**;Adunări și scăderi**

***;8.*** *(a+b-d)+(a-b-d)*

*;a-byte, b-word, d-qword – Interpretare fără semn*

;a+b-d

mov al,[a] ;al=a

mov ah,0 ;ax=a

add ax,[b] ;ax=ax+b=a+b

mov dx,0 ;dx:ax=a+b

push dx

push ax

pop eax

mov edx,0 ;edx:eax=a+b

mov ecx,dword[d]

mov ebx,dword[d+4] ;ebx:ecx=d

add ecx,eax

adc ebx,edx ;ebx:ecx=a+b-d

;a-b-d

mov al,[a] ;al=a

mov ah,0 ;ax=a

sub ax,[b] ;ax=ax-b=a-b

mov dx,0 ;dx:ax=a+b

push dx

push ax

pop eax

mov edx,0 ;edx:eax=a+b

sub eax,dword[d]

sbb edx,dword[d+4] ;edx:eax=a-b-d

;(a+b-d)+(a-b-d)

add ecx,eax

adc ebx,edx ;ecx:ebx=(a+b-d)+(a-b-d)

***;8.*** *(b+c+d)-(a+a), interpretare cu semn*

*;a-byte, b-word, c-double word, d-qword – Interpretare cu semn*

;b+c

mov ax,[b] ;ax=b

cwde ;eax=ax=b

add eax,c ;eax=b+c

cdq ;edx:eax=b+c

;b+c+d

add eax,dword[d]

adc edx,dword[d+4] ;edx:eax=b+c+d

mov ecx,eax

mov ebx,edx ;ebx:ecx=b+c+d

;a+a

mov al,[a] ;al=a

add al,[a] ;al=a+a

cbw ;ax=a+a

cdq ;edx:eax=a+a

;(b+c+d)-(a+a)

sub ecx,eax

sbb ebx,edx ;ebx:ecx=(b+c+d)-(a+a)

**; Înmulțiri și împărțiri**

;25.(a\*a+b+x)/(b+b)+c\*c, interpretare fără semn

;a-word, b-byte, c-dword, x-qword

;a\*a+x

mov ax,[a] ;ax=a

mul word[a] ;dx:ax=a\*a

push dx

push ax

pop eax ;eax=a\*a

mov edx,0 ;edx:eax=a\*a

mov ebx,dword[x]

mov ecx,dword[x+4] ;ecx:ebx=x

add eax,ebx

adc edx,ecx ;edx:eax=a\*a+x+CF

;b+b

mov bl,[b] ;bl=b

add bl,[b] ;bl=b+b

mov bh,0 ;bx=b+b

mov cx,0 ;cx:bx=b+b

push cx

push bx

pop ebx ;ebx=b+b

;(a\*a+b+x)/(b+b)

div ebx ;eax=(a\*a+x+CF)/(b+b) edx=(a\*a+x+CF)%(b+b)

mov ecx,eax ;ecx=(a\*a+x+CF)/(b+b)

mov ebx,0 ;ebx:ecx=(a\*a+x+CF)/(b+b)

;c\*c

mov eax,[c] ;eax=c

mul dword[c] ;edx:eax=c\*c

;(a\*a+b+x)/(b+b)+c\*c

add eax,ecx

adc edx,ebx ;edx:eax=(a\*a+x+CF)/(b+b)+c\*c

;25.(a\*a+b+x)/(b+b)+c\*c, interpretare cu semn

;a-word, b-byte, c-dword, x-qword

;b+b

mov al,[b] ;al=b

cbw ;ax=b

cwde ;eax=b

add eax,[b] ;eax=b+b

mov ecx,eax ;ecx=b+b

;a\*a+x

mov ax,[a] ;ax=a

imul word[a] ;dx:ax=a\*a

push dx

push ax

pop eax

cdq ;edx:eax=a\*a

add eax,dword[x]

adc edx,dword[x+4] ;edx:eax=a\*a+x+CF

;(a\*a+b+x)/(b+b)

idiv ecx ;eax=(a\*a+x+CF)/(b+b) edx=(a\*a+x+CF)%(b+b)

cdq ;edx:eax=(a\*a+x+CF)/(b+b)

mov ebx,edx

mov ecx,eax ;ebx:ecx=(a\*a+x+CF)/(b+b)

