

Міністерство освіти і науки України  
Запорізький національний технічний університет

МЕТОДИЧНІ ВКАЗІВКИ  
до виконання розрахунково-графічного завдання  
з дисципліни "Архітектура комп'ютерів"  
для студентів спеціальностей  
7.05010201 "Комп'ютерні системи та мережі" та  
7.05010203 "Спеціалізовані комп'ютерні системи"  
всіх форм навчання

2015

Методичні вказівки до виконання розрахунково-графічного завдання з дисципліни "Архітектура комп'ютерів" для студентів спеціальностей 7.05010201 "Комп'ютерні системи та мережі" та 7.05010203 "Спеціалізовані комп'ютерні системи" всіх форм навчання / Укл. Скрупський С.Ю., Тіменко А.В., Сирота А.В. – Запоріжжя: ЗНТУ, 2015. – 14 с.

Укладачі: С.Ю. Скрупський, к.т.н., доцент  
А.В. Тіменко, асистент  
А.В. Сирота, ст. викладач

Рецензент: О.І. Вершина, к.т.н., доцент

Відповідальний за випуск: С.Ю. Скрупський, к.т.н., доцент

Затверджено:  
на засіданні кафедри  
"Комп'ютерні системи та  
мережі"  
Протокол № 6 від 02.02.2015

**ЗМІСТ**

|                                      |    |
|--------------------------------------|----|
| ВСТУП .....                          | 4  |
| 1.ІНДИВІДУАЛЬНІ ЗАВДАННЯ .....       | 4  |
| 2. ПРИКЛАД РОЗВ'ЯЗАННЯ ЗАВДАННЯ..... | 7  |
| ЛІТЕРАТУРА .....                     | 14 |

## ВСТУП

Розрахунково-графічне завдання (РГЗ) спрямоване на укріплення базових знань з архітектури комп'ютерів, зокрема, з програмування мікропроцесорної техніки мовою Асемблер. В даному РГЗ основним завданням є введення даних в процесор, обробка та виведення за допомогою інструментів мови Асемблер.

Варіанти індивідуальних завдань наведені у п.1. Декілька основних прикладів вирішення завдань вказано у п.2. Для вирішення завдань необхідно буде скористатися додатковою літературою, яку наведено у відповідному переліку. Особливу увагу приділити таблиці переривань `int 21h`.

РГЗ, по-перше, дозволяє підвищити кваліфікацію розробника (студента), по-друге – надає можливість оцінити рівень цієї кваліфікації. Виходячи з цього, виконання роботи (самостійність, своєчасність), зміст, якість і грамотність оформлення, дотримання графіка – все це фактори, які впливають на оцінку.

## 1. ІНДИВІДУАЛЬНІ ЗАВДАННЯ

Необхідно розв'язати завдання мовою Асемблер будь-якого мікропроцесора, наприклад, i8086, i8051 та ін. В пояснювальній записці навести блок-схему логічних кроків алгоритма вирішення завдання (або відповідну UML-діаграму діяльності) та повний лістинг програми з коментарями. Коректність виконання завдання підтвердити зображеннями (або таблицями) стану регістрів процесора після виконання кожної команди. Їх можна отримати за допомогою відладчика. Номер індивідуального завдання, наведений в табл. 1, відповідає останнім двом цифрам в заліковій книжці по модулю 25.

