

Національний технічний університет України
«Київський політехнічний інститут ім. І. Сікорського»

Факультет інформатики та обчислювальної техніки
Кафедра технічної кібернетики

*Звіти до комп'ютерних практикумів з модуля
«Системне програмування»*

Прийняв
доцент кафедри ТК
Лісовиченко О.І.

Виконав
Студент групи ІТ-61
Лесогорський К.С.

Комп'ютерний практикум №4

Тема: Програмування розгалужених алгоритмів.

Завдання:

1. Написати програму знаходження суми елементів масиву.
2. Написати програму пошуку максимального (або мінімального) елемента масиву.
3. Написати програму пошуку координат всіх входжень заданого елемента в двовірному масиві.
4. Написати програму сортування одномірного масиву цілих чисел загального вигляду.

Текст програми read_array.asm:

```
%define SIZE 20
%macro CHECK_INPUT_VALID_INTEGER 1
    cmp eax,%1
    jne invalid_integer_input
%endmacro

extern printf,scanf
global print_array, read_array,
invalid_integer_input, array, array_size,
read_2d_array, array_2d_size

section .data
in_mes_size db "Enter array size in range 1 to 20", 10,0
size_error_mes db "Invalid array size",10,0
parse_error db "Invalid character found", 10, 0
in_mes_digit db "Enter next digit:",10,0
in_format db "%d",0
new_line db 10,0
out_format db "%d ",0
in_mes_2d db "Enter two array sizes(rows,cols) in format 0 0",10,0
in_2d_format db "%d %d",0
section .bss
array resd SIZE
array_size resd 1
temp_var resd 1
array_2d_size resd 2
section .text

read_array:
    push in_mes_size
    call printf
    add esp,4
    push array_size
    push in_format
    call scanf
    add esp,8
    CHECK_INPUT_VALID_INTEGER 1

    parse_array_no_inp_mes:
    mov eax, [array_size]
    cmp eax, 0
    jle invalid_array_size
    cmp eax, SIZE
    jg invalid_array_size
    mov esi, [array_size]
    mov edi, array
    push in_mes_digit

    parse_next_int:
    call printf
    push temp_var
    push in_format
    call scanf
    add esp,8
    mov eax, [temp_var]
    mov [edi], eax
    add edi, 4
    dec esi
    jnz parse_next_int

    finalize:
    add esp,4
    ret

    print_array:
    mov esi, [array_size]
    mov edi, array
    sub esp,4
    push out_format
    write_next_int:
    mov eax, [edi]
    mov [esp+4], eax
    call printf
    add edi,4
    dec esi
    jnz write_next_int
    add esp,8
    push new_line
```

```
call printf
add esp,4
```

```
ret
```

```
invalid_array_size:
push size_error_mes
call printf
add esp,4
mov eax,1
int 80h
```

```
invalid_integer_input:
push parse_error
call printf
add esp,4
mov eax,1
int 80h
```

```
read_2d_array:
push in_mes_2d
call printf
add esp,4
push array_2d_size + 4
push array_2d_size
push in_2d_format
call scanf
add esp,12
CHECK_INPUT_VALID_INTEGER 2
mov eax, [array_2d_size + 4]
mov ebx, [array_2d_size]

imul ebx
mov [array_size], eax
jmp parse_array_no_inp_mes
```

Текст програми array_sum_minmax.asm:

```
%macro CHECK_OVERFLOW_SUM 0
jo inform_overflow
%endmacro
global main
extern printf,read_array,array,array_size
section .data
mes db "This program calculates sum of the array and
finds max and min element",10,0
overflow db "Overflow occured during sum
calculation",10,0
fin_format db "sum: %d, min: %d, max:%d",10,0
section .text

main:
push mes
call printf
add esp,4
call read_array
mov edi, array
mov esi, [array_size]

xor ebx,ebx
xor ecx,ecx
xor edx,edx

process_array:
mov eax,[edi]
add ebx,eax

CHECK_OVERFLOW_SUM
cmp eax, ecx
jge check_greater
mov ecx,eax
check_greater:
cmp eax, edx
jle check_index
mov edx, eax
check_index:
add edi, 4
dec esi
jnz process_array

push edx
push ecx
push ebx
push fin_format
call printf
add esp, 16
ret

inform_overflow:
push overflow
call printf
add esp,4
mov eax,1
int 80h
```