;c\*c

mov eax,[c] ;eax=c

imul dword[c] ;edx:eax=c\*c

;(a\*a+b+x)/(b+b)+c\*c

add eax,ecx

adc edx,ebx ;edx:eax=(a\*a+x+CF)/(b+b)+c\*c

**;** **Adunări și scăderi**

**;16.***c-a-(b+a)+c*

*;a-byte, b-word, d-qword – Interpretare fără semn*

;c-a

mov al,[a] ;ax=a

mov ah,0 ;ax=a

mov dx,0 ;dx:ax=a

push dx

push ax

pop eax ;eax=a

mov ecx,[c]

sub ecx,eax ;ecx=c-a

;b+a

mov ax,[b] ;ax=b

mov bl,[a] ;bl=a

mov bh,0 ;bx=a

add ax,bx ;ax=b+a

mov dx,0 ;dx:ax=b+a

push dx

push ax

pop eax ;eax=b+a

;c-a-(b+a)

sub ecx,eax ;ecx=c-a-(b+a)

;c-a-(b+a)+c

add ecx,[c] ;ecx=c-a-(b+a)+c

**;20***.(a+c)-b+a+(d-c)*

*;a-byte, b-word, d-qword – Interpretare fără semn*

;a+c

mov al,[a] ;al=a

mov ah,0 ;ax=a

mov dx,0 ;dx:ax=a

add ax,word[c]

add dx,word[c+2] ;dx:ax=a+c

;(a+c)-b

sub ax,[b] ;dx:ax=(a+c)-b

;(a+c)-b+a

mov bl,[a] ;bl=a

mov bh,0 ;bx=a

add ax,bx ;dx:ax=(a+c)-b+a

push dx

push ax

pop eax ;eax=(a+c)-b+a

mov edx,0 ;edx:eax=(a+c)-b+a

;d-c

mov ebx,dword[d]

mov ecx,dword[d+4] ;ecx:ebx=d

sub ebx,[c] ;ecx:ebx=d-c

;(a+c)-b+a+(d-c)

add eax,ebx

adc edx,ecx ;edx:eax=(a+c)-b+a+(d-c)+CF

**;17**.*(c+d-a)-(d-c)-b*

*;a-byte, b-word, c-double word, d-qword – Interpretare cu semn*

;c+d

mov eax,[c] ;eax=c

cdq ;edx:eax=c

add eax,dword[d]

adc edx,dword[d+4] ;edx:eax=c+d+CF

mov ebx,eax

mov ecx,edx ;ecx:ebx=c+d+CF

;c+d-a

mov al,[a] ;al=a

cbw ;ax=a

cwd ;edx:eax=a

sub ebx,eax

sbb ecx,edx ;ecx:ebx=c+d+CF-a-CF

;d-c

mov eax,dword[d]

mov edx,dword[d+4] ;edx:eax=d

sub eax,[c] ;edx:eax=d-c

;(c+d-a)-(d-c)

sub ebx,eax

sbb ecx,edx ;ecx:ebx=(c+d+CF-a-CF)-(d-c)-CF

;(c+d-a)-(d-c)-b

mov ax,[b] ;ax=b

cwde ;eax=b

sub ebx,eax ;ecx:ebx=(c+d+CF-a-CF)-(d-c)-CF-b

**;27.** *(d+d-c)-(c+c-a)+(c+a)*

*;a-byte, b-word, c-double word, d-qword – Interpretare cu semn*

;d+d

mov eax,dword[d]

mov edx,dword[d+4] ;edx:eax=d

mov ebx,dword[d]