Таблиця 1 – Індивідуальні завдання

| № варіанту | Завдання                                                                                                                                                                                             |
|------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 01         | Написати програму, в якій міститься масив. Програма знаходить і виводить на екран максимальний елемент масиву                                                                                        |
| 02         | Ввести з клавіатури числа $a$ , $x$ , $b$ . Порахувати $y = a * x + b * x$ , і вивести на екран                                                                                                      |
| 03         | Ввести масив, знайти різницю між найбільшим і останнім елементами масиву                                                                                                                             |
| 04         | Необхідно ввести рядок, вивести його в іншому кольорі, на наступному рядку підрахувати кількість символів рядка                                                                                      |
| 05         | Ввести матрицю $C$ ( $M$ , $N$ ). Всі позитивні елементи матриці розділити на константу $E$ , негативні - помножити.<br>$M = 3$ , $N = 3$ , $E = 4$ .                                                |
| 06         | Необхідно ввести чотиризначні десяткові числа. Вивести на екран їх суму в десятковому записі.                                                                                                        |
| 07         | Дана матриця $4 \times 4$ , необхідно розділити всі позитивні елементи головної діагоналі на 2.                                                                                                      |
| 08         | Задано арифметичний вираз $(a + b) \wedge 3 - (a \wedge 3 + 3a \wedge 2b)$ , $a = 4$ , $b = 2$ . Вихідні дані описати як початкові поля в оперативній пам'яті. Результат помістити у виділені слова. |
| 09         | Необхідно ввести одновимірний масив. Створити новий масив, помістивши на місце непарних елементів елементи вихідного масиву, а парні елементи заповнити нулями.                                      |
| 10         | Розрахувати вираз $AX = \min \{ \max (a \wedge 2, b / c), \min (b \wedge 2, c / a) \}$ . Числа $a$ , $b$ , $c$ вводимо з клавіатури. Числа повинні бути позитивними.                                 |

|    |                                                                                                                                                                      |
|----|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 11 | Вводимо рядок з клавіатури. Необхідно вивести перший символ, весь рядок і останній символ. Виконати перевірку на наявність порожнього рядка                          |
| 12 | Ввести послідовність англійських букв з клавіатури. Вивести в 12 рядку екрана всі голосні літери (зеленим кольором), а в 20 рядку всі приголосні (червоним кольором) |
| 13 | Створити програму, яка буде підраховувати кількість невід'ємних чисел в масиві. Масив ввести з клавіатури                                                            |
| 14 | Написати програму, яка відсортовує введені числа за зростанням. Числа вводяться з клавіатури, результат вивести на екран                                             |
| 15 | Задане ціле беззнакове число $X < 65536$ . Визначити, чи є в запису числа цифра, рівна середньому арифметичному інших цифр. Отримане значення вивести на екран.      |
| 16 | Необхідно ввести одновимірний масив з клавіатури, вивести найбільше число з масиву і його порядковий номер на екран                                                  |
| 17 | Створити програму, яка знаходить саме коротке слово і виводить його на екран. Слова вводити на клавіатурі через прогалину                                            |
| 18 | Програма підраховує кількість букв «С» в самому довгому слові. Слова вводити з клавіатури                                                                            |
| 19 | Сформувані масив з 10 двобайтових чисел. Переписати у другій масив все позитивні числа                                                                               |
| 20 | Необхідно написати програму, яка буде виводити на екран $2x$ , $3x$ , і $4x$ -значні числа на екран                                                                  |
| 21 | Ввести в програму матрицю розміром $3 \times 4$ розміром в байт. Програма знаходить максимальний від'ємний елемент в матриці і виводить результат на екран           |
| 22 | Необхідно ввести матрицю розміром $M * N$ . Знайти суму значень, розташованих на головній діагоналі і вивести результат на екран                                     |
| 23 | Написати програму, в яку з клавіатури вводиться матриця $4 \times 4$ . Необхідно знайти суму чисел в непарних рядках і стовпцях матриці, результат вивести на екран  |
| 24 | Дана матриця чисел, потрібно знайти суму елементів в кожному рядку і відсортувати рядки в порядку зростання сум.                                                     |
| 25 | Ввести рядок з клавіатури. Вивести посимвольно, кожен символ в окремому рядку.                                                                                       |