Текст програми array_2d_allhits.asm:

```
%macro CHECK_INPUT_VALID_INTEGER add esp,8
0%
cmp eax,1
jne invalid_integer_input          mov ebx,[array_2d_size + 4]
%endmacro

extern print_array, array, read_2d_array,
array_size, array_2d_size,
invalid_integer_input, printf, scanf
global main

section .data
mes db "This programm finds all matches of
number in 2d array",10,0
inp_mes db "Enter a number to look for:",10,0
match_format db "Hit! Row: %d, Col: %d",10,0
0
inp_format db "%d"
section .bss
temp_val resd 1
section .text
main:
push mes
call printf
add esp,4

call read_2d_array
push inp_mes
call printf
add esp,4

push temp_val
push inp_format
call scanf
CHECK_INPUT_VALID_INTEGER

                                mov edi, array
                                mov esi, [array_size]
                                process_element:
                                mov ecx, [edi]
                                cmp ecx, [temp_val]
                                jne check_index

                                mov eax,[array_size]
                                sub eax, esi

                                xor edx,edx
                                div ebx

                                push edx

                                push eax
                                push match_format

                                call printf
                                add esp,12

                                check_index:
                                add edi, 4
                                dec esi
                                jnz process_element

                                ret
```

Текст програми array_sort.asm:

```
global main
extern printf, read_array, print_array, array,
array_size

section .data
mes db "This program is bubble sort.",10,0
default_array_mes db 'You entered this
array:',10,0
sorted_array_mes db 'Sorted array:',10,0
section .text
main:
push mes
call printf

                                add esp,4
                                call read_array

                                push default_array_mes
                                call printf
                                add esp,4
                                call print_array

                                call sort_array

                                call print_array
```

ret

sort_array:

mov esi, [array_size]

mov edi, array

dec esi

external_loop:

mov eax,0

mov ebx,0; flag

internal_loop:

mov ecx, [edi + eax * 4]

mov edx, [edi + eax * 4 + 4]

cmp ecx,edx

jle skip_swap

swap:

mov [edi + eax * 4 + 4], ecx

mov [edi + eax * 4], edx

mov ebx,1

skip_swap:

inc eax

cmp eax,esi

jne internal_loop

out_internal:

cmp ebx,0

je finally

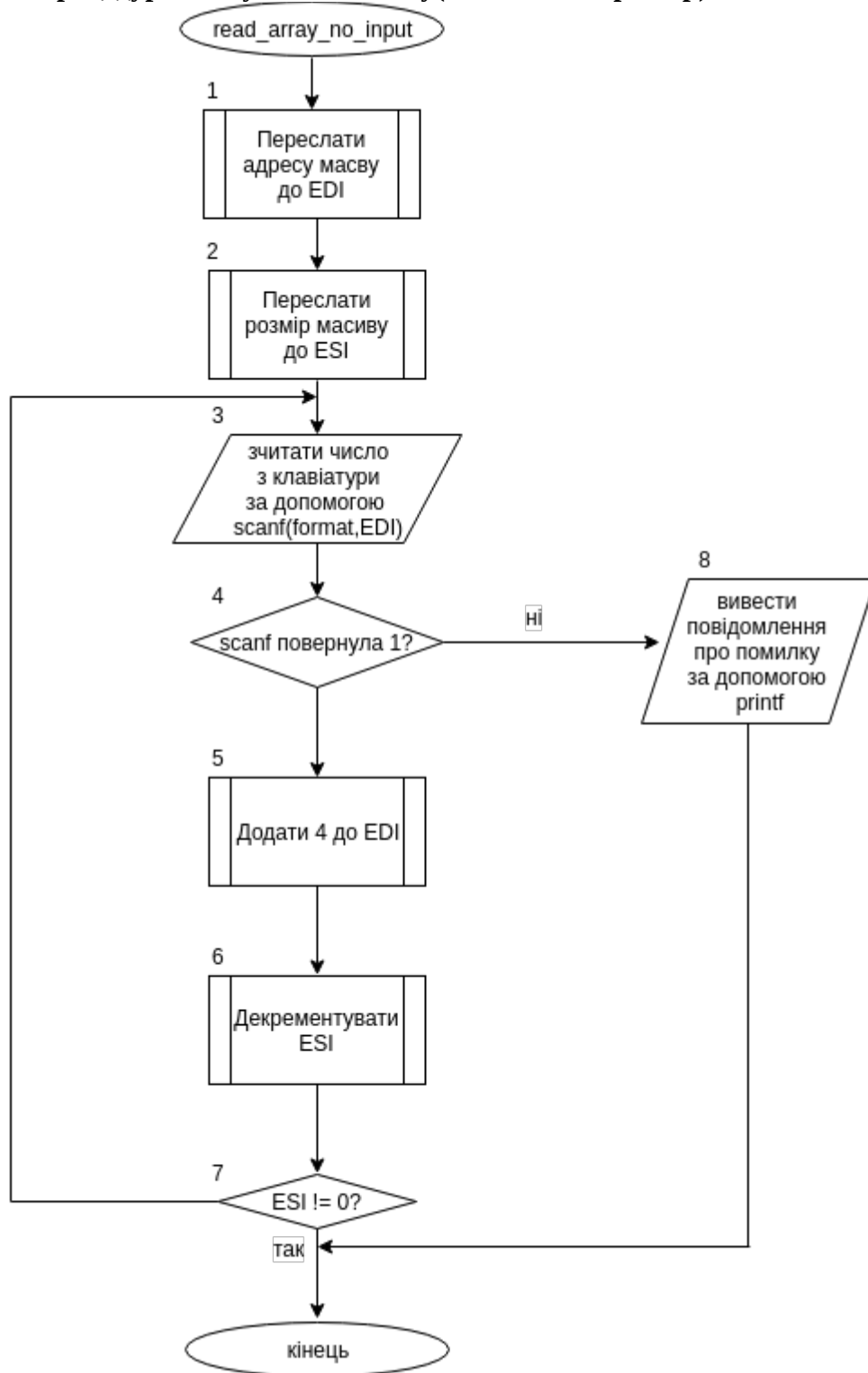
dec esi

jnz external_loop

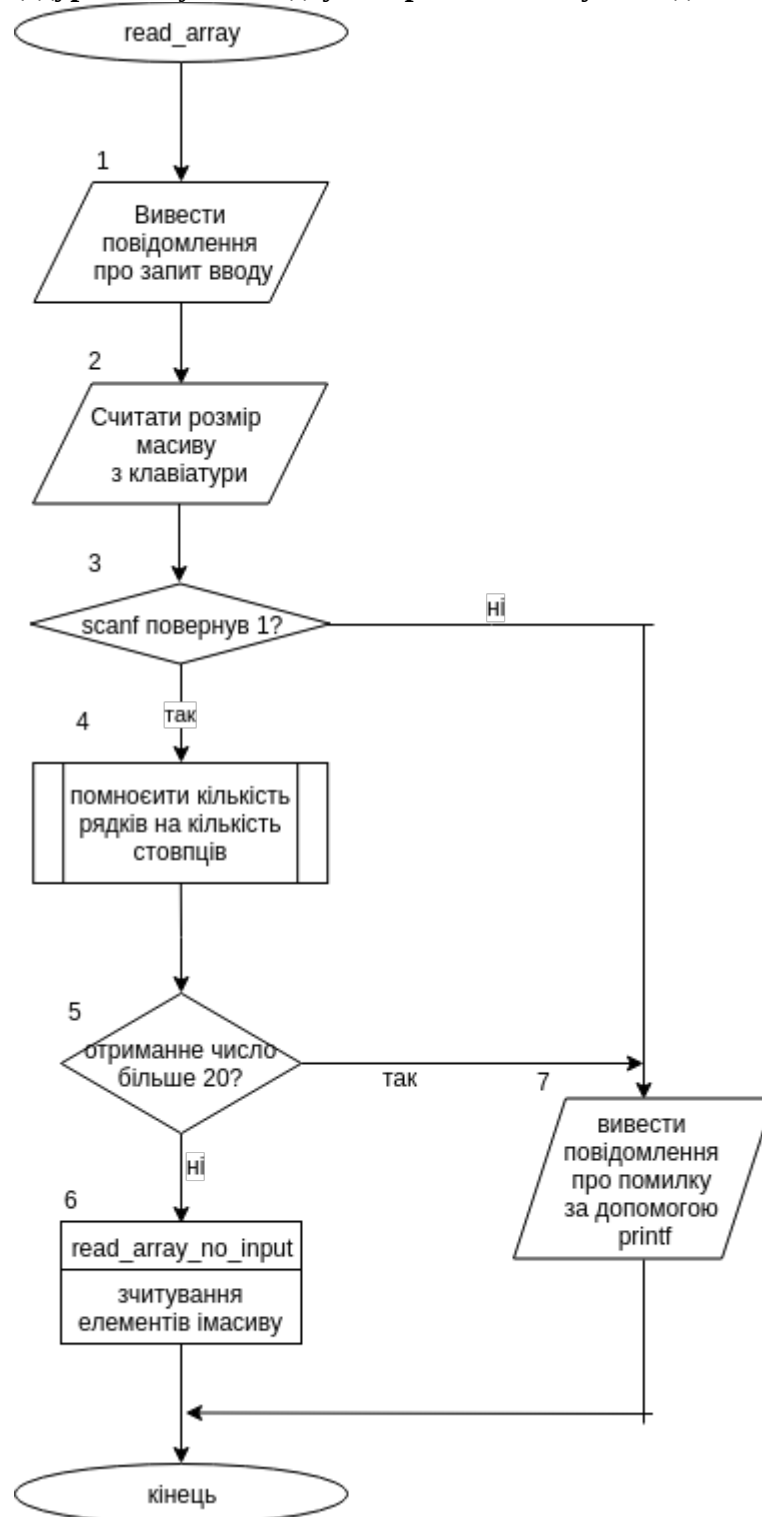
