ОТЧЁТ

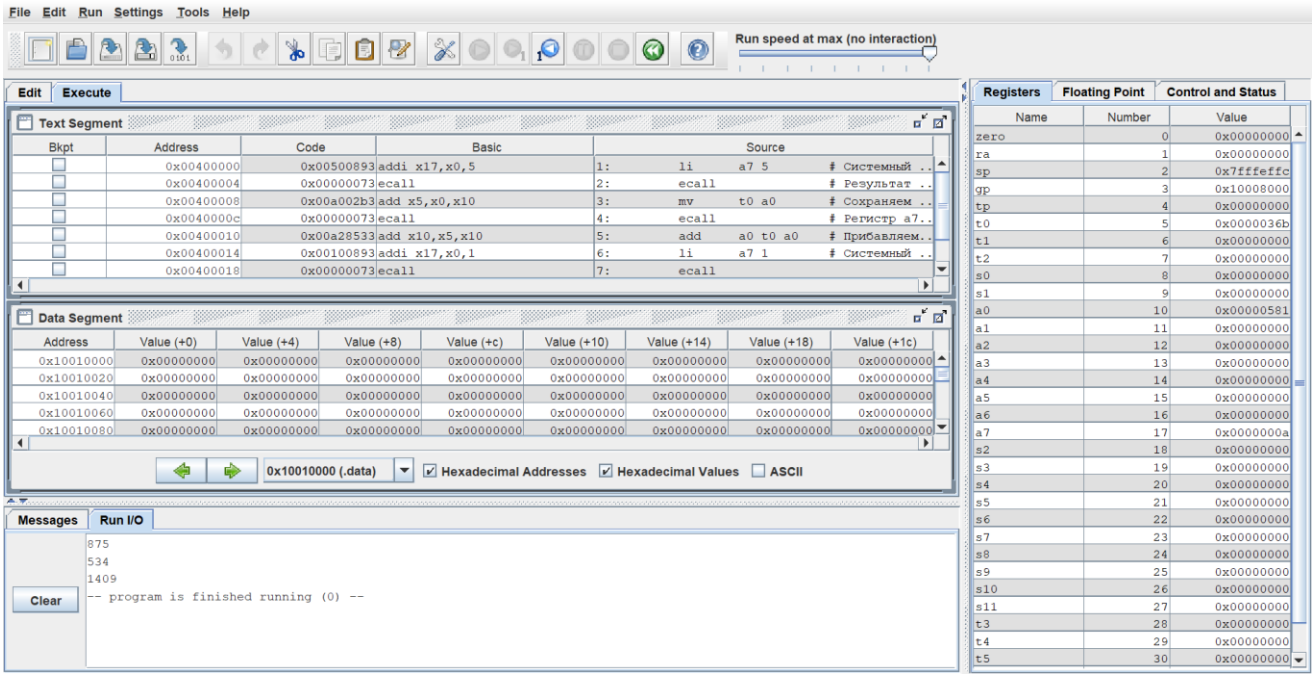
Об использовании эмулятора RARS

Выполнила:
Студентка 2 курса
Программной
инженерии

01-add-int01

```
1       li      a7 5        # Системный вызов №5 — ввести десятичное число
2       ecall               # Результат — в регистре a0
3       mv      t0 a0       # Сохраняем результат в t0
4       ecall               # Регистр a7 не менялся, тот же системный вызов
5       add     a0 t0 a0    # Прибавляем ко второму число первое
6       li      a7 1        # Системный вызов №1 — вывести десятичное число
7       ecall
8       li      a7 10       # Системный вызов №10 — останов программы
9       ecall
```

Результат выполнения программы:



li, mv – псевдокоманды

Типы форматов команд:

1) li – псевдокоманда, вызывающая addi(Immediate)
2) mv - псевдокоманда, вызывающая add(Register)
3) add – R(Register)
4) ecall – формат отсутствует

