DB 4*

*Data3 DB 6*

Čo bude v BX po vykonaní nasledujúcich inštrukcií?

*mov bx, offset Data2*

*inc byte ptr [bx]*

**Odpoveď: 1**

Čo bude v registri AL po vykonaní nasledujúcich inštrukcií? (Napíšte v desiatkovej sústave.)

*mov al, 10101010b*

*neg al*

**Odpoveď: 86**

Čo bude v registri AL po vykonaní nasledujúcich inštrukcií? (Napíšte v desiatkovej sústave.)

*mov al, 11001010b*

*neg al*

**Odpoveď: 54**

Čo bude v registri AL po vykonaní nasledujúcich inštrukcií? (Napíšte v desiatkovej sústave.)

*mov al, 01010101b*

*neg al*

**Odpoveď: -85**

Aké číslo (v desiatkovej sústave) bude v registri AX po vykonaní nasledujúcej postupnosti

inštrukcií?

*mov ah, 2*

*mov bh, 3*

*add ah, bh*

*add ah, bh*

*mov al, 2*

**Odpoveď: 2050**

Aké číslo (v desiatkovej sústave) bude v registri AX po vykonaní nasledujúcej postupnosti

inštrukcií?

*mov ah, 2*

*mov al, 2*

*mov bh, 3*

*add ah, bh*

**Odpoveď: 1282**

**Pamatove miesta**

Predpokladajte, že pamäťové miesta 200h a 201h obsahujú čísla 12h a 34h. (Uvedené adresy

chápte ako offset v dátovom segmente.) Čo bude v registri AH po vykonaní nasledujúcich inštrukcií?

*mov bx, 200h*

*mov di, 1*

*mov ax, 10Ah*

*add ah, [bx+di]*

**Odpoveď: 35h**

Predpokladajte, že pamäťové miesta 200h a 201h obsahujú čísla 12h a 34h. (Uvedené adresy

chápte ako offset v dátovom segmente.) Čo bude v registri AH po vykonaní nasledujúcich inštrukcií?

*mov bx, 200h*

*mov di, 0*

*mov ax, 10Ah*

*add ah, [bx+di]*

**Odpoveď: 13h**

Predpokladajte, že pamäťové miesta 200h a 201h obsahujú čísla 34h a 12h. (Uvedené adresy

chápte ako offset v dátovom segmente.) Čo bude v AH po vykonaní nasledujúcich inštrukcií?

*mov ax, 10Ah*

*add ax, [200h]*

**Odpoveď: 13h**

Predpokladajte, že pamäťové miesta 200h a 201h obsahuje čísla 12h a 34h. (Uvedené adresy

chápte ako offset v dátovom segmente.) Čo bude v registri AL po vykonaní nasledujúcich inštrukcií?

*mov bx, 200h*

*mov di, 0*

*mov ax, 10Ah*

*add ah, [bx+di]*

**Odpoveď: 0Ah**

Predpokladajte, že pamäťové miesta 200h a 201h obsahujú čísla 4Ah a 8Fh. (Uvedené

adresy chápte ako offset v dátovom segmente.) Aký bude obsah registra AH po vykonaní nasledujúcich inštrukcií?

*mov ax,10Ah*

*add ax,[200h]*

**Odpoveď:** 90h

Predpokladajte, že pamäťové miesta 200h a 201h obsahujú čísla 4Ah a 8Fh. (Uvedené

adresy chápte ako offset v dátovom segmente.) Aký bude obsah pamäťového miesta

200h po vykonaní nasledujúcich inštrukcií?

*mov ax,[200h]*

*mov bl,al*

*mov al,ah*

*mov ah,bl*

*mov [200h],ax*

**Odpoveď:** 8Fh

**Registre**

Pri každom registri uveďte koľko bitov obsahuje

CX **16**

DI **16**

BH **8**

DS **16**

DL  **8**

V inštrukcii

*mov al, [.]*

sa v hranatej zátvorke **nemôže** vyskytovať register

• A. SI

• B. BX

**• C. DX**

• D. DI

Je deklarovaná premenná

*Pocet DB 0*

V inštrukcii

*mov ax, [.]*

sa v hranatej zátvorke nemôže vyskytovať:

• A. 200

• B. DI

• C. SI

**• D. Pocet**

Pocet DB 0

V inštrukcii

*mov al, [.]*

sa v hranatej zátvorke nemôže vyskytovať:

• A. Pocet

• B. 200

**• C. DX**

• D. DI

Je deklarovaná premenná

*Pocet DB 0*

V inštrukcií

*mov al, [.]*

sa v hranatej zátvorke nemôže vyskytovať:

• A. 200

**• B. DX**

• C. Pocet

• D. Di

Inštrukcie

*mov ax, @data*

*mov ds, ax*

uložia do registra DS

• A. offset prvej inštrukcie v programe

• B. offset prvej premennej

• C. bázovú adresu kódového segmentu

**• D. bázovú adresu datového segmentu**

Procedúra pristupuje k parametrom v zásobníku pomocou nepriameho adresovania

s registrom:

• A. SP

• B. DX

• C. BX

**• D. BP**

Ktorý register nemôže byť ľavým operandom v inštrukcii mov..., eax

EDI

ECX

**EIP**

EBP

**Teoria**

Superskalárna architektúra znamená, že:

• A. ďalšia inštrukcia sa začne spracovávať až po dokončení predchádzajúcej

• B. inštrukcia sa súčasne spracováva vo viacerých funkčných blokoch

**• C. procesor má viac výkonných jednotiek**

• D. viac inštrukcií sa spracováva súčasne

Predpovedanie skokov sa robí pri inštrukcií

• A. volania procedúry

**• B. podmieneného skoku**

• C. návratu z procedúry

• D. nepodmieneného skoku

Dekódovať inštrukciu znamená:

• A. vypočítať jej adresu

• B. vypočítať adresy operandov

**• C. rozložiť inštrukciu na mikrooperácie**

• D. preložiť ju do strojového kódu

Čítač inštrukcií obsahuje:

1. **offset inštrukcie, ktorá sa bude vykonávať ako nasledujúca**
2. bázovú adresu kódového segmentu
3. offset operandu
4. poradové číslo inštrukcie v programe

Technológia SIMD (Single Instruction Multiple Data) znamená. že:

**A. rovnaká operácia sa vykonáva paralelne s viacerými dátovými položkami**

B. procesor má viac výkonných jednotiek

C. viac inštrukcií sa spracováva súčasne

D. vykonáva sa predpovedanie skokov, aby sa zabránilo

Lokálne premenné v procedúre sú uložené v ... segmente.

Doplňte chýbajúce slovo.

**Odpoveď: zásobnikovom**

Kam uloží inštrukcia call návratovú adresu (adresu, na ktorej bude program pokračovať po n n ávrate z procedúry)?

**Odpoveď: do zásobníka**

Prúdové spracovanie inštrukcií znamená, že:

• A. inštrukcia sa súčasne spracováva vo viacerých funkčných blokoch

• B. ďalšia inštrukcia sa začne spracovávať až po dokončení predchádzajúcej

• C. procesor má viac výkonných jednotiek

**• D. viac inštrukcií sa spracováva súčasne**

Predpovedanie skokov sa v procesore Pentium IV vykonáva

• A. po vykonaní inštrukcie skoku

**• B. po načítaní inštrukcie skoku z pamäti do procesora asi**

• C. po vyhodnotení podmienky v inštrukcii skoku

• D. po dekódovaní inštrukcie skoku

Register SS počas vykonávania programu obsahuje:

• A. offset vrcholu zásobníka

• B. offset inštrukcie, ktorá sa bude vykonávať akonasledujúca

**• C. bázovú adresu zásobníkového segmentu**

• D. offset operandu

Register CS počas vykonávania program obsahuje:

• A. poradové číslo inštrukcie v programe

• B. offset operandu

• C. offset inštrukcie, ktorá sa bude vykonávať ako nasledujúca

• **D. bázovú adresu kódového segmentu**

**Navestia, skoky, cykly**

Predpokladajte, že register AX a BX obsahujú čísla *bez znamienka*. Chceme skočiť na

návestie Koniec, ak číslo v AX je väčšie než číslo v BX. Napíšte správnu inštrukciu skoku, ktorá bude

nasledovať po príkaze

*cmp ax, bx*

**Odpoveď: ja Koniec**

Predpokladajte, že register AX a BX obsahujú čísla bez znamienka. Chceme skočiť na

návestie Koniec, ak číslo v AX je rovnaké ako číslo v BX. Napíšte správnu inštrukciu skoku, ktorá bude nasledovať po príkaze

*cmp ax, bx*

**Odpoveď: jz Koniec**

Predpokladajte, že register AX a BX obsahujú čísla bez znamienka. Chceme skočiť na

návestie Koniec, ak číslo v AX je menšie než číslo v BX. Napíšte správnu inštrukciu skoku, ktorá bude

nasledovať po príkaze

*cmp ax, bx*

**Odpoveď: jb Koniec**

Chceme skočiť na návestie Koniec, ak číslo v AX je väčšie ako -1. Napíšte vhodnú inštrukciu

skoku, ktorá nasleduje po inštrukcii

*cmp ax, -1*

**Odpoveď: jg Koniec**

Chceme skočiť na návestie Koniec, ak číslo v AX je väčšie ako -1. Napíšte vhodnú inštrukciu skoku.

*cmp ax, -1*

**Odpoveď: jnle Konice**

Predpokladajte, že v registri AX je 0. Vykoná sa skok v nasledujúcej sekvencii príkazov?

*cmp ax, -1*

*jnle Koniec*

**Odpoveď: Ano**

Predpokladajte, že v registri AX je 0. Vykoná sa skok v nasledujúcej sekvencii príkazov?

*cmp ax, -1*

*jle Koniec*

**Odpoveď: Nie**

Predpokladajte, že v registri AX je -1. Vykoná sa skok v nasledujúcej sekvencii príkazov?

*cmp ax, 0*

*jnle Koniec*

**Odpoveď: Nie**

Predpokladajte, že v registri AX je 0. Vykoná sa skok v nasledujúcej sekvencii príkazov?

*cmp ax, -1*

*jge Koniec*

**Odpoveď: Ano**

Predpokladajte, že v registri AX je -1. Vykoná sa skok v nasledujúcej sekvencii príkazov?

*cmp ax, -1*

*jnle Koniec*

**Odpoveď: Nie**

V nasledujúcom programe chceme 5-krát opakovať Cyklus. Doplňte chýbajúcu inštrukciu.

*mov ax, 10*

*mov bx, offset Postupnost*

*...*

*mov di, 0*

*Cyklus: mov [bx+di] , ax*

*inc di*

*inc di*

*loop Cyklus*

**Odpoveď: mov cx, 5**