mov ecx,dword[d+4] ;ebx:ecx=d

add ebx,eax

adc ecx,edx ;ecx:ebx=d+d

;d+d-c

sub ebx,[c] ;ecx:ebx=d+d-c

;c+c

mov ecx,[c] ;ecx=c

add ecx,[c] ;ecx=c+c

;c+c-a

mov al,[a] ;al=a

cbw ;ax=a

cwde ;eax=a

add ebx,eax ;ebx=c+c-a

mov eax,ebx ;eax=c+c-a

cdq ;edx:eax=c+c-a

;(d+d-c)-(c+c-a)

sub ebx,eax

sbb ecx,edx ;ecx:ebx=(d+d-c)-(c+c-a)-CF

;c+a

mov al,[a] ;al=a

cbw ;ax=a

cwde ;eax=a

mov edx,[c] ;edx=c

sub edx,eax ;edx=c+a

mov eax,edx ;eax=c+a

cdq ;edx:eax=c+a

;(d+d-c)-(c+c-a)+(c+a)

add ebx,eax

adc ecx,edx ;ecx:ebx=(d+d-c)-(c+c-a)-CF+(c+a)+CF

**;Înmulțiri și împărțiri**

***;8****.1/a+200\*b-c/(d+1)+x/a-e*

*;a,b,e-word c,d-byte x-qword - Interpretare fără semn*

;1/a

mov ax,1 ;ax=1

mov dx,0 ;dx:ax=1

idiv word[a] ;ax=1/a dx=1%a

mov bx,0

mov cx,ax ;bx:cx=1/a

;200\*b

mov ax,200 ;ax=200

imul word[b] ;dx:ax=200\*b

;1/a+200\*b

add ax,cx

adc dx,cx ;dx:ax=1/a+200\*b

push dx

push ax

pop ecx ;ecx=1/a+200\*b

;c/(d+1)

mov al,[c] ;al=c

mov ah,0 ;ax=c

add byte[d],1 ;d=d+1

idiv byte[d] ;al=c/d ah=c%d

mov ah,0 ;ax=c/d

mov dx,0 ;dx:ax=c/d

push dx

push ax

pop eax ;eax=c/d

;1/a+200\*b-c/(d+1)

sub ecx,eax ;ecx=1/a+200\*b-c/(d+1)

;x/a

mov al,[a] ;al=a

mov ah,0 ;ax=a

mov dx,0 ;dx:ax=a

push dx

push ax

pop ebx ;ebx=a

mov eax,dword[x]

mov edx,dword[x+4];edx:eax=x

idiv ebx ;eax=x/a edx=x%a

;1/a+200\*b-c/(d+1)+x/a

add ecx,eax ;ecx=1/a+200\*b-c/(d+1)+x/a

;1/a+200\*b-c/(d+1)+x/a-e

sub ecx,[e] ;ecx=1/a+200\*b-c/(d+1)+x/a-e

***;8****.1/a+200\*b-c/(d+1)+x/a-e*

*;a,b,e-word c,d-byte x-qword - Interpretare cu semn*

;1/a

mov ax,1 ;ax=1

cwd ;dx:ax=1

idiv word[a] ;ax=1/a dx=1%a

mov bx,ax ;bx=ax=1/a

;200\*b

mov ax,200 ;ax=200

imul word[b] ;ax=ax\*b=200\*b

;1/a+200\*b

add bx,ax ;bx=1/a+200\*b+CF

;c/(d+1)

mov al,[c] ;al=c

cbw ;ax=al=c

add byte[d],1 ;d=d+1

idiv byte[d] ;al=c/(d+1) ah=c%(d+1)

cbw ;ax=al=c/(d+1)

;1/a+200\*b-c/(d+1)

sub bx,ax ;bx=1/a+200\*b-c/(d+1)

mov ax,bx ;ax=1/a+200\*b-c/(d+1)

cwde ;eax=1/a+200\*b-c/(d+1)

mov ebx,eax ;ebx=1/a+200\*b-c/(d+1)