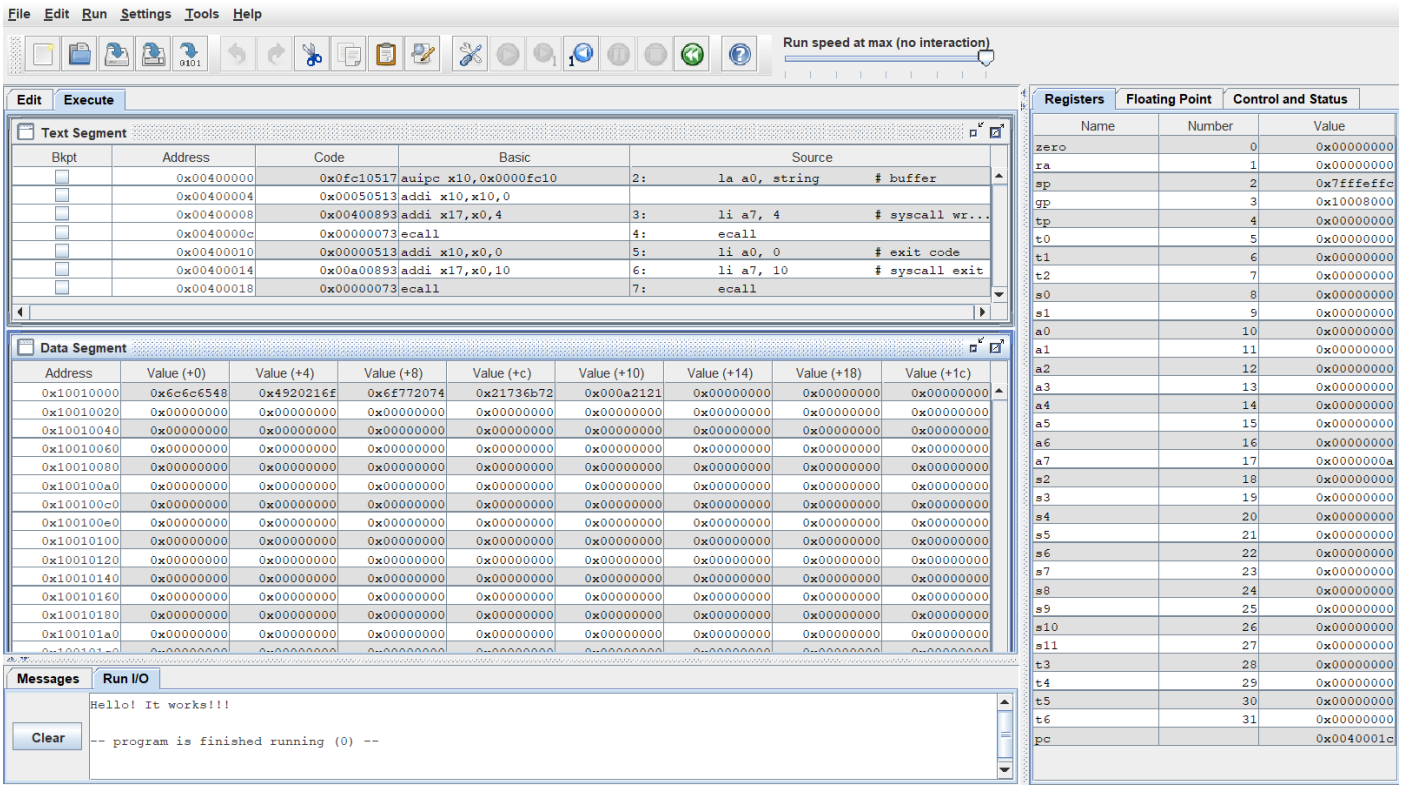
Системные вызовы:

1) Системный вызов №5 (ReadInt) – считывание десятичного числа из консоли
2) Системный вызов №1 (PrintInt) – вывод десятичного числа в консоль
3) Системный вызов №10 (Exit) – завершение программы с кодом 0

02-hello01

```
1  .text
2          la a0, string        # buffer
3          li a7, 4             # syscall write (4)
4          ecall
5          li a0, 0             # exit code
6          li a7, 10            # syscall exit
7          ecall
8  .data
9      string:   .asciz "Hello! It works!!!\n"
```

Результат выполнения программы:



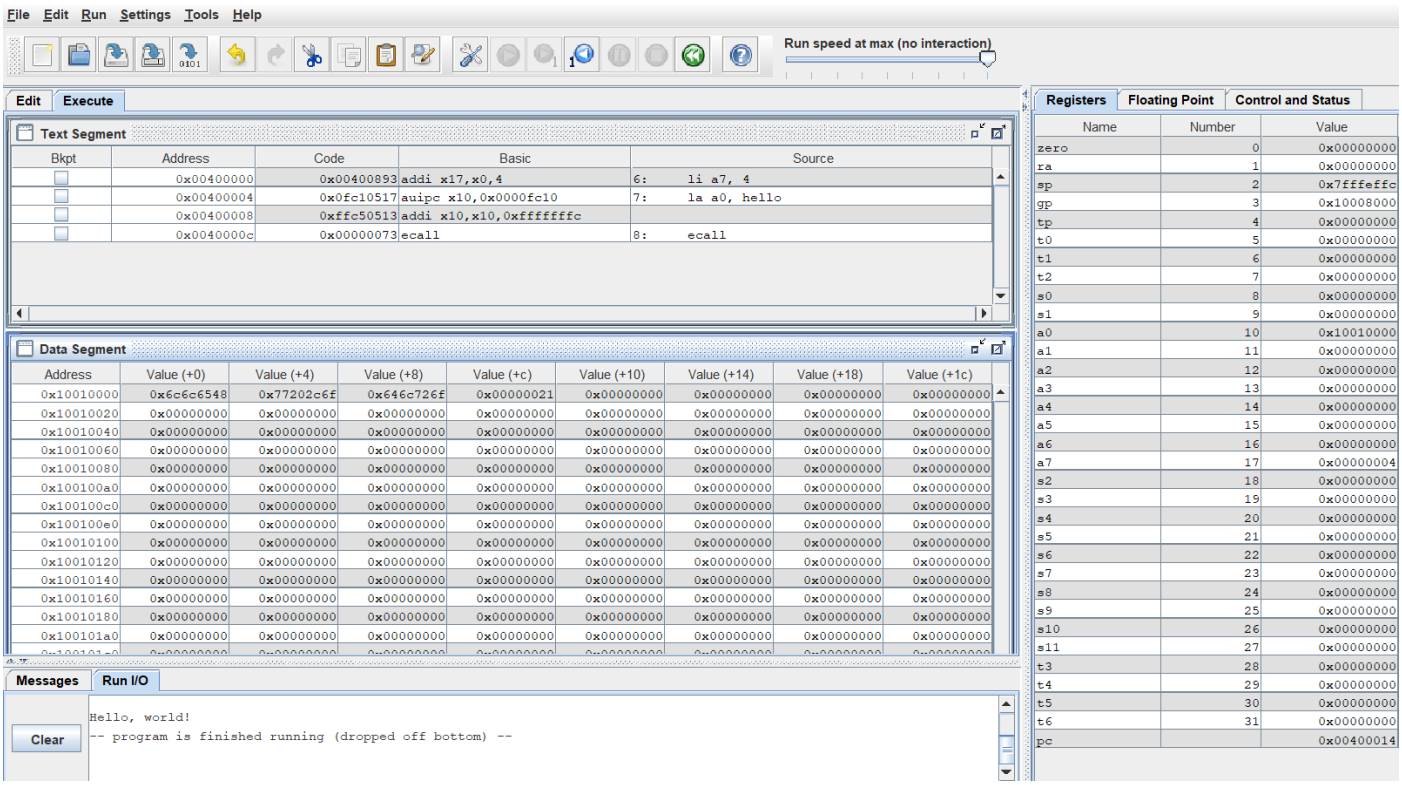
Системные вызовы:

1) Системный вызов №4 (PrintString) – вывод строки в консоль
2) Системный вызов №10 (Exit) – завершение программы с кодом 0

03-hello02

```
1       .data
2   hello:
3       .asciz "Hello, world!"
4       .text
5   main:
6       li a7, 4
7       la a0, hello
8       ecall
```

Результат выполнения программы:



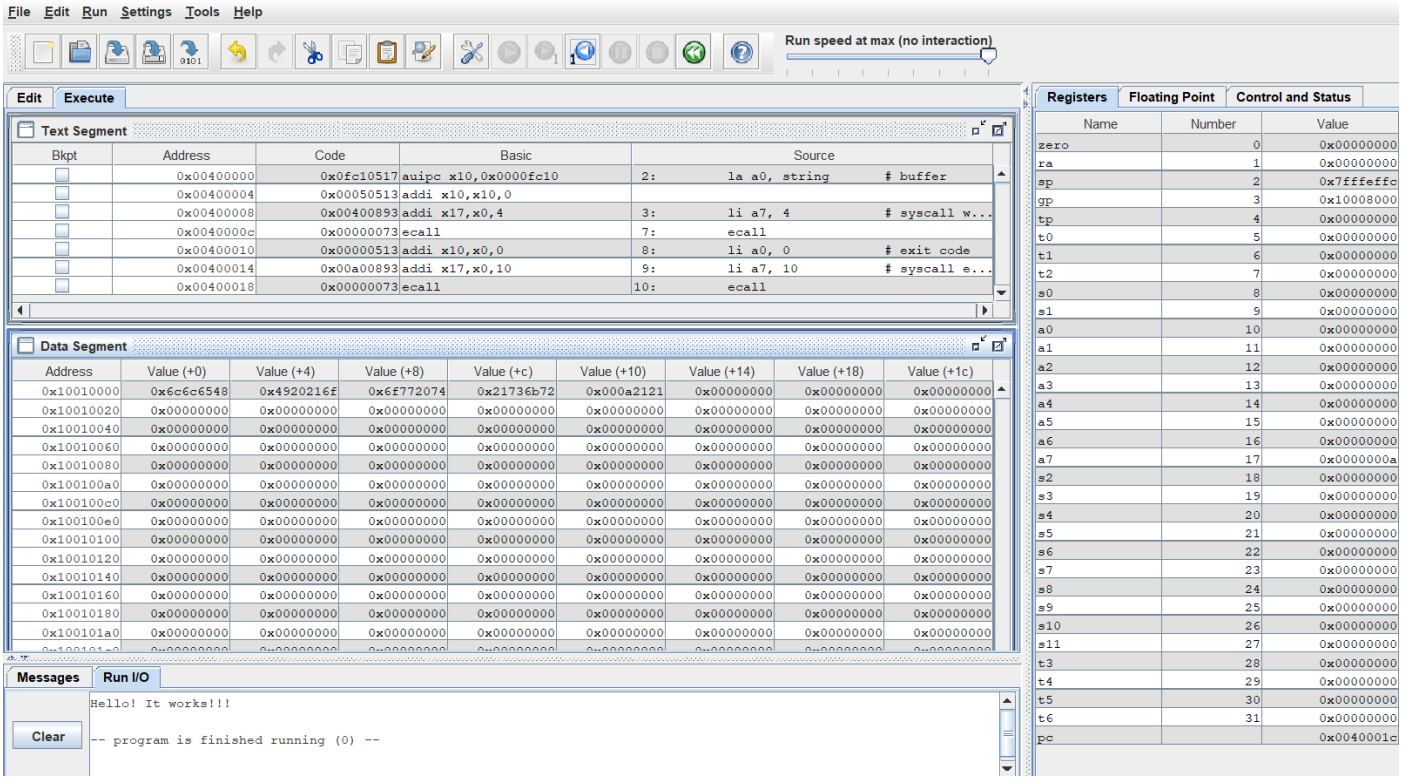
Системные вызовы:

1) Системный вызов №4 (PrintString) – вывод строки в консоль

04-hello03

```
1   .text
2           la a0, string           # buffer
3           li a7, 4                # syscall write (4)
4   .data
5       string:   .asciz "Hello! It works!!!\n"
6   .text
7           ecall
8           li a0, 0                # exit code
9           li a7, 10               # syscall exit
10          ecall
```

Результат выполнения программы:



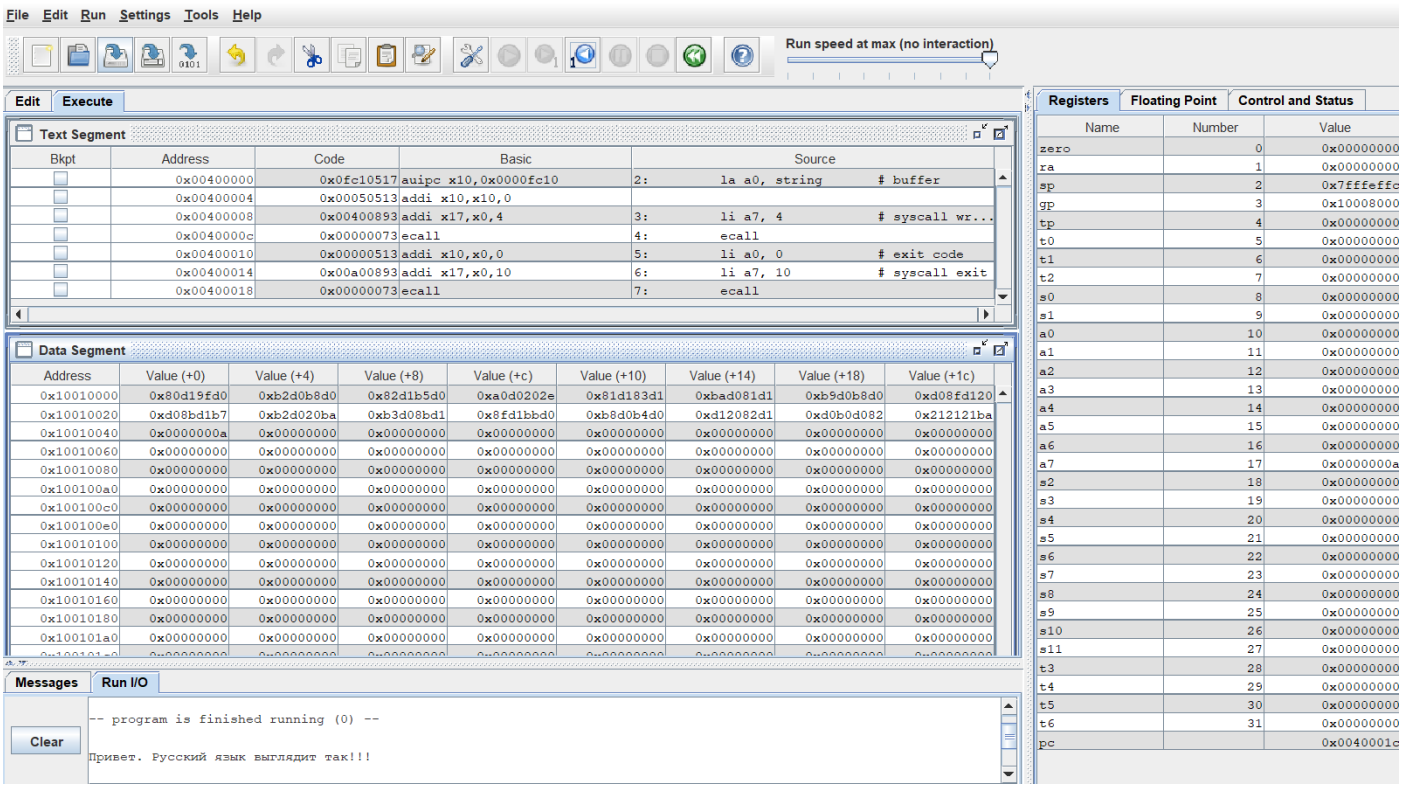
Системные вызовы:

1) Системный вызов №4 (PrintString) – вывод строки в консоль
2) Системный вызов №10 (Exit) – завершение программы с кодом 0

05-hello-ru

```
1  .text
2          la a0, string        # buffer
3          li a7, 4             # syscall write (4)
4          ecall
5          li a0, 0             # exit code
6          li a7, 10            # syscall exit
7          ecall
8  .data
9      string:   .asciz "Привет. Русский язык выглядит так!!!\n"
```

Результат выполнения программы:



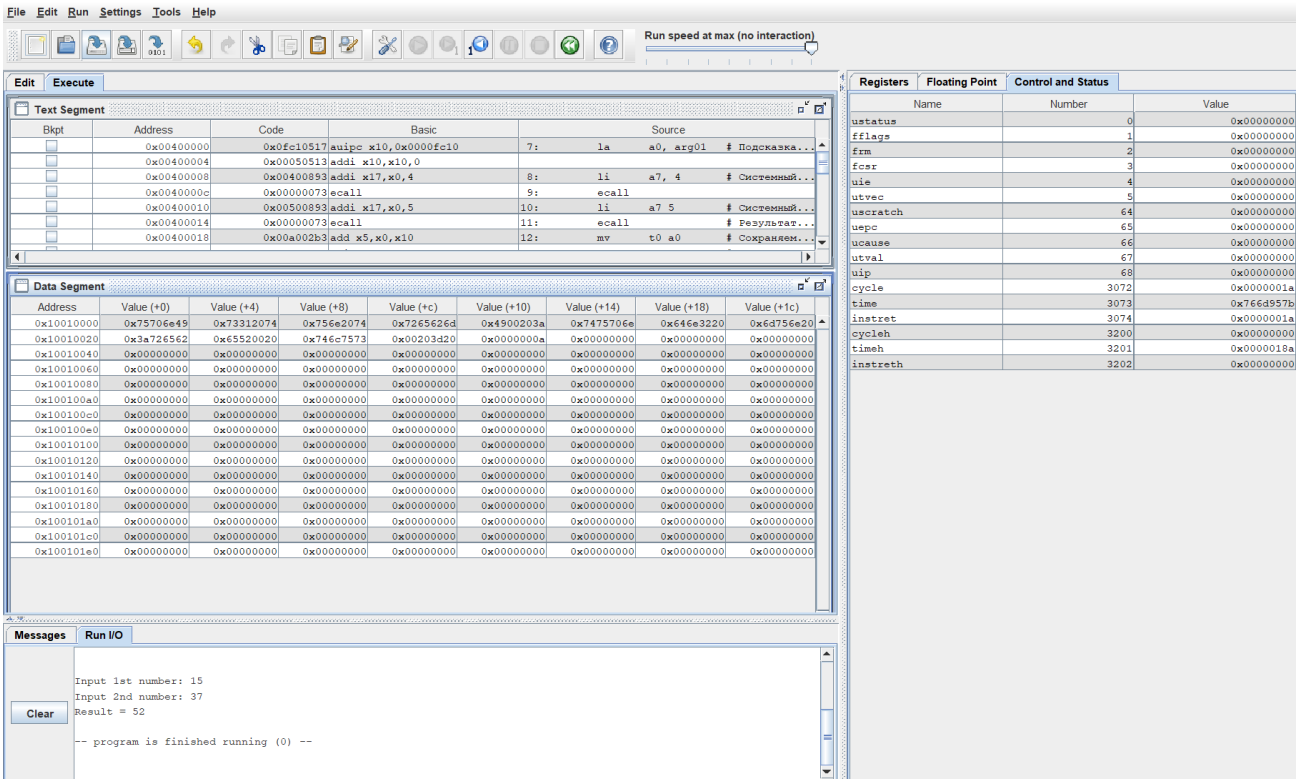
Системные вызовы:

1) Системный вызов №4 (PrintString) – вывод строки в консоль
2) Системный вызов №10 (Exit) – завершение программы с кодом 0

06-add-int02

```
.data
    arg01:   .asciz "Input 1st number: "
    arg02:   .asciz "Input 2nd number: "
    result:  .asciz "Result = "
    ln:      .asciz "\n"
.text
        la      a0, arg01    # Подсказка для ввода первого числа
        li      a7, 4        # Системный вызов №4
        ecall
        li       a7 5         # Системный вызов №5 — ввести десятичное число
        ecall                 # Результат — в регистре a0
        mv       t0 a0        # Сохраняем результат в t0

        la      a0, arg02    # Подсказка для ввода второго числа
        li      a7, 4        # Системный вызов №4
        ecall
        li       a7 5         # Системный вызов №5 — ввести десятичное число
        ecall                 # Результат — в регистре a0
        mv       t1 a0        # Сохраняем результат в t1

        la a0, result         # Подсказка для выводимого результата
        li a7, 4              # Системный вызов №4
        ecall
        add      a0 t0 t1     # Складываем два числа
        li       a7 1         # Системный вызов №1 — вывести десятичное число
        ecall

        la a0, ln             # Перевод строки
        li a7, 4              # Системный вызов №4
        ecall

        li       a7 10        # Системный вызов №10 — останов программы
        ecall
```

Результат выполнения программы:



Системные вызовы:

1) Системный вызов №4 (PrintString) – вывод строки в консоль
2) Системный вызов №5 (ReadInt) – считывание десятичного числа из консоли
3) Системный вызов №1 (PrintInt) – вывод десятичного числа в консоль
4) Системный вызов №10 (Exit) – завершение программы с кодом 0