

VNUHCM – UNIVERSITY OF SCIENCE
FACULTY OF INFORMATION TECHNOLOGY



REPORT

HW2 – MIPS ASSEMBLY PROGRAMMING

COMPUTER SYSTEM

GROUP:

21127636 – Tăng Kim Lân

22127014 – Nguyễn Kim Anh

22127092 – Lê Bảo Giang

22127133 – Đinh Vũ Huân

22127214 – Võ Thị Kim Khôi

22127460 – Quách Trần Quán Vinh

MỤC LỤC

I. Thông tin nhóm	2
II. Ý tưởng (Pseudocode)	2
• Bài 1:	2
• Bài 2:	3
• Bài 3:	4
• Bài 4:	4
III. Test code	5
• Bài 1:	5
• Bài 2:	5
• Bài 3:	6
• Bài 4:	6
IV. Link source code	6

I. Thông tin nhóm

Thành viên:

21127636 – Tăng Kim Lân

22127014 – Nguyễn Kim Anh

22127092 – Lê Bảo Giang

22127133 – Đinh Vũ Huân

22127214 – Võ Thị Kim Khôi

22127460 – Quách Trần Quán Vinh

II. Ý tưởng (Pseudocode)

- **Bài 1:**

```
DATA:
    Định nghĩa các thông điệp cho đầu vào và kết quả

TEXT:
    MAIN:
        Tải giá trị ngay lập tức 4 vào $v0
        Tải địa chỉ của num1_mes vào $a0
        Thực hiện syscall

        Tải giá trị ngay lập tức 5 vào $v0
        Thực hiện syscall
        Di chuyển giá trị từ $v0 đến $t0

        Tải giá trị ngay lập tức 4 vào $v0
        Tải địa chỉ của num2_mes vào $a0
        Thực hiện syscall

        Tải giá trị ngay lập tức 5 vào $v0
        Thực hiện syscall
        Di chuyển giá trị từ $v0 đến $t1

        Nếu $t0 bằng với $t1, nhảy đến printEqual

        Đặt giá trị nhỏ hơn từ $t0 đến $t1 và lưu trong $t2
        Nếu $t2 không bằng với zero, nhảy đến printLess

        Tải giá trị ngay lập tức 4 vào $v0
        Tải địa chỉ của result_mes vào $a0
        Thực hiện syscall

        Tải giá trị ngay lập tức 4 vào $v0
        Tải địa chỉ của case1 vào $a0
        Thực hiện syscall
        Nhảy đến END

    printEqual:
        Tải giá trị ngay lập tức 4 vào $v0
        Tải địa chỉ của result_mes vào $a0
        Thực hiện syscall

        Tải giá trị ngay lập tức 4 vào $v0
        Tải địa chỉ của case3 vào $a0
        Thực hiện syscall
```

```

        Nhảy đến END

printLess:
    Tải giá trị ngay lập tức 4 vào $v0
    Tải địa chỉ của result_mes vào $a0
    Thực hiện syscall

    Tải giá trị ngay lập tức 4 vào $v0
    Tải địa chỉ của case2 vào $a0
    Thực hiện syscall

END:
    Tải giá trị ngay lập tức 10 vào $v0
    Thực hiện syscall

```

• Bài 2:

```

Khởi tạo các biến chuỗi:
prompt = "Nhập: "
result_uppercase = "\nKết quả: chữ hoa"
result_lowercase = "\nKết quả: chữ thường"
invalid = "\nNhập không hợp lệ! Dữ liệu nhập phải là ký tự"

Khởi tạo các phạm vi ASCII:
lowercase_start = 97
lowercase_end = 122
uppercase_start = 65
uppercase_end = 90

nếu (character >= lowercase_start VÀ character <= lowercase_end) thì
    Chuyển đến is_lowercase
nếu khác (character < uppercase_start HOẶC character > uppercase_end) thì
    Chuyển đến error
khác
    Chuyển đến is_uppercase

Nhãn is_uppercase:
    In ra result_uppercase
    Chuyển đến END

Nhãn is_lowercase:
    In ra result_lowercase
    Chuyển đến END

Nhãn error:
    In ra invalid

Nhãn END:
    Kết thúc chương trình

```

- **Bài 3:**

```
Khai báo các dữ liệu
    Arr ← vùng nhớ 40 bytes
main:
    Đọc số nguyên nhập từ input vào $t0 bằng syscall
    Branch if $t0 != 0:
        đi đến END
    Branch if $t0 < 0:
        đi đến Error
    Branch if $t0 > 10:
        đi đến error
    Khởi tạo $t1 ← 0, $s0 ← 0

Input:
    Branch if $t1 == $t0: đi đến Next
    Đọc 1 số nguyên từ input vào Arr[$s0] bằng syscall
    $s0 ← $s0 + 4
    $t1 ← $t + 1
    Lặp lại Input

Next:
    Gán $s0 ← 0

Output:
    Branch if $t0 == 0, đi đến END
    Tải số nguyên tại vùng nhớ Arr[$s0] vào $t1
    In ra giá trị $t1 bằng syscall
    $s0 ← $s0 + 4
    $t0 ← $t0 - 1
    Lặp lại Output

Error:
    Output báo lỗi bằng syscall
END:
    Kết thúc chương trình
```

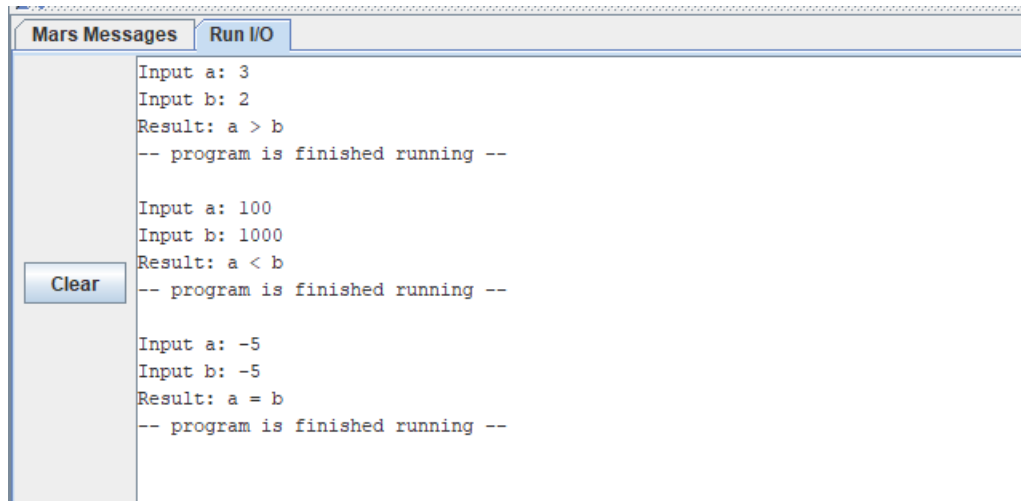
- **Bài 4:**

```
Khai báo các dữ liệu
    str1 ← vùng nhớ 20 bytes
main:
    Đọc string từ input vào str1 bằng syscall
    Gán $a0 trỏ vào str1
    Khởi gán $t2 ← 0
Loop:
    Lấy dữ liệu được trỏ bởi $a0 vào $t0
    Branch if $t0 == NULL: đi đến END
    Tăng địa chỉ $a0 thêm 1
    $t2 ← $t2 + 1
    Lặp lại Loop

END:
    $t2 ← $t2 - 1 #Loại bỏ NULL
    Đưa giá trị từ $t2 vào $a0
    In ra độ dài chuỗi trong $a0 bằng syscall
    Kết thúc chương trình
```

III. Test code

- **Bài 1:**



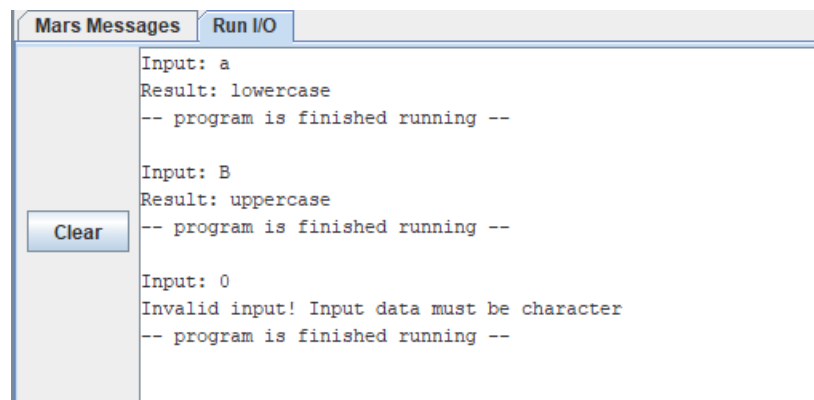
The screenshot shows a window titled "Mars Messages" with a "Run I/O" button. The window contains three sets of test results. Each set starts with "Input a:" and "Input b:" values, followed by a "Result:" line, and ends with "-- program is finished running --". A "Clear" button is located on the left side of the window.

```
Input a: 3
Input b: 2
Result: a > b
-- program is finished running --

Input a: 100
Input b: 1000
Result: a < b
-- program is finished running --

Input a: -5
Input b: -5
Result: a = b
-- program is finished running --
```

- **Bài 2:**



