Министерство науки и высшего образования РФ

ФГАОУ ВО Пермский национальный исследовательский

политехнический университет

Кафедра «Вычислительная математика, механика и биомеханика»

Отчет по лабораторной работе № 4

тема «Знакомство с lc3»

по дисциплине «Языки программирования и методы трансляции»

Выполнил: студент группы ИСТ-22-1б Зверев А.А.

Проверил: Батин Сергей Евгеньевич

Пермь, 2023

**Содержание**

[Задание 1 3](#__RefHeading___Toc446_729700978)

[1.1. Постановка задачи 3](#__RefHeading___Toc448_729700978)

[1.2. Решение задачи, код программы 3](#__RefHeading___Toc450_729700978)

[1.3. Тестирование работы программы 4](#__RefHeading___Toc452_729700978)

[Задание 2 5](#__RefHeading___Toc454_729700978)

[2.1. Постановка задачи 5](#__RefHeading___Toc456_729700978)

[2.2. Решение задачи, код программы 5](#__RefHeading___Toc458_729700978)

[2.3. Тестирование работы программы 7](#__RefHeading___Toc460_729700978)

[Задание 3 8](#__RefHeading___Toc462_729700978)

[3.1. Постановка задачи 8](#__RefHeading___Toc464_729700978)

[3.2. Решение задачи, код программы 8](#__RefHeading___Toc466_729700978)

[3.3. Тестирование работы программы 9](#__RefHeading___Toc468_729700978)

[Задание 4 11](#__RefHeading___Toc470_729700978)

[4.1. Постановка задачи 11](#__RefHeading___Toc472_729700978)

[4.2. Решение задачи, код программы 12](#__RefHeading___Toc474_729700978)

[4.3. Тестирование работы программы 15](#__RefHeading___Toc476_729700978)

# Задание 1

## 1.1. Постановка задачи

Написать и запустить программу, выводящую строку Hello World.

## 1.2. Решение задачи, код программы

.ORIG x3000

LEA R0, HELLO\_STR

PUTS

HALT

HELLO\_STR .STRINGZ "Hello, World"

.END

## 1.3. Тестирование работы программы

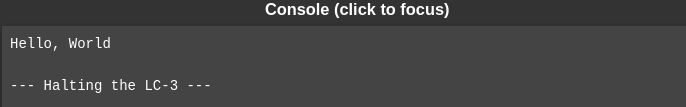


Рис.1.1 Тестирование программы lab4\_1

# Задание 2

## 2.1. Постановка задачи

Написать и запустить программу, принимающую строку пользователя и выводящую ее в терминал.

## 2.2. Решение задачи, код программы

**.ORIG x3000**

**START**

**LEA R0, INPUTING**

**PUTS**

**LEA R1, BUFFER**

**ADD R2, R1, #0**

**READ**

**GETC**

**OUT**

**ADD R4, R0, #-10**

**BRz PRINT**

**STR R0, R2, #0**

**ADD R2, R2, #1**

**BRnzp READ**

**PRINT**

**LEA R0, BUFFER**

**PUTS**

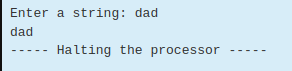
**HALT**

**INPUTING .STRINGZ "Enter a string: "**

**BUFFER .BLKW #80**

**.END**

## 2.3. Тестирование работы программы



Тестирование программы lab4\_2

# Задание 3

## 3.1. Постановка задачи

Написать и запустить программу, реализующую сортировку заданного пользователем целочисленного массива из 10 целых чисел.

## 3.2. Решение задачи, код программы

**.ORIG x3000**

**LEA R0, FIRSTPROMPT**

**PUTS**

**LEA R0, SPACE**

**PUTS**

**LEA R0, SAMPLE**

**PUTS**

**LD R3, POINTER**

**LD R6, IOCOUNTER**

**INPUT**

**IN**

**AND R2, R2, x0**

**AND R5, R5, x0**

**LD R5, HUNDRED**

**LD R2, HEXN48**

**ADD R0, R0, R2**

**ADD R2, R0, x0**

**ADD R0, R0, x0**

**FIRST\_NUM**

**ADD R0, R0, R2**

**ADD R5, R5, x-1**

**BRp FIRST\_NUM**

**ADD R1, R0, x0**

**AND R0, R0, x0**

**IN**

**AND R2, R2, x0**

**AND R5, R5, x0**

**LD R5, TEN**

**LD R2, HEXN48**

**ADD R0, R0, R2**

**ADD R2, R0, x0**

**AND R0, R0, x0**

**SECOND\_NUM**

**ADD R0, R0, R2**

**ADD R5, R5, x-1**

**BRp SECOND\_NUM**

**ADD R4, R0, x0**

**AND R0, R0, x0**

**IN**

**AND R2, R2, x0**

**LD R2, HEXN48**

**ADD R0, R0, R2**

**AND R2, R2, x0**

**ADD R2, R1, R4**

**ADD R2, R0, R2**

**STR R2, R3, x0**

**ADD R3, R3, x1**

**ADD R6, R6, x-1**

**BRp INPUT**

**JSR BUBBLESORT**

**JSR PRODLOOP**

**HALT**

**BUBBLESORT**

**AND R3, R3, x0**

**LD R3, POINTER**

**AND R4, R4, x0**

**LD R4, IOCOUNTER**

**AND R5, R5, x0**

**LD R5, IOCOUNTER**

**OUTLOOP**

**ADD R4, R4, x-1**

**BRz SORTED**

**ADD R5, R4, x0**

**LD R3, POINTER**

**INLOOP**

**LDR R0, R3, x0**

**LDR R1, R3, x1**

**AND R2, R2, x0**

**NOT R2, R1**

**ADD R2, R2, x1**

**ADD R2, R0, R2**

**BRn AUTOSWAP**

**STR R1, R3, x0**

**STR R0, R3, x1**

**AUTOSWAP**

**ADD R3, R3, x1**

**ADD R5, R5, x-1**

**BRp INLOOP**

**BRzp OUTLOOP**

**SORTED RET**

**RET**

**PRODLOOP**

**LEA R0, PROMPTEX**

**PUTS**

**LD R3, POINTER**

**LD R6, IOCOUNTER**

**RESLOOP**

**AND R1, R1, x0**

**AND R2, R2, x0**

**AND R4, R4, x0**

**AND R5, R5, x0**

**AND R0, R0, x0**

**LD R0, SPACE**

**OUT**

**AND R0, R0, x0**

**LDR R0, R3, x0**

**LD R2, HUNDRED**

**NOT R2, R2**

**ADD R2, R2, x1**

**MINUS1**

**ADD R1, R1, x1**

**ADD R0, R0, R2**

**BRzp MINUS1**

**REMAINDER1**

**AND R2, R2, x0**

**LD R2, HUNDRED**

**ADD R0, R0, R2**

**ADD R1, R1, x-1**

**STI R1, FIRSTNUM**

**AND R2, R2, x0**

**LD R2, TEN**

**NOT R2, R2**

**ADD R2, R2, x1**

**MINUS2**

**ADD R4, R4, x1**

**ADD R0, R0, R2**

**BRzp MINUS2**

**REMAINDER2**

**AND R2, R2, x0**

**LD R2, TEN**

**ADD R5, R0, R2**

**STI R5, THIRDNUM**

**ADD R4, R4, x-1**

**STI R4, SECONDNUM**

**AND R0, R0, x0**

**LDI R0, FIRSTNUM**

**AND R2, R2, x0**

**LD R2, HEX48**

**ADD R0, R0, R2**

**OUT**

**AND R0, R0, x0**

**LDI R0, SECONDNUM**

**AND R2, R2, x0**

**LD R2, HEX48**

**ADD R0, R0, R2**

**OUT**

**AND R0, R0, x0**

**LDI R0, THIRDNUM**

**AND R2, R2, x0**

**LD R2, HEX48**

**ADD R0, R0, R2**

**OUT**

**ADD R3, R3, x1**

**ADD R6, R6, x-1**

**BRp RESLOOP**

**HALT**

**FIRSTPROMPT .STRINGZ "Input 10 num (0-999) with 3 digits"**

**SAMPLE .STRINGZ "Input exaple: 020 - 20, 003 - 3 and etc."**

**PROMPTEX .STRINGZ "Numbers in ascending order:"**

**SPACE .STRINGZ "\n"**

**HEXN48 .FILL xFFD0**

**HEX48 .FILL x0030**

**HUNDRED .FILL x0064**

**TEN .FILL x000A**

**POINTER .FILL x4000**

**IOCOUNTER .FILL #10**

**FIRSTNUM .FILL x400A**

**SECONDNUM .FILL x400B**

**THIRDNUM .FILL x400C**

**.END**

## 3.3. Тестирование работы программы

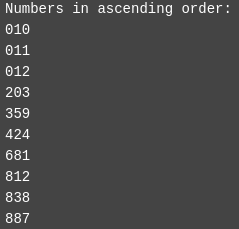


Рис.3.1 Тестирование программы lab3\_3

# Задание 4

## 4.1. Постановка задачи

Написать и запустить программу, реализующую сортировку заданной пользователем строки из 10 цифр.

## 4.2. Решение задачи, код программы

**.ORIG x3000**

**LEA R0, PROMPT**

**PUTS**

**LEA R1, NUMBERS**

**LEA R4, CONSTATEXP**

**LD R2, COUNT**

**READ**

**GETC**

**OUT**

**STR R0, R4, #0**

**ADD R4, R4, #1**

**LD R3, NEG48**

**ADD R0, R0, R3**

**STR R0, R1, #0**

**ADD R1, R1, #1**

**ADD R2, R2, #-1**

**BRp READ**

**LEA R1, NUMBERS**

**LD R2, COUNT**

**ADD R2, R2, #-1**

**SORTLOOP**

**LD R4, COUNT**

**ADD R4, R4, #-1**

**LEA R3, NUMBERS**

**INLOOP**

**LDR R0, R3, #0**

**LDR R1, R3, #1**

**NOT R5, R1**

**ADD R5, R5, #1**

**ADD R5, R0, R5**

**BRn SWAP**

**STR R1, R3, #0**

**STR R0, R3, #1**

**SWAP**

**ADD R3, R3, #1**

**ADD R4, R4, #-1**

**BRp INLOOP**

**ADD R2, R2, #-1**

**BRp SORTLOOP**

**LD R0, NEWLINE**

**OUT**

**LEA R1, ORIGINAL**

**LD R2, COUNT**

**OUTCONSTATEXP**

**LDR R0, R1, #0**

**OUT**

**ADD R1, R1, #1**

**ADD R2, R2, #-1**

**BRp OUTCONSTATEXP**

**LD R0, NEWLINE**

**OUT**

**LEA R1, NUMBERS**

**LD R2, COUNT**

**OUTNUMBERS**

**LDR R0, R1, #0**

**LD R3, POS48**

**ADD R0, R0, R3**

**OUT**

**ADD R1, R1, #1**

**ADD R2, R2, #-1**

**BRp OUTNUMBERS**

**HALT**

**PROMPT .STRINGZ "Enter 10 numbers: "**

**NUMBERS .BLKW 10**

**CONSTATEXP .BLKW 10**

**COUNT .FILL #10**

**NEG48 .FILL #-48**

**POS48 .FILL #48**

**NEWLINE .FILL x000A**

**.END**

## 4.3. Тестирование работы программы

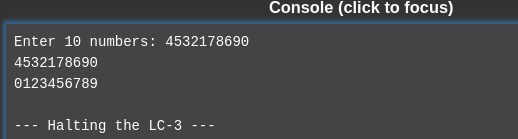


Рис.4.1 Тестирование программы lab4\_4

# Задание 5

## 5.1. Постановка задачи

Реализовать с использованием flex и bison программу, транслирующую простые инструкции ветвления и цикла в ассемблер lc-3.

## 5.2. Решение задачи, код программы

## 5.3. Тестирование работы программы

Рис.4.1 Тестирование программы lab4\_5