

# ИДЗ 1

Исполнитель: Чапурина Валерия Сергеевна

Группа: БПИ237

Вариант: 6

## Условие задания

Разработать программу, которая вводит одномерный массив А, состоящий из N элементов (значение N вводится при выполнении программы), после чего формирует из элементов массива А новый массив

В по правилам, указанным в варианте, и выводит его. Память под массивы может выделяться статически, на стеке, автоматически по выбору разработчика с учетом требований к оценке работы.

При решении задачи необходимо использовать подпрограммы для реализации ввода, вывода и формирования нового массива массива.

Допустимы (при необходимости) дополнительные подпрограммы.

Максимальное количество элементов в массиве не должно превышать 10 (ограничение обуславливается вводом данных с клавиатуры). При этом необходимо обрабатывать некорректные значения как для нижней, так и для верхней границ массивов в зависимости от условия задачи.

6

Сформировать массив В, состоящих из элементов массива А, значения которых кратны введенному числу X.

## Текст программы

```
.include "macrolib.s"
.data
.align 2
    array: .space 64    # Выделяем 64 байта (больше, чем надо) - это будет массив В
    arrend:           # Граница массива
.text
```

```

# принимает число в регистре %x, кладёт |x| в регистр %y
.macro abs(%x, %y)
    bge %x, zero, positive    # Если %x ≥ 0, переход к метке "positive"
    neg %y, %x                # Иначе кладём -%x в %y
    j end_abs                 # Переход к концу

    positive:
        mv %y, %x             # Сохраняем значение %x в %y

    end_abs:
.end_macro

main:
    print_str("Введите n - количество элементов в массиве (от 1 до 10):")
    read_int(s0)
    print_str("\n")

    # Запишем верхнюю границу отрезка для n в t1
    li t1 10

    # Проверим соответствие границ
    blez s0 number_out_of_range_down
    bgt s0 t1 number_out_of_range_up

    print_str("Введите X - число, на которое должны делиться элементы массива (≥ 1)")
    read_int(s1)
    print_str("\n")

    # Проверим, что X > 0
    blez s1 number_out_of_range_down

    la t0 array    # Указатель на текущий элемент массива
    li s2 0        # Счётчик i = 0 - количество считанных элементов массива A
    li s3 0        # Счётчик j = 0 - количество элементов массива B
    j input_loop

number_out_of_range_down:
    print_str("Введённое число должно быть ≥ 1")
    exit
number_out_of_range_up:
    print_str("Введённое число должно быть ≤ 10")
    exit
input_loop:
    addi s2 s2 1    # Увеличиваем счётчик i на 1
    bgt s2 s0 init  # Если счётчик стал равным n - завершаем цикл ввода

```

```

print_str("Введите число - элемент массива A: ")

# Получаем от пользователя элемент массива
li a7 5
ecall

abs(a0, t3)
rem t2 t3 s1 # t2 = t3 % s1 = остаток от модуля введённого числа от деления на X
bgtz t2 input_loop # если остаток не равен 0, продолжаем ввод

sw a0 (t0) # Записываем введённое число в массив
addi t0 t0 4 # Перемещаем указатель на 4 байта (размер int)
addi s3 s3 1 # Увеличиваем счётчик j на 1

j input_loop # Продолжаем цикл ввода элементов
init:
la t0 array # Массив заполнили ⇒ указатель должен быть на первом
# элементе массива, чтобы пройти по нему заново
li s2 1 # i = 1
j print_loop
print_loop:
bgt s2 s3 exit # i > j ⇒ выводим результат

lw a0 (t0) # Записываем в a0 текущий элемент массива
print_int(a0)
print_str(" ")

j next_iter
next_iter:
addi s2 s2 1 # ++i
addi t0 t0 4 # двигаем указатель на 4 байта
j print_loop
exit:
li a7, 10
ecall