## 2. ПРИКЛАД РОЗВ'ЯЗАННЯ ЗАВДАННЯ

2.1 Програма дозволяє вводити строку не більше 80 символів та вивести цю строку на консоль червоним кольором.

```
.model tiny
.code
.386
org 100h
start:
    mov ah,0Ah
    lea dx,String
    mov si,dx
    int 21h
    mov ah,2
    mov dl,0Ah
    int 21h
    inc si
    xor cx,cx
    mov cl,[si]
    inc si
m1:
    push cx
    lodsb
    mov ah,9
    mov bh,0
    mov cx,1
    mov bl,4
    int 10h
    mov ah,3
    int 10h
    inc dl
    mov ah,2
    int 10h
    pop cx
    loop m1
    mov ah,0
    int 16h
    mov ax,3
    int 10h
    ret
```

```
String db 80
end start
```

## 2.2. Програма виводить матрицю 5x5 на консоль.

```
.model small
.stack 100h
.data
mass    db 1,2,3,4,5
        db 1,2,3,4,5
        db 1,2,3,4,5
        db 1,2,3,4,5
        db 1,2,3,4,5
.code
start:
    mov ax,@data
    mov ds,ax

    xor si,si
    mov cx,5
    mov ah,2
@lp1:   push cx
        mov cx,5
@lp2:   mov dl,mass[si]
        add dl,30h
        int 21h
        inc si
        loop @lp2
        mov dl,0dh
        int 21h
        mov dl,0ah
        int 21h
        pop cx
        loop @lp1

    xor ax,ax
    int 16h

    mov ax,4c00h
    int 21h
end start
```

2.3 Програма дозволяє ввести в буфер рядок символів та вивести його на консоль.

```
data segment
    string db 100,100 dup ('$')
data ends

code segment
assume cs:code,ds:data
start:
    mov ax, @data
    mov ds, ax

    mov ah,0ah
    lea dx,string
    int 21h

    mov string+1,0ah

    mov ah,9
    lea dx,string+1
    int 21h

    mov ah, 10h
    int 16h

    mov ax, 4c00h
    int 21h
code ends
end start
```

2.4 Програма виведення цілого знакового або беззнакового числа.

```
OutInt proc

    ;; якщо число знакове, необхідно розкоментувати
наступні рядки
    ;; Перевіряємо число на знак.
    ; test ax, ax
    ; jns oi1
```

```

;; Якщо воно негативне, виведемо мінус і залишимо
його модуль.
; mov cx, ax
; mov ah, 02h
; mov dl, '-'
; int 21h
; mov ax, cx
; neg ax
;; Кількість цифр будемо тримати в CX.
; oi1:
    xor cx, cx
    mov bx, 10 ; основа сс. 10 для десятичної
oi2:
    xor dx, dx
    div bx
; Ділимо число на основу сс. У залишку виходить
;остання цифра.
; Відразу виводити її не можна, тому збережемо її
    push dx
    inc cx
; з залишком повторюємо те ж саме, відокремлюючи
;від нього чергову цифру праворуч, поки не залишиться
;нуль, що означає, що далі зліва лише нулі.
    test ax, ax
    jnz oi2
; приступимо до виводу.
    mov ah, 02h
oi3:
    pop dx
; Витягаємо чергову цифру, переводимо її в символ і
;виводимо розкоментувати якщо основа сс> 10, тобто
;для виведення потрібні літери
; cmp dl, 9
; jbe oi4
; add dl, 7
; oi4:
    add dl, '0'
    int 21h
; Повторимо рівно стільки разів, скільки цифр
;нараховували
    loop oi3

```

```
ret
```

```
OutInt endp
```

2.5 Процедури виведення числа в HEX-системі. Число для виведення повинно знаходитися в ax. Результат поміщується в es: di

```
byte2hex      proc      near
               push     cx
               mov      cx,2
@@L1:         rol      dl,4
               mov      ax,300fh
               and      al,dl
               aaa
               aad      11h
               stosb
               loop     @@L1
               pop      cx
               ret
byte2hex      endp

word2hex      proc      near
               push     cx
               mov      cx,2
@@L1:         rol      dx,8
               call     byte2hex
               loop     @@L1
               pop      cx
               ret
word2hex      endp

dword2hex     proc      near
               mov      cx,2
@@L1:         rol      edx,16
               call     word2hex
               loop     @@L1
               ret
dword2hex     endp
```

