

Задание 10

10.1 Описать макроопределение SUM X, которое записывает в регистр EAX сумму элементов массива X (массив описан с помощью конструкции dup). Элементы массива – байты числа со знаком. Регистры (кроме EAX) не портить. Считать, что сумма уместится в двойное слово. Наличие и правильность передачи параметра не проверять. Описать массив B из 50 элементов и выписать макрорасширение для вызова SUM B

NByte macro x

local L

xor eax, eax

push ecx

push ebx

mov ecx, length x

L:movsx ebx, x[ecx-1]

add eax, ebx

Loop L

pop ebx

pop ecx

endm

10.2 Написать справа макрос с заголовком SgnCF macro V

Для записи значения флага переноса CF в знаковый разряд переменной V (байт, слово, двойное слово). Остальные разряды V обнулить. Регистры не портить. Макрорасширение должно содержать не более трёх команд. Проверку наличия и правильности типа параметра не проводить.

SgnCF macro V

mov V, 0

adc V, 0

ror V, 1

endm

10.3 Написать справа макроопределения с заголовком ADD2 macro A, Z

Оно должно реализовать операцию $A := +$ количество бит со значением '1' в Z. Параметр A – 32 разрядный регистр, параметр Z имеет формат m32 или m8. В макрорасширении должно быть не более 4-х команд. Проверку наличия и правильности типов параметров не проводить. Параметр Z не портить. Можно без восстановления использовать любые регистры.

```
ADD2 macro A, Z
```

```
    local L
```

```
    push A
```

```
    mov ecx, 8*type Z
```

```
L:ror Z, 1
```

```
    adc dword ptr[esp], 0
```

```
    Loop L
```

```
    pop A
```

```
endm
```

10.4 Описать макрос CntB macro x, где x список <x1, x2..., xi>, xi – имена переменных. Макрос подсчитывает и печатает количество байтовых переменных в списке x. Регистры не использовать. Выписать макрорасширение для вызова CntB <x,a,b>, где x - переменная типа dword a, b – типа byte. Макрокоманду печати не раскрывать.

```
CntB macro x:VarArg
```

```
    local Num
```

```
    Num=0
```

```
    for i, <x>
```

```
        if type i eq 1
```

```
            Num=Num+1
```

```
        endif
```

```
    endm
```

```
    outwordln Num
```

```
endm
```

10.5 Описать справа макроопределение NByte macro X, для своего параметра-списка X = <x1, x2,..., xn> вычисляет и присваивает регистру EAX, число параметров из списка, имеющих форматы r8 или m8. Регистры не портить. Наличие и правильность параметра не проверять. Макрорасширение должно содержать минимальное количество команд.

.code

CntB macro x:VarArg

 xor eax, eax

 for i, <x>

 if type i eq 1

 inc eax

 endif

 endm

 outwordIn Num

endm

Start:

 CntB al,ebx

10.6 Описать макрос NULL X, N, T. Здесь x – имя массива из N слов или двойных слов, T – это First или Last, N – константа. Макрос записывает ноль в первый (при T = first) или в последний (при T = LAST) элемент массива x. Регистры не портить. Правильность и наличие параметров не проверять. Макрорасширение должно состоять из минимально возможного числа команд.

NULL macro X, N, T

local i

if type X eq 2

ifdif <T>, <last>

mov word ptr x, 0

else

mov word ptr x[N*2-2], 0

endif

else

ifdif <T>, <last>

mov dword ptr x, 0

else

mov dword ptr x[N*4-4], 0

endif

endif

endm

10.7 Описать справа макроопределение NByte **macro** Name,S, которое для своего параметра-списка $S=\langle x_1, x_2, \dots, x_n \rangle$ вычисляет число M параметров x_i из списка, которые имеют формат m8. Макрорасширение должно состоять из описания переменной с именем Name типа **db** при чётном № и **dw** при нечётном №, где № – Ваш номер в ведомости. Величина M должна быть начальным значением этой переменной. Регистры не портить. Наличие и правильность параметров не проверять.

NByte macro Name, S:VarArg

local K, M

M=0

for i, <S>

K=0

if type i EQ 1

for j, <al, ah, bl, bh, cl, ch, dl, dh>

ifidni <i>, <j>

K=1

exitm

endif

endm

if K EQ 0

M=M+1

endif

endif

endm

Name dw M

endm