```

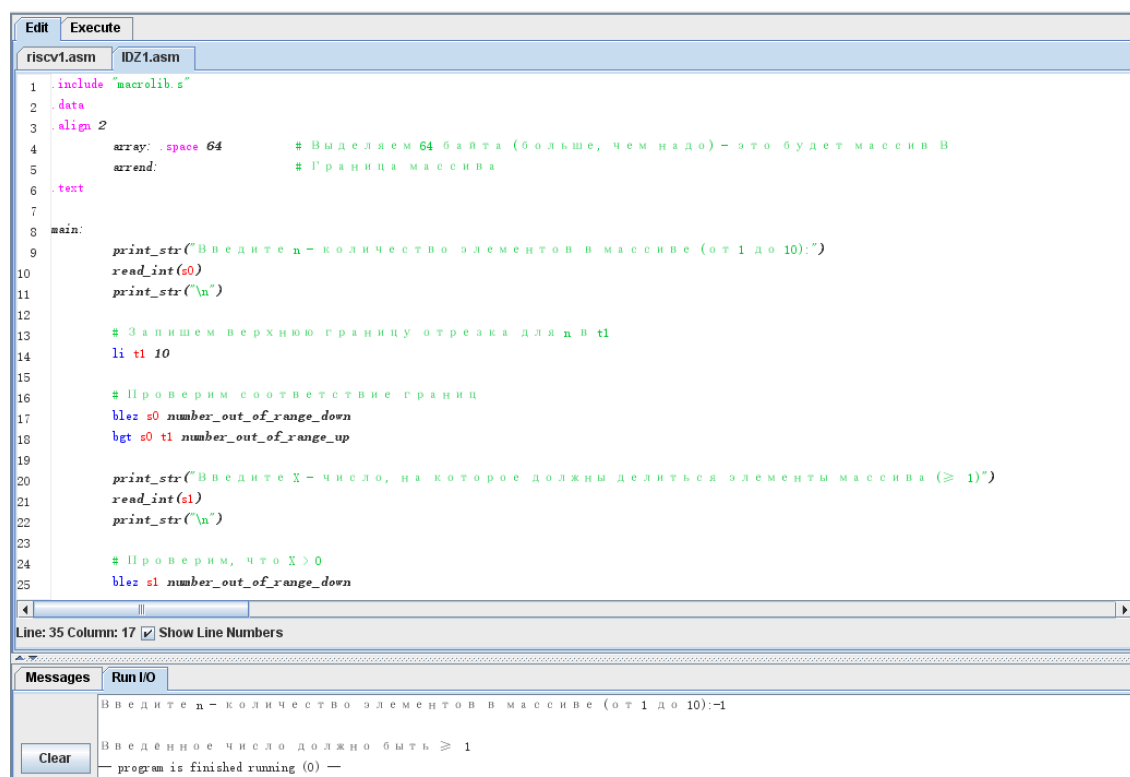
## Тестирование

### Тест 1 - количество элементов массива A отрицательное

Ввод: N = - 1

Предполагаемый вывод: *Введённое число должно быть  $\geq 1$*

Вывод:



The screenshot shows an IDE with two tabs: 'riscv1.asm' and 'IDZ1.asm'. The 'IDZ1.asm' tab is active, displaying assembly code. The code includes a macro library, defines data (array, arrend), and a main function. The main function prompts the user to enter the number of elements in the array (n), reads the input, and prints it. It then sets a limit (t1) to 10 and checks if the input is within the range [1, 10]. If not, it prompts the user to enter a number X that the array elements should be divisible by (X ≥ 1). The code also includes comments in Russian. The output window at the bottom shows the execution results, including the prompts and the user's input.

```
1 .include "macrolib.s"
2
3 .data
4     array: .space 64          # Выделяем 64 байта (больше, чем надо) - это будет массив В
5     arrend:                  # Граница массива
6
7 .text
8
9 main:
10     print_str("Введите n - количество элементов в массиве (от 1 до 10):")
11     read_int($0)
12     print_str("\n")
13
14     # Запишем верхнюю границу отрезка для n в t1
15     li t1, 10
16
17     # Проверим соответствие границ
18     blez $0, number_out_of_range_down
19     bgt $0, t1, number_out_of_range_up
20
21     print_str("Введите X - число, на которое должны делиться элементы массива (≥ 1)")
22     read_int($1)
23     print_str("\n")
24
25     # Проверим, что X > 0
26     blez $1, number_out_of_range_down
```

