1) Procesor i8086 nepřímo explicitně adresuje: **Jméno a st. skupina:** .....

a) pomocí kteréhokoliv šestnáctibitového registru
b) pouze pomocí indexových registrů SI a DI
c) pouze pomocí registrů BX, SI, DI a BP
d) pouze pomocí registrů BX, SI, DI a SP

2) V registru AX máme bezznaménkové kladné číslo 0 .. 65535. Celočíselné dělení AX dvěma provedeme

a) instrukcí RCR AX,1
b) instrukcí SHR AX,1
c) instrukcí SAR AX,1
d) žádná z uvedených instrukcí nedělí celočíselně bezznaménkové číslo dvěma

3) Nastavená paleta pro režim 320x200 (13h) dovoluje zobrazit

a) 256 různých barev z 256 možných
b) 64 různých barev z 256 možných
c) 256 různých barev z 262144 možných
d) 256 různých barev z 16777216 možných

4) Bezznaménkové kladné číslo v registru AX 0..65535 chceme vynásobit 0.75 a získat celočíselnou část výsledku. Tuto operaci

a) provedeme například instrukcemi MOV CX,0.75; MUL CX
b) provedeme například instrukcemi MOV CX,750; MUL CX; DIV 1000
c) provedeme například instrukcemi MOV CX,49152; MUL CX; MOV AX,DX
d) nelze provést bez použití numerického koprocesoru

5) Program

```
PUSH AX
MOV DI,FLAG
MOV AL,20h
MOV [CS:DI],AL
OUT 20h,AL
POP AX
IRET
```

pro obsluhu i8086 hardwarového přerušení IRQ 0 je chybný, protože

a) obsahuje neexistující instrukci
b) nenahrává hodnotu do registru CS, podle kterého pak adresuje
c) nenahrává si registr DS, který mohl ukazovat mimo segment našeho programu
d) nevrací hodnoty v registrech do původního stavu před přerušením

1) Procesor i8086 ma takto nastavené registry: **Jméno a st. skupina:** .....
DS=2000h, ES=0FFFFh, BX=11h. Pak instrukce MOV AL,[ES:BX] nahraje do AL bajt z fyzické adresy:

a) 020011h
b) 000001h
c) 100001h
d) 120011h

2) Použití instrukcí RET far nebo RET near na konci podprogramu:

a) je vždy dáno typem CALLu, jakým podprogram voláme
b) je libovolné, fungují vždy obě varianty
c) je libovolné, jde o dva různé názvy jediné instrukce
d) je nesmysl, podprogram se ukončuje pomocí MOV AX,4C00h INT 21h

3) V registru AL je nějaké bezznaménkové číslo větší než 200. Chceme toto číslo jedinou instrukcí zmenšit o 5 a nastavit CARRY. Pak použijeme

a) instrukci SUB AL,5
b) instrukci SBB AL,5
c) instrukci ADD AL,251
d) žádná z uvedených instrukcí nevyhovuje

4) V registru AX máme číslo v doplňkovém kódu -32768 .. 32767. Celočíselné dělení AX dvěma provedeme

a) instrukcí SHR AX,1
b) instrukcí SAR AX,1
c) instrukcí RCR AX,1
d) žádná z uvedených instrukcí nedělí celočíselně číslo v doplňkovém kódu dvěma

5) Program

```
PUSH AX
MOV AX,20h
OR [SI],AH
OUT 20h,AX
POP AX
IRET
```

pro obsluhu i8086 přerušení IRQ 0 je chybný, protože

a) adresuje registrem SI, který může být nastaven přerušeným programem jakkoliv
b) nesprávně odhlašuje přerušení řadiči
c) nenahrává si registr DS, který mohl ukazovat mimo náš program
d) obsahuje neexistující instrukci

1) Procesor i8086 nepřímo explicitně adresuje: **Jméno a st. skupina:** .....

a) pomocí kteréhokoliv šestnáctibitového registru
b) pouze pomocí indexových registrů SI a DI
c) pouze pomocí registrů BX, SI, DI a BP
d) pouze pomocí registrů BX, SI, DI a SP

2) V registru AX máme bezznaménkové kladné číslo 0 .. 65535. Celočíselné dělení AX dvěma provedeme

a) instrukcí RCR AX,1
b) instrukcí SHR AX,1
c) instrukcí SAR AX,1
d) žádná z uvedených instrukcí nedělí celočíselně bezznaménkové číslo dvěma

3) Nastavená paleta pro režim 320x200 (13h) dovoluje zobrazit

a) 256 různých barev z 256 možných
b) 64 různých barev z 256 možných
c) 256 různých barev z 262144 možných
d) 256 různých barev z 16777216 možných

4) Bezznaménkové kladné číslo v registru AX 0..65535 chceme vynásobit 0.75 a získat celočíselnou část výsledku. Tuto operaci

a) provedeme například instrukcemi MOV CX,0.75; MUL CX
b) provedeme například instrukcemi MOV CX,750; MUL CX; DIV 1000
c) provedeme například instrukcemi MOV CX,49152; MUL CX; MOV AX,DX
d) nelze provést bez použití numerického koprocesoru

5) Program

```
PUSH AX
MOV DI,FLAG
MOV AL,20h
MOV [CS:DI],AL
OUT 20h,AL
POP AX
IRET
```

pro obsluhu i8086 hardwarového přerušení IRQ 0 je chybný, protože

a) obsahuje neexistující instrukci
b) nenahrává hodnotu do registru CS, podle kterého pak adresuje
c) nenahrává si registr DS, který mohl ukazovat mimo segment našeho programu
d) nevrací hodnoty v registrech do původního stavu před přerušením

2) Procesor i8086 ma takto nastavené registry: **Jméno a st. skupina:** .....
DS=2000h, ES=0FFFFh, BX=11h. Pak instrukce MOV AL,[ES:BX] nahraje do AL bajt z fyzické adresy:

a) 020011h
b) 000001h
c) 100001h
d) 120011h

2) Použití instrukcí RET far nebo RET near na konci podprogramu:

a) je vždy dáno typem CALLu, jakým podprogram voláme
b) je libovolné, fungují vždy obě varianty
c) je libovolné, jde o dva různé názvy jediné instrukce
d) je nesmysl, podprogram se ukončuje pomocí MOV AX,4C00h INT 21h

3) V registru AL je nějaké bezznaménkové číslo větší než 200. Chceme toto číslo jedinou instrukcí zmenšit o 5 a nastavit CARRY. Pak použijeme

a) instrukci SUB AL,5
b) instrukci SBB AL,5
c) instrukci ADD AL,251
d) žádná z uvedených instrukcí nevyhovuje

4) V registru AX máme číslo v doplňkovém kódu -32768 .. 32767. Celočíselné dělení AX dvěma provedeme

a) instrukcí SHR AX,1
b) instrukcí SAR AX,1
c) instrukcí RCR AX,1
d) žádná z uvedených instrukcí nedělí celočíselně číslo v doplňkovém kódu dvěma

5) Program

```
PUSH AX
MOV AX,20h
OR [SI],AH
OUT 20h,AX
POP AX
IRET
```

pro obsluhu i8086 přerušení IRQ 0 je chybný, protože

a) adresuje registrem SI, který může být nastaven přerušeným programem jakkoliv
b) nesprávně odhlašuje přerušení řadiči
c) nenahrává si registr DS, který mohl ukazovat mimo náš program
d) obsahuje neexistující instrukci