;x/a

mov ax,[a] ;ax=a

cwde ;eax=ax=a

mov ecx,eax ;ecx=a

mov eax,dword[x]

mov edx,dword[x+4] ;edx:eax=x

idiv ecx ;eax=x/a edx=x%a

;1/a+200\*b-c/(d+1)+x/a

add ebx,eax ;ebx=1/a+200\*b+c/b+x/a

;1/a+200\*b-c/(d+1)+x/a-e

sub ebx,[e] ;ebx=11/a+200\*b+c/b+x/a-e

**;30.** a\*b-(100-c)/(b\*b)+e+x, interpretare fără semn

;a-word, b,c,byte, e-dword, x-qword

;100-c

mov cl,100 ;cl=100

sub cl,[c] ;cl=100-c

mov ch,0 ;cx=100-c

mov dx,0 ;dx:cx=100-c

;b\*b

mov al,[b] ;al=b

mul [b] ;ax=b\*b

mov bx,ax ;bx=b\*b

;(100-c)/(b\*b)

mov ax,cx ;dx:ax=100-c

div bx ;ax=(100-c)/(b\*b) dx=(100-c)%(b\*b)

mov cx,ax ;cx=(100-c)/(b\*b)

mov bx,0 ;bx:cx=(100-c)/(b\*b)

;a\*b

mov al,[b] ;al=b

mov ah,0 ;ax=b

mul [a] ;dx:ax=b\*b

;a\*b-(100-c)/(b\*b)

sub ax,cx

sbb dx,bx ;dx:ax=a\*b-(100-c)/(b\*b)-CF

;a\*b-(100-c)/(b\*b)+e

add ax,word[e]

adc dx,word[e+2] ;dx:ax=a\*b-(100-c)/(b\*b)-CF+e+CF

push dx

push ax

pop eax ;eax=a\*b-(100-c)/(b\*b)-CF+e+CF

mov edx,0 ;edx:eax=a\*b-(100-c)/(b\*b)-CF+e+CF

;a\*b-(100-c)/(b\*b)+e+x

add eax,dword[x]

adc edx, dword[x+4] ;a\*b-(100-c)/(b\*b)-CF+e+CF+x+CF

**;30.** a\*b-(100-c)/(b\*b)+e+x, interpretare cu semn

;a-word, b,c,byte, e-dword, x-qword

;b\*b

mov al,[b] ;al=b

imul [b] ;ax=b\*b

mov bx,ax ;bx=b\*b

;100-c

mov al,100 ;al=100

sub al,[c] ;al=100-c

cbw ;ax=100-c

cwd ;dx:ax=100-c

;(100-c)/(b\*b)

idiv bx ;ax=(100-c)/(b\*b) dx=(100-c)%(b\*b)

cwd ;dx:ax=(100-c)/(b\*b)

mov cx,dx

mov bx,ax ;cx:bx=(100-c)/(b\*b)

;a\*b

mov al,[b] ;al=b

cbw ;ax=b

imul [a] ;dx:ax=b\*b

;a\*b-(100-c)/(b\*b)

sub ax,bx

sbb dx,cx ;dx:ax=a\*b-(100-c)/(b\*b)-CF

;a\*b-(100-c)/(b\*b)+e

add ax,word[e]

adc dx,word[e+2] ;dx:ax=a\*b-(100-c)/(b\*b)-CF+e+CF

push dx

push ax

pop eax ;eax=a\*b-(100-c)/(b\*b)-CF+e+CF

cdq ;edx:eax=a\*b-(100-c)/(b\*b)-CF+e+CF

;a\*b-(100-c)/(b\*b)+e+x

add eax,dword[x]

adc edx, dword[x+4] ;a\*b-(100-c)/(b\*b)-CF+e+CF+x+CF