Line: 35 Column: 17 ☒ Show Line Numbers

Messages Run I/O

Введите n - количество элементов в массиве (от 1 до 10):-1

Введённое число должно быть  $\geq 1$

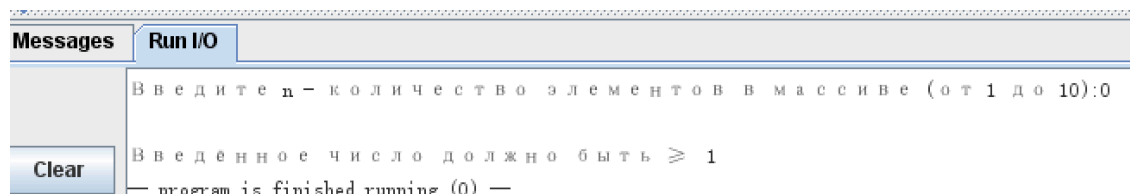
program is finished running (0) —

Тест 2 - количество элементов массива A равно 0

Ввод: N = 0

Предполагаемый вывод: *Введённое число должно быть  $\geq 1$*

Вывод:



The screenshot shows the 'Messages' and 'Run I/O' tabs of the IDE. The 'Run I/O' tab is active, displaying the execution results. The output shows the prompt 'Введите n - количество элементов в массиве (от 1 до 10):' followed by the user's input '0'. The program then prints 'Введённое число должно быть  $\geq 1$ ' and 'program is finished running (0) —'.

Messages Run I/O

Введите n - количество элементов в массиве (от 1 до 10):0

Введённое число должно быть  $\geq 1$

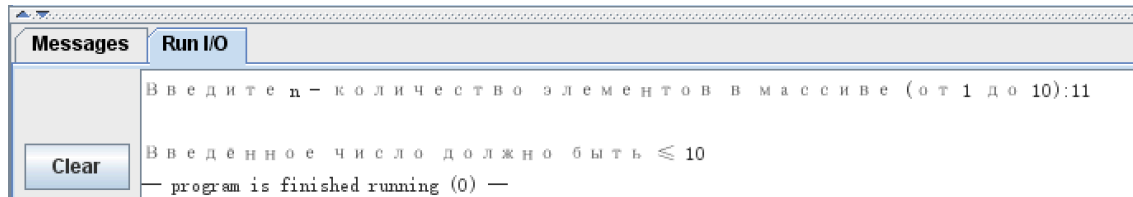
program is finished running (0) —

Тест 3 - количество элементов массива A больше верхней границы (10)

Ввод: N = 11

Предполагаемый вывод: *Введённое число должно быть  $\leq 10$*

Вывод:

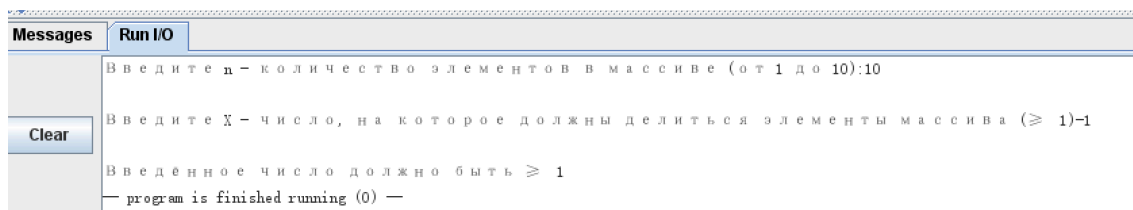


**Тест 4 - количество элементов A может быть 10, но X не может быть отрицательным**

Ввод: N = 10, X = -1

Предполагаемый вывод: *Введённое число должно быть  $\geq 1$*

Вывод:

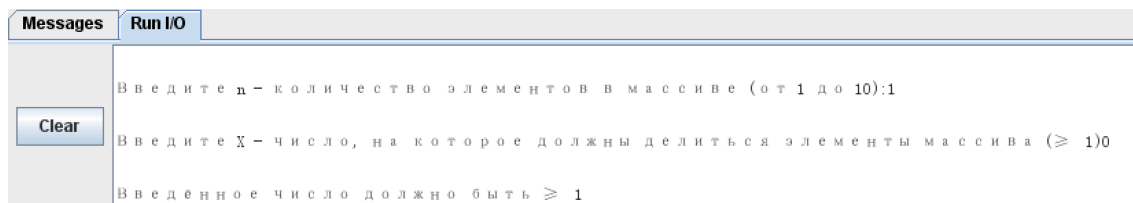


**Тест 5 - количество элементов A может быть 1, но X не может быть равен 0**

Ввод: N = 1, X = 0

Предполагаемый вывод: *Введённое число должно быть  $\geq 1$*

Вывод:



**Тест 6 - в A один элемент, X = 1**

Ввод: N = 1, X = 1, A = [1]

Предполагаемый вывод: 1

**Вывод:**

| Messages | Run I/O   |
|----------|---|
|          | Введите $n$ – количество элементов в массиве (от 1 до 10):1                     |
| Clear    | Введите $X$ – число, на которое должны делиться элементы массива ( $\geq 1$ ):1 |
|          | Введите число – элемент массива $A$ : 1   |
|          | 1   |

**Тест 7 - в A один элемент и он делится на X**

**Ввод:**  $N = 1, X = 2, A = [8]$

**Предполагаемый вывод:** 8

**Вывод:**

| Messages | Run I/O   |
|----------|---|
|          | Введите $n$ – количество элементов в массиве (от 1 до 10):1                     |
| Clear    | Введите $X$ – число, на которое должны делиться элементы массива ( $\geq 1$ ):2 |
|          | Введите число – элемент массива $A$ : 8   |
|          | 8   |

**Тест 8 - в A один элемент и он не делится на X**

**Ввод:**  $N = 1, X = 2, A = [7]$

**Предполагаемый вывод:**

**Вывод:**

| Messages | Run I/O   |
|----------|---|
|          | Введите $n$ – количество элементов в массиве (от 1 до 10):1                     |
| Clear    | Введите $X$ – число, на которое должны делиться элементы массива ( $\geq 1$ ):2 |
|          | Введите число – элемент массива $A$ : 7   |
|          | 7   |

**Тест 9 - в A какое-то количество элементов от 2 до 9, некоторые из них делятся на X, элемент равен 0**

**Ввод:**  $N = 5, X = 3, A = [0, 3, 5, 7, 9]$

**Предполагаемый вывод:** 0 3 9

**Вывод:**

| Messages         | Run I/O  |
|------------------|--|
| <div>Clear</div> | Введите n - количество элементов в массиве (от 1 до 10): 5                     |
|                  | Введите X - число, на которое должны делиться элементы массива ( $\geq 1$ ): 3 |
|                  | Введите число - элемент массива A: 0   |
|                  | Введите число - элемент массива A: 3   |
|                  | Введите число - элемент массива A: 5   |

| Messages         | Run I/O                              |
|------------------|--------------------------------------|
| <div>Clear</div> | Введите число - элемент массива A: 5 |
|                  | Введите число - элемент массива A: 7 |
|                  | Введите число - элемент массива A: 9 |
|                  | 0 3 9                                |
|                  | — program is finished running (0) —  |

**Тест 10 - в A какое-то количество элементов от 2 до 9, некоторые из них делятся на X, N = X, отрицательный элемент**

**Ввод: N = 4, X = 4, A = [-4, 3, 8, 7]**

**Предполагаемый вывод: -4 8**

**Вывод:**

| Messages         | Run I/O  |
|------------------|--|
| <div>Clear</div> | Введите n - количество элементов в массиве (от 1 до 10): 4                     |
|                  | Введите X - число, на которое должны делиться элементы массива ( $\geq 1$ ): 4 |
|                  | Введите число - элемент массива A: -4  |
|                  | Введите число - элемент массива A: 3   |

| Messages         | Run I/O                              |
|------------------|--------------------------------------|
| <div>Clear</div> | Введите число - элемент массива A: 3 |
|                  | Введите число - элемент массива A: 8 |
|                  | Введите число - элемент массива A: 7 |
|                  | -4 8                                 |
|                  | — program is finished running (0) —  |

