

АВС - Домашнее задание №2

Арыслан Якшибаев, группа БПИ196*

2020-10-08 Thu 21:14

Contents

Условие

Разработать программу, которая вводит одномерный массив $A[N]$, формирует из элементов массива A новый массив B по правилам, указанным в таблице, и выводит его.

Вариант: 8 Массив B из... элементов A в обратном порядке

Код основной программы

```
;; Арыслан Якшибаев, группа БПИ196.  
;;  
;; Разработать программу, которая вводит одномерный массив A[N],  
;; формирует из элементов массива A новый массив B по правилам,  
;; указанным в таблице, и выводит его.  
;;  
;; Вариант: 10  
;;  
;; Массив B из... элементов A в обратном порядке  
;;
```

```
%include "util.asm"
```

```
SECTION .data
```

*aayakshibaev@edu.hse.ru

```

initPrompt      db "Введите длину массива", 0h
arrayPrompt     db "Введите числа массива", 0h
inputMsg        db ">> ", 0h
outputMsg       db "<< ", 0h
space           db " ", 0h
firstArrayString db "Введённый массив: ", 0h
newArrayString  db "Новый массив: ", 0h
zeroMsg         db "Пустой массив, нечего делать. ", 0h

SECTION .bss

sinput          resb 255      ; reserve space for 255 bytes
array           resb 255      ; reserve space for 255 bytes
newArray        resb 255      ; reserve space for 255 bytes

SECTION .text

global _start

_start:
    ;; Prompt for N
    mov eax, initPrompt
    call sprintfLF

    ;; Read the number N
    call readInt
    push eax

    ;; if N == 0, quit
    cmp eax, 0
    jle gotZero

    ;; mov eax, outputMsg
    ;; call sprintf_err
    ;; mov eax, [arrayLen]
    ;; call sprintfLF_err

    ;; Prompt for array

```

```

mov eax, arrayPrompt
call sprintLF

;; Read the array from STDIN
pop eax
push eax
mov ebx, array
call readIntArray

;; Copy array to newArray
mov eax, array
mov ebx, newArray
pop ecx
push ecx
call copyArray

;; Reverse newArray
mov ebx, newArray
call reverseArray

;; Print decorative text
mov eax, firstArrayString
call sprint

;; print initial array
mov ebx, array
pop eax
push eax
call printIntArray

;; Decorative text
mov eax, newArrayString
call sprint

;; Print new, reversed array
mov ebx, newArray
pop eax
call printIntArray

finishProgram:

```

```

        call quit

gotZero:
        ;; Display "0 elements, nothing to do"
        mov eax, zeroMsg
        call sprintf
        jmp finishProgram

```

Сборка (Linux x64)

```

nasm -f elf main.asm
ld -m elf_i386 main.o -o main

```

Запуск

```
./main
```

Примеры работы

Обычное использование

```

./main
Введите длину массива
>> 5
Введите числа массива
>> 1
>> 2
>> 3
>> 4
>> 5
Введённый массив: 1 2 3 4 5
Новый массив:      5 4 3 2 1

```

Пустой массив

```

./main
Введите длину массива
>> 0
Пустой массив, нечего делать.

```

Один элемент в массиве

```
./main  
Введите длину массива  
>> 1  
Введите числа массива  
>> 5  
Введённый массив: 5  
Новый массив:      5
```

Негативное количество элементов

```
./main  
Введите длину массива  
>> -1  
Пустой массив, нечего делать.
```