2.6 Процедури введення цілого позитивного або негативного числа. Введене число буде знаходитися в ах. Змінюючи число - основу системи числення, можна вводити числа в різних системах до десяткової.

```

InputInt proc
    mov ah,0ah
    xor di,di
    mov dx,offset buff ; адреса буферу
    int 21h ; приймаємо рядок
    mov dl,0ah
    mov ah,02
    int 21h ;
; обробка буферу
    mov si,offset buff+2 ; беремо адресу початку
рядку
    cmp byte ptr [si], "-" ; якщо перший символ
від'ємний
    jnz ii1
    mov di,1 ; ставимо флаг
    inc si ; та пропускаємо його
ii1:
    xor ax,ax
    mov bx,10 ; основа сс
ii2:
    mov cl,[si] ; беремо символ з буферу
    cmp cl,0dh ; перевіряємо чи не останній він
    jz endin

; якщо символ не останній, перевіряємо його на
вірність
    cmp cl,'0' ; якщо ввели невірний символ <0
    jl er
    cmp cl,'9' ; якщо ввели невірний символ >9
    ja er

    sub cl,'0' ; робим з символу число
    mul bx ; множимо на 10
    add ax,cx ; додаємо к останнім
    inc si ; покажчик на наступний символ
    jmp ii2 ; повторюємо

```

```

er:          ; якщо була помилка, то виводимо
повідомлення та виходимо
    mov dx, offset error
    mov ah,09
    int 21h
    int 20h
; всі символи буферу опрацьовані, число
знаходиться в ах
endin:
    cmp di,1 ; якщо встановлений флаг, то
    jnz ii3
    neg ax   ; робимо число від'ємним
ii3:
    ret

error db "incorrect number$"
buff   db 6,7 Dup(?)
InputInt endp

```

2.7 Процедура виведення числа в двійковій системі числення.

```

OutBin proc
    mov bx,ax
    mov cx,16
ob1:
    shl bx,1
    jc ob2

    mov dl,'0'
    jmp ob3

ob2:
    mov dl,'1'

ob3:
    mov ah,2
    int 21h
    loop ob1

OutBin endp

```

## ЛІТЕРАТУРА

1. Абель П. Язык ассемблера для IBM PC и программирование:/Пер. з англ. Ю. В.Сальникова. - М.: Высш. шк.,1992. – 477 с.
2. Варрон Д. Ассемблеры и загрузки. - М.: Мир, 1974 – 76 с.
3. Врэдли Д. Программирование на языке ассемблера для персональной ЭВМ фирмы IBM PC. - М.: Мир, 1988 - 448 с.
4. Дао Л. Программирование микропроцессора 8088. - М.: Мир, 1988 - 357 с.
8. Лямин Л. В. Макроассемблер MASM - М.: Радио и связь, 1994 -320 с.
9. Нортон Э., Сохуэ П. Д. Язык ассемблера для IBM PC - М.: Финансы и статистика, 1992 - 352 с.
10. Пильщиков В. И. Программирование на языке ассемблера IBM PC - М.: Диалог-МИФИ, 1994 - 288 с.
11. Скэнлон Л. Программирование ЭВМ IBM PC и XT. Программирование на языке ассемблера - М.: Радио и связь, 1991 – 336 с.
12. Turbo-Assembler 3.0.Quich Reference Guide. Borland International, Inc., 1991 – 386 p.
13. Turbo-Assembler 3.0.User's Guide.Borland International,Inc., 1991 – 650 p.
14. Шнайдер А. Язык ассемблера для персонального компьютера фирмы IBM - М.: Мир, 1988 - 406 с.
15. Ужен Лю, Гибсон Г. МП семейство 8086 /8088 - М: Радио и связь., 1987 – 304с.