**Тест 11 - в A какое-то количество элементов от 2 до 9, все делятся на X**

Ввод:  $N = 5$ ,  $X = 3$ ,  $A = [0, 3, 6, 27, 9]$

Предполагаемый вывод: 0 3 6 27 9

Вывод:

| Messages | Run I/O  |
|----------|--|
|          | Введите n - количество элементов в массиве (от 1 до 10): 5                     |
| Clear    | Введите X - число, на которое должны делиться элементы массива ( $\geq 1$ ): 3 |
|          | Введите число - элемент массива A: 0   |
|          | Введите число - элемент массива A: 3   |
|          | Введите число - элемент массива A: 6   |

| Messages | Run I/O                               |
|----------|---------------------------------------|
|          | Введите число - элемент массива A: 6  |
|          | Введите число - элемент массива A: 27 |
|          | Введите число - элемент массива A: 9  |
| Clear    | 0 3 6 27 9                            |
|          | — program is finished running (0) —   |

Тест 12 - в A какое-то количество элементов от 2 до 9, ни один не делится на X

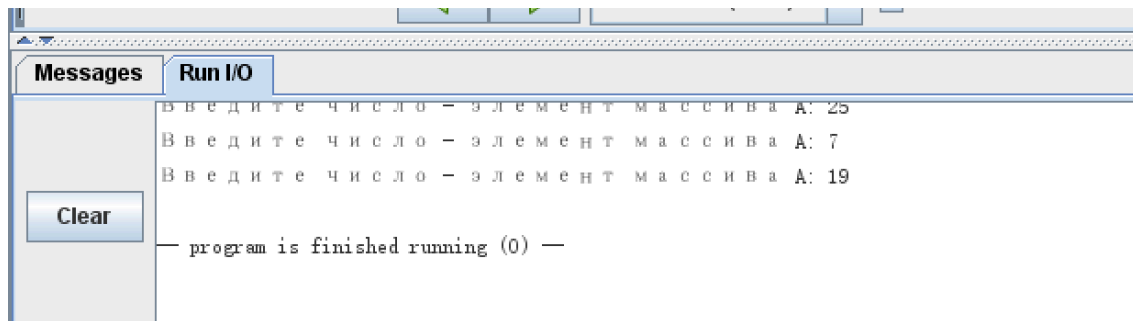
Ввод:  $N = 5$ ,  $X = 3$ ,  $A = [-1, 5, 25, 7, 19]$

Предполагаемый вывод:

Вывод:

| Messages | Run I/O  |
|----------|--|
|          | Введите n - количество элементов в массиве (от 1 до 10): 5                     |
| Clear    | Введите X - число, на которое должны делиться элементы массива ( $\geq 1$ ): 3 |
|          | Введите число - элемент массива A: -1  |
|          | Введите число - элемент массива A: 5   |
|          | Введите число - элемент массива A: 25  |



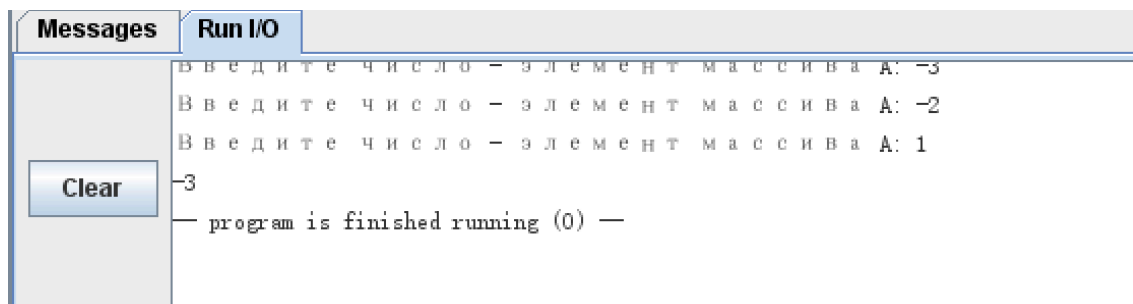
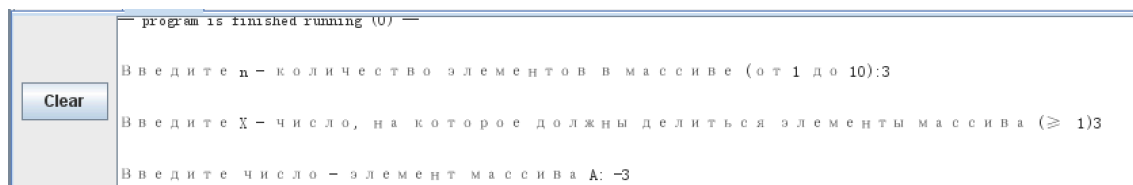


