

Министерство образования и науки Российской Федерации  
Ярославский государственный университет им. П. Г. Демидова  
Кафедра компьютерной безопасности и математических  
методов обработки информации

**Н. Б. Федотов**

# **Практикум на ЭВМ. Ассемблер**

Методические указания

*Рекомендовано  
Научно-методическим советом университета  
для студентов, обучающихся по специальности  
Математика*

Ярославль 2011

УДК 002  
ББК 3973.2–018.1я73  
Ф33

*Рекомендовано  
Редакционно-издательским советом университета  
в качестве учебного издания. План 2010/2011 учебного года*

### **Рецензент**

кафедра компьютерной безопасности и математических методов обра-  
ботки информации

**Федотов, Н. Б. Практикум на ЭВМ. Ассемблер** : методи-  
Ф33 ческие указания / Н. Б. Федотов ; Яросл. гос. ун-т им. П. Г. Де-  
мидова. – Ярославль: ЯрГУ, 2011. – 68 с.

Данная работа содержит описание лабораторных работ для прак-  
тического освоения программирования на языке ассемблера. Для каж-  
дой лабораторной работы формулируется ее цель, приводится список  
заданий и для одного из заданий дается его полное решение в виде  
отлаженной программы и набора тестов, на котором она проверялась.  
Каждая программа содержит достаточно подробные комментарии,  
полезные для самостоятельного обучения, а каждый тест включает в  
себя его описание, исходные данные и ожидаемые результаты.

Предназначено для студентов, обучающихся по специальности  
010101.65 «Прикладная математика и информатика» (дисциплина  
«Практикум на ЭВМ», блок ЕН), очной формы обучения. .

УДК 002  
ББК 3973.2–018.1я73

© Ярославский государственный  
университет им. П. Г. Демидова, 2011

Прежде чем переходить к описанию лабораторных работ, сделаем замечания общего характера. Каждая работа должна содержать комментарии, описывающие содержание работы, смысл важнейших ее частей, а также отдельных команд, имеющих ключевое значение. Далее, крайне важно приучиться использовать мнемонические имена. И наконец, важнейшей частью работы является разработка достаточно полного набора тестов, который должен следовать определенной логике, быть упорядочен, как правило, по размерности исходных данных и содержать описание теста, исходные данные и ожидаемый результат. Полезно набор тестов разработать прежде, чем писать программу. Кроме того, программы необходимо тестировать каждый раз заново при внесении любых изменений в ее текст.

## Лабораторная работа № 1

Цель работы – познакомиться со структурой программы, технологией выполнения (см. образец программы и приложение 1), арифметическими командами (см. приложение 3), внутренним представлением целых чисел, работой в отладчике td.exe. В работе предполагается сравнение результатов выполнения арифметических команд в отладчике с результатами расчетов, выполненных вручную.

### Образец программы

```

title      lab1      первая программа на ассемблере
stk        segment para stack 'stack'
            db        64 dup('stack')
stk        ends
data       segment para public 'data'
x          dw        112
y          dw        122
a          dw        100 dup(0)
data       ends
code       segment para public 'code'
            assume    cs:code,ds:data
start:     mov        ax,data
            mov        ds,ax
            mov        ax,x                ;ax=x

```

	add	ax,y	;ax=x+y
	mov	a,ax	;a[0]=x+y
	mov	ax,x	;ax=x
	sub	ax,y	;ax=x-y
	mov	a+2,ax	;a[1]=x-y
	mov	ax,x	
	imul	y	;dx:ax=x*y
	mov	a+4,ax	;a[2]=ax
	mov	a+6,dx	;a[3]=dx
	mov	ax,x	
	cwd		;dx:ax=ax
	idiv	y	;ax=x/y dx=x mod y
	mov	a+8,ax	;a[4]=x/y
	mov	a+10,dx	;a[5]=x mod y
	mov	ax,x	
	inc	ax	;ax=x+1
	mov	a+12,ax	;a[6]=x+1
	mov	ax,y	
	dec	ax	;ax=y-1
	mov	a+14,ax	;a[7]=y-1
	mov	ax,x	
	sar	ax,3	;ax=x/8
	mov	a+16,ax	;a[8]=x/8
	mov	ax,y	
	sal	ax,4	;ax=y*16
	mov	a+18,ax	;a[9]=y*16
	mov	ah,4ch	;ah=4ch
	int	21h	
code	ends		
	end	start	

## Варианты заданий

Варианты заданий определяются значениями констант x,y:

- |              |              |              |              |
|--------------|--------------|--------------|--------------|
| 1. 135,1222  | 2. 89,4177   | 3. 92,1158   | 4. 101,412   |
| 5. 201,1276  | 6. 105,2242  | 7. 139,3157  | 8. 1322,178  |
| 9. 161,3183  | 10. 111,2136 | 11. 151,3223 | 12. 79,2177  |
| 13. 99,3157  | 14. 121,2173 | 15. 133,3226 | 16. 145,4252 |
| 17. 119,1217 | 18. 111,4148 | 19. 141,1193 | 20. 120,4255 |

## Лабораторная работа № 2

Цель работы – научиться программировать арифметические выражения с рационализацией по числу промежуточных вычислений, по выбору оптимальных команд. Использование внешних подпрограмм ввода-вывода и макросов.

### Образец программы

```
        title      lab2
stk      segment para stack'stack'
        db        64 dup('stack')
stk      ends
data     segment para public 'data'
input_x  db        10,13,'x= ','$'
input_y  db        10,13,'y= ','$'
output_z db        10,13,'z= ','$'
data     ends
code     segment para public 'code'
        extrn     rdint:far
        extrn     wrint:far
;--- макрос вывода строки
wrstr    macro str
        mov     dx,offset str
        mov     ah,9
        int     21h
        endm
        assume   cs:code,ds:data,ss:stk
start:   mov     ax,data
        mov     ds,ax
;--- распределение регистров
x        equ     bx
y        equ     bp
z        equ     si
;--- вывод запроса на число x
wrstr    input_x
        call    rdint
        mov     x,ax
;--- вывод запроса на число y
wrstr    input_y
        call    rdint
        mov     y,ax
;--- вычисление ((y-1) * x - (y-1)/x + (y-1)mod x)*16
;--- проверка на ОДЗ
        cmp     x,0
```