

ДОМАШНЕЕ ЗАДАНИЕ №2 ПО АРХИТЕКТУРЕ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

КОРИЦКИЙ АЛЕКСАНДР
ВАРИАНТ 12

БПИ196

УСЛОВИЕ

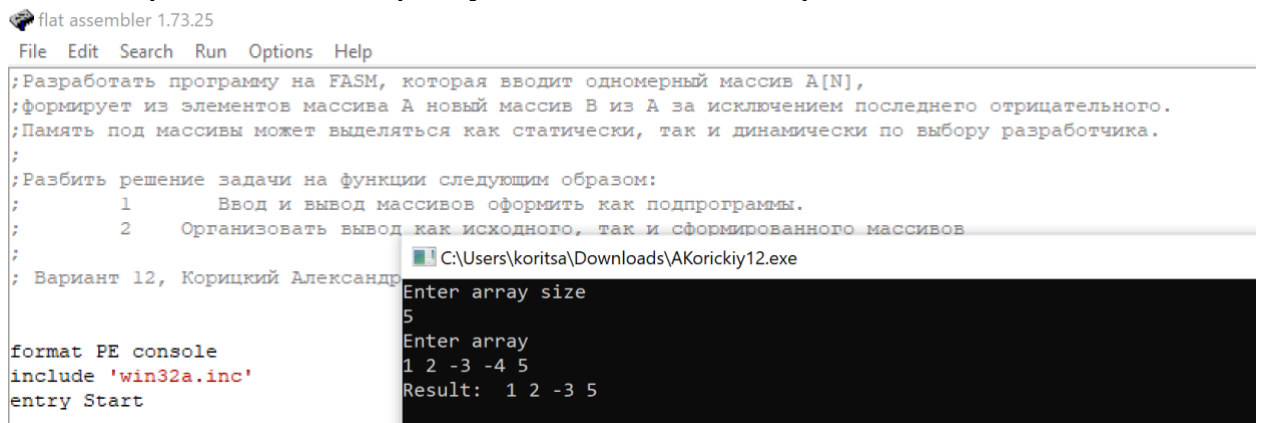
Разработать программу на FASM, которая вводит одномерный массив $A[N]$, формирует из элементов массива A новый массив B из A за исключением последнего отрицательного. Память под массивы может выделяться как статически, так и динамически по выбору разработчика.

Разбить решение задачи на функции следующим образом:

1. Ввод и вывод массивов оформить как подпрограммы.
2. Организовать вывод как исходного, так и сформированного массивов

ТЕСТЫ ПРОГРАММЫ

- Первый тест, который удалит последний отрицательный.



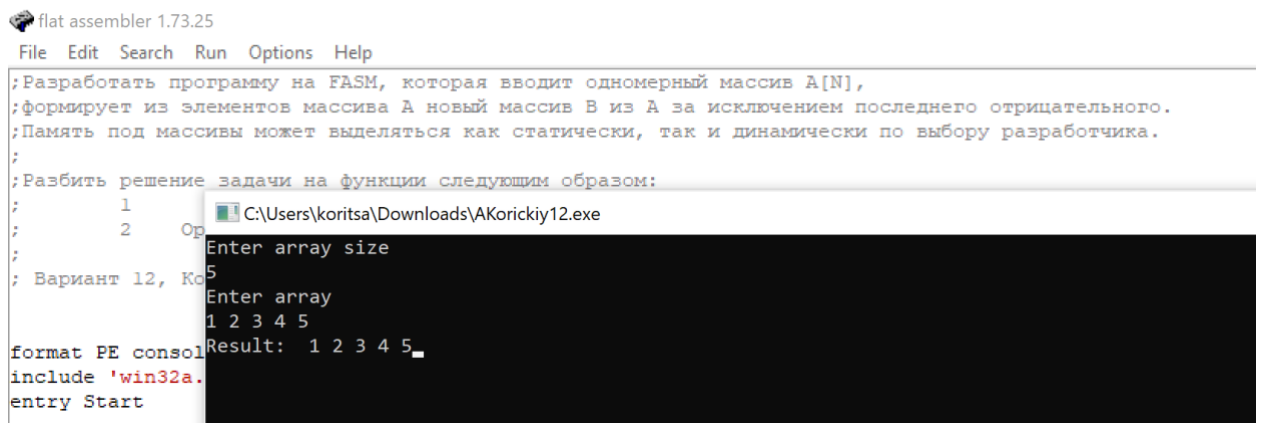
```
flat assembler 1.73.25
File Edit Search Run Options Help

;Разработать программу на FASM, которая вводит одномерный массив A[N],
;формирует из элементов массива A новый массив B из A за исключением последнего отрицательного.
;Память под массивы может выделяться как статически, так и динамически по выбору разработчика.
;
;Разбить решение задачи на функции следующим образом:
; 1 Ввод и вывод массивов оформить как подпрограммы.
; 2 Организовать вывод как исходного, так и сформированного массивов
;
; Вариант 12, Корицкий Александр

format PE console
include 'win32a.inc'
entry Start

Enter array size
5
Enter array
1 2 -3 -4 5
Result: 1 2 -3 5
```