**Тест 13 - в A какое-то количество элементов от 2 до 9, один отрицательный элемент делится на X, один не делится**

**Ввод: N = 3, X = 3, A = [-3, -2, 1]**

**Предполагаемый вывод: -3**

**Вывод:**



**Соответствие требованиям на оценку: 7**

## 4–5 баллов

- Приведено решение задачи на ассемблере. Ввод данных осуществляется с клавиатуры. Вывод данных осуществляется на дисплей.
- В программе должны присутствовать комментарии, поясняющие выполняемые действия.
- Допускается использование требуемых подпрограмм без параметров и локальных переменных.
- В отчете должно быть представлено полное тестовое покрытие. Приведены результаты тестовых прогонов. Например, с использованием скриншотов.

При невыполнении хотя бы одного из требований оценка снижается.

- Решение задачи на ассемблере приведено, ввод данных совершается с клавиатуры, на дисплей осуществляется вывод элементов сформированного массива В
- В программе присутствуют комментарии
- В решении задачи используется подпрограмма для ввода и формирования нового массива В ( `input_loop` ). Так как никаких действий с массивом А производить не нужно, нам нужны только его элементы, которые вводятся пользователем с клавиатуры, было решено не выделять память под массив А и не записывать туда его элементы, а сразу формировать массив В. Так же используется подпрограмма `print_loop` для вывода элементов массива В. И некоторые дополнительные вспомогательные подпрограммы. Так же в решении были использованы макро из `macrolib` для создания более “лаконичного” и понятного решения и самостоятельно написанный макро `abs`, принимающий на вход 2 параметра: регистр с целым числом, модуль которого надо найти, и регистр, в который модуль нужно записать

Разработать программу, которая вводит одномерный массив  $A$ , состоящий из  $N$  элементов (значение  $N$  вводится при выполнении программы), после чего формирует из элементов массива  $A$  новый массив  $B$  по правилам, указанным в варианте, и выводит его. Память под массивы может выделяться статически, на стеке, автоматически по выбору разработчика с учетом требований к оценке работы.

- В отчёте предоставлено полное тестовое покрытие и приведены результаты тестовых прогонов

## 6–7 баллов

При разработке программы на данную оценку необходимо учитывать все требования, предъявляемые на предшествующие оценки.

- В программе необходимо использовать подпрограммы с передачей аргументов через соответствующие регистры, определяемые конвенцией по их использованию. При нехватке этих регистров для их хранения необходимо использовать стек.
- в подпрограммы передаются аргументы через соответствующие регистры. Через регистры типа *t* передаются аргументы, значения в которых мы не боимся потерять (и дальше не используем в программе), через регистры типа *s* передаём значения, которые нам будут нужны далее в программе

## Переменные в программе

- *s0* - тут хранится *N*. Оно нужно нам в *input\_loop*, поэтому записываем в регистр *s*
- *t1* - тут храним верхнюю границу *N*, оно нам нужно только в *main* для проверки условия
- *s1* - тут хранится *X*. Оно нам нужно в *input\_loop* для поиска остатка от деления на *X*
- *t0* - указатель на текущий элемент массива *B*. На протяжении всей программы он меняется, поэтому выбран "временный" регистр
- *s2* - *i*, *s3* - *j* - нам нужны на протяжении всей программы
- *t2* - остаток для деления, нужен нам только для одной проверки
- *t3* - модуль числа. Нужен нам только для нахождения остатка от деления

Все переменные поместились в регистры, поэтому стек не использовался

- Локальные переменные допускается размещать в свободных регистрах, при нехватке которых необходимо их сохранять на стеке. Или сразу все локальные переменные отображать на стек.
- В местах вызова функции добавить комментарии, описывающие передачу фактических параметров и перенос возвращаемого результата. При этом необходимо отметить, в каких регистрах или где на стеке отображаются соответствующие фактические параметры.
- Информацию о проведенных изменениях отобразить в отчете наряду с информацией, необходимой на предыдущую оценку.

При невыполнении хотя бы одного из требований оценка снижается на балл.

- перед макро `abs` описано, как принимаются параметры и возвращается результат; в остальных местах понятно из комментариев