finally:

ret

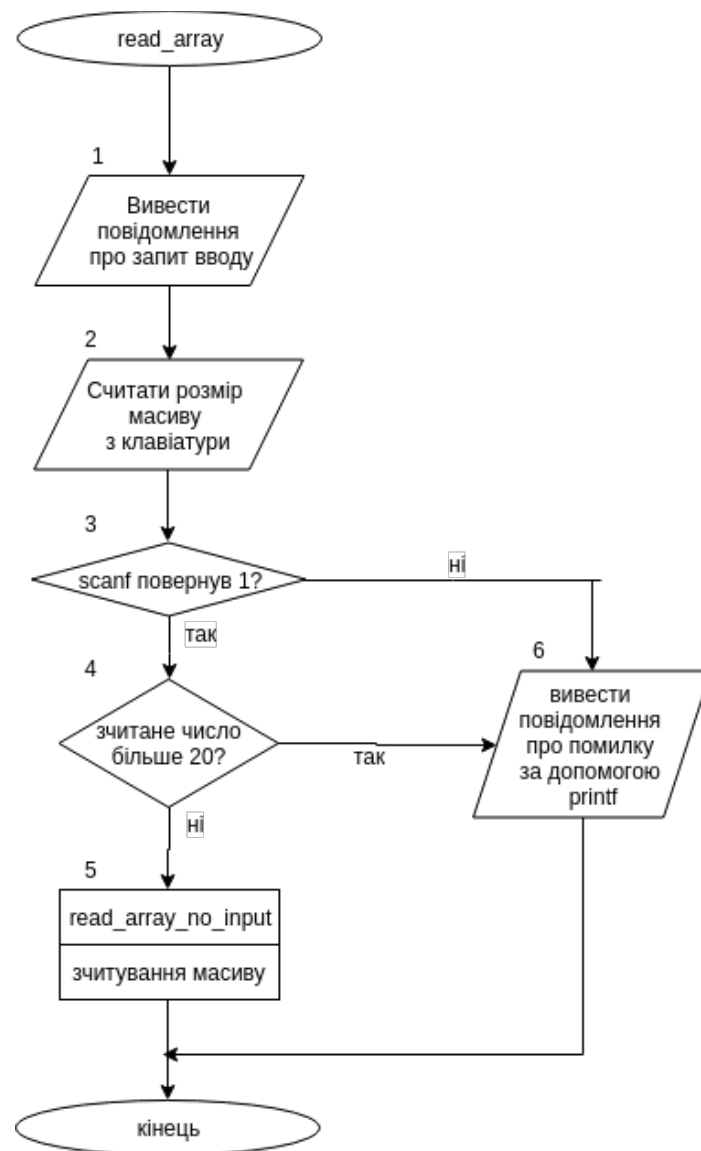
Блок-схема процедури зчитування масиву(знаючи його розмір):



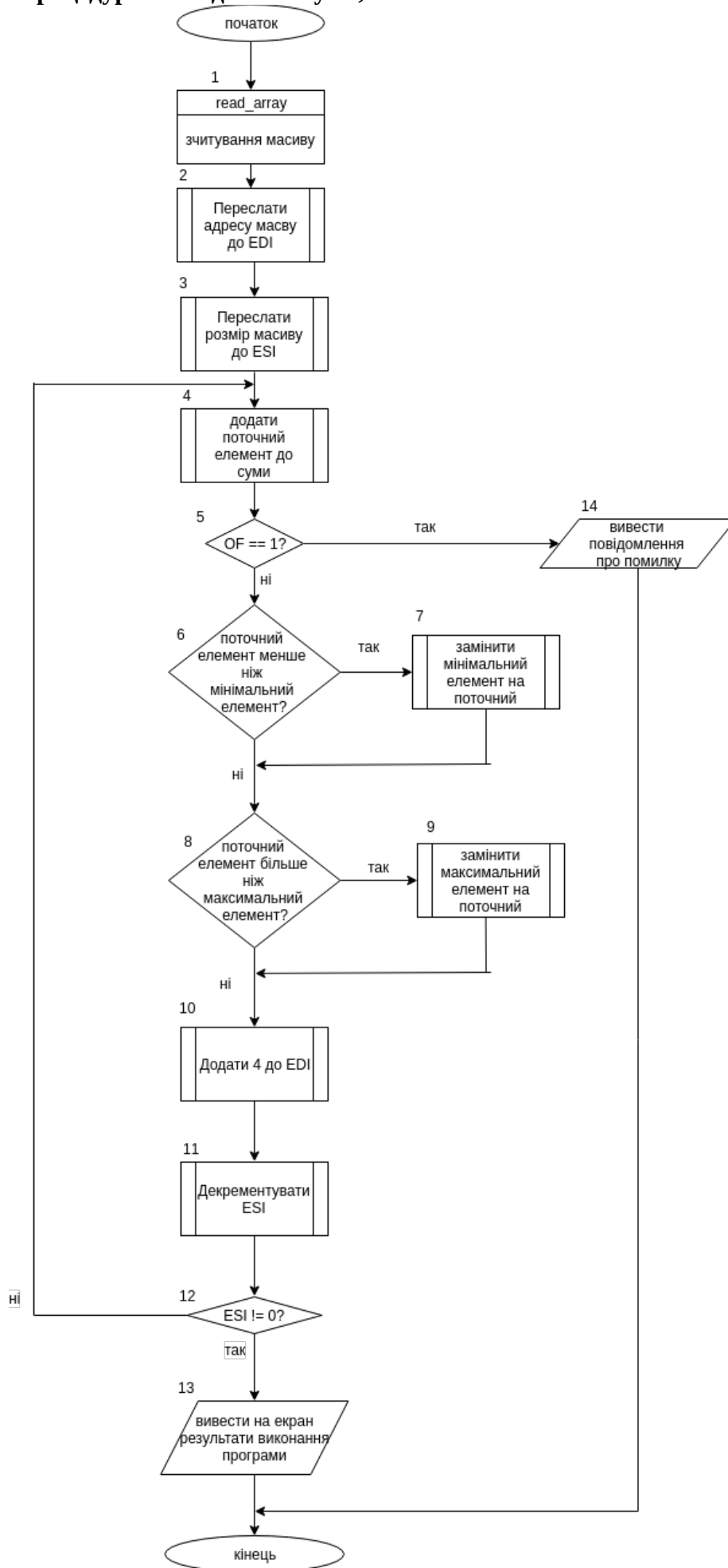
Блок-схема процедури зчитування двувимірного масиву з введенням його розміру:



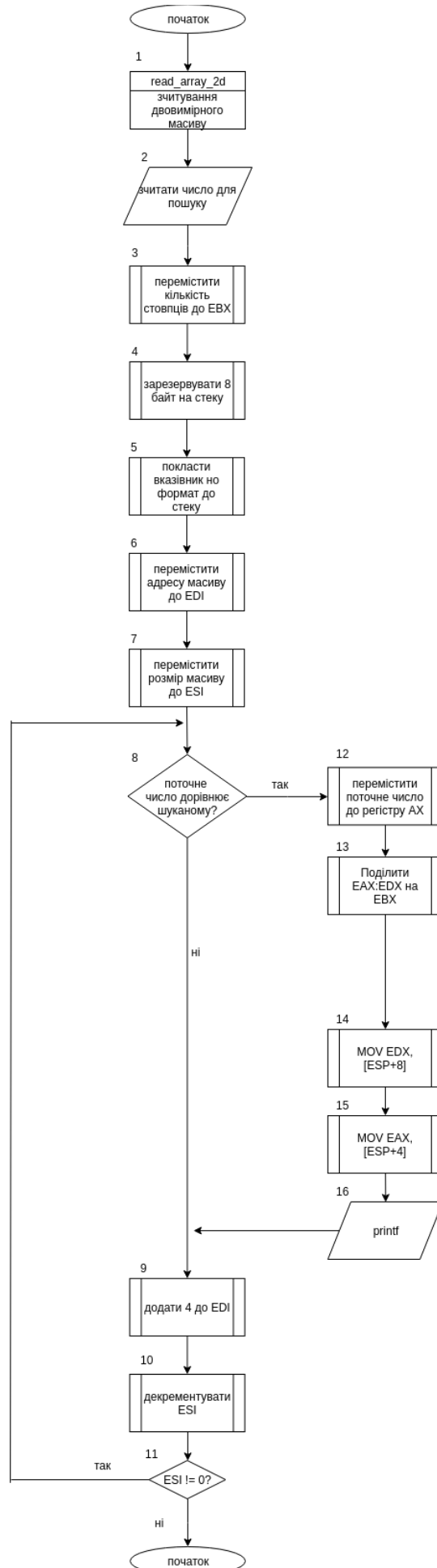
Блок-схема процедури зчитування масиву з введенням його розміру:



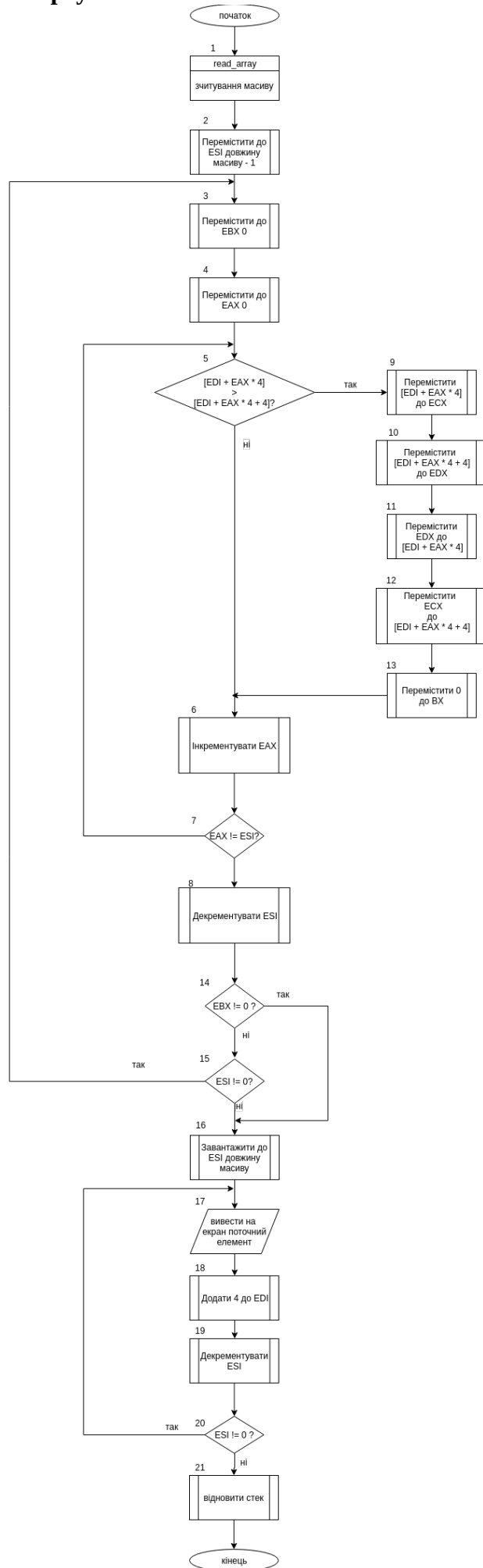
Блок-схема процедури знаходження суми, максимального і мінімального елементів:



Блок-схема процедури знаходження всіх входжень елементу до двовимірного масиву:



Блок-схема процедури сортування:



Результати виконання програм:

```
mogave@mogave-Aspire-5739G ~/Desktop/asm/03 $ echo "Пошук мінімального та максимального елементів, обрахунку суми"; ./array_sum_minmax
Пошук мінімального та максимального елементів, обрахунку суми
This program calculates sum of the array and finds max and min element
Enter array size in range 1 to 20
5
Enter next digit:
-2
Enter next digit:
1
Enter next digit:
2
Enter next digit:
3
Enter next digit:
4
sum: 8, min: -2, max:4
```

```
mogave@mogave-Aspire-5739G ~/Desktop/asm/03 $ echo "Пошук входження елементів до двохвимірної масиви"; ./array_2d_allhits
Пошук входження елементів до двохвимірної масиви
This program finds all matches of number in 2d array
Enter two array sizes(rows,cols) in format 0 0
3 3
Enter next digit:
1
Enter next digit:
2
Enter next digit:
3
Enter next digit:
4
Enter next digit:
1
Enter next digit:
5
Enter next digit:
6
Enter next digit:
7
Enter next digit:
1
Enter a number to look for:
1
Hit! Row: 0, Col: 0
Hit! Row: 1, Col: 1
Hit! Row: 2, Col: 2
```

```
mogave@mogave-Aspire-5739G ~/Desktop/asm/03 $ echo "Сортування масиву"; ./array_sort
Сортування масиву
This program is bubble sort.
Enter array size in range 1 to 20
5
Enter next digit:
0
Enter next digit:
3
Enter next digit:
-2
Enter next digit:
10
Enter next digit:
5
You entered this array:
0 3 -2 10 5
-2 0 3 5 10
```

Висновок:

Була написана програма знаходження суми елементів масиву та пошуку максимального елементу масиву, пошуку координат заданого елементу масиву, сортування масиву методом бульбашкового сортування.