- Проверим, что не удаляет ни одного символа, если они все положительные.



```
flat assembler 1.73.25
File Edit Search Run Options Help

;Разработать программу на FASM, которая вводит одномерный массив A[N],
;формирует из элементов массива A новый массив B из A за исключением последнего отрицательного.
;Память под массивы может выделяться как статически, так и динамически по выбору разработчика.
;
;Разбить решение задачи на функции следующим образом:
; 1 Ввод и вывод массивов оформить как подпрограммы.
; 2 Организовать вывод как исходного, так и сформированного массивов
;
; Вариант 12, Корицкий Александр

format PE console
include 'win32a.inc'
entry Start

Enter array size
5
Enter array
1 2 3 4 5
Result: 1 2 3 4 5
```

- Проверим с нулем.

```
flat assembler 1.73.25
File Edit Search Run Options Help
;Разработать программу на FASM, которая вводит одномерный массив A[N],
;формирует из элементов массива A новый массив B из A за исключением последнего отрицательного.
;Память под массивы может выделяться как статически, так и динамически по выбору разработчика.
;
;Разбить решение задачи на функции следующим образом:
; 1 Ввод и вывод массивов оформить как подпрограммы.
; 2 Организовать вывод результата на экран.
;
; Вариант 12, Коричневый
format PE console
include 'win32a.inc'
entry Start
```

Все верно.

ЛИСТИНГ ПРОГРАММЫ

```
format PE console
include 'win32a.inc'
entry Start
```

```
.code:
```

```
proc Start
    call LoadDataSize
    call AllocMemory
    call LoadData
    call FormSecondArray
    call Print
    call FreeMemory
    call Stop
    ret
endp
proc FreeMemory
    cinvoke free,[StartArray]
    cinvoke free,[NewArray]
    ret
endp
proc LoadDataSize
    cinvoke printf,Enter_size
    cinvoke scanf,scnaf_integer,StartArraySize
    ret
endp
proc LoadData
    cinvoke printf,Enter_elements
```

```

    mov eax,[StartArray]
    mov ecx,[StartArraySize]
    .loadLoop:
    pusha
    cinvoke scanf,scnaf_integer,eax
    popa
    add eax,4
    loop .loadLoop
    ret
endp
proc AllocMemory
    mov eax,[StartArraySize]
    shl eax,2
    push eax
    cinvoke malloc,eax
    mov [StartArray],eax
    mov eax,[StartArraySize]
    pop eax
    cinvoke malloc,eax
    mov [NewArray],eax
    ret
endp
proc FormSecondArray
    mov edx,[StartArray]
    mov ecx,[StartArraySize]
    lea edx,[edx+ecx*4-4]
    mov ebx,[NewArray]
    xor eax,eax
    xor esi,esi
    .l:
    mov edi,[edx]
    cmp edi,0
    jge .ok
    cmp esi,0
    jne .ok
    inc esi
    jmp .skip
    .ok:
    mov [ebx],edi
    add ebx,4
    inc eax
    .skip:
    sub edx,4
    dec ecx
    jnz .l

```

```

    mov [NewArraySize],eax
    mov eax,[NewArray]
    mov ecx,[NewArraySize]
    lea edx,[eax+ecx*4-4]
.l2:
    mov ebx,[eax]
    xchg [edx],ebx
    mov [eax],ebx
    add eax,4
    sub edx,4
    cmp eax,edx
    jb .l2
    ret
endp
proc Print
    cinvoke printf,Result
    mov edx,[NewArray]
    mov ecx,[NewArraySize]
    cmp ecx,0
    je .skip
.m4:
    pusha
    cinvoke printf,printf_integer,[edx]
    popa
    add edx,4
    loop .m4
.skip:
    ret
endp
proc Stop
    cinvoke scanf,scnaf_integer,a
    ret
endp

.data:
a dd 0
StartArray dd 0
NewArray dd 0
StartArraySize dd 0
NewArraySize dd 0
scnaf_integer db '%d',0
printf_integer db ' %d',0
Enter_size db 'Enter array size ',13,10,0
Enter_elements db 'Enter array ',13,10,0
Result db 'Result: ',0

```

```
section '.import' import readable
```

```
library msvcrt,'msvcrt.dll'
```

```
import msvcrt,\n    printf,'printf',\n    scanf,'scanf',\n    free,'free',\n    malloc,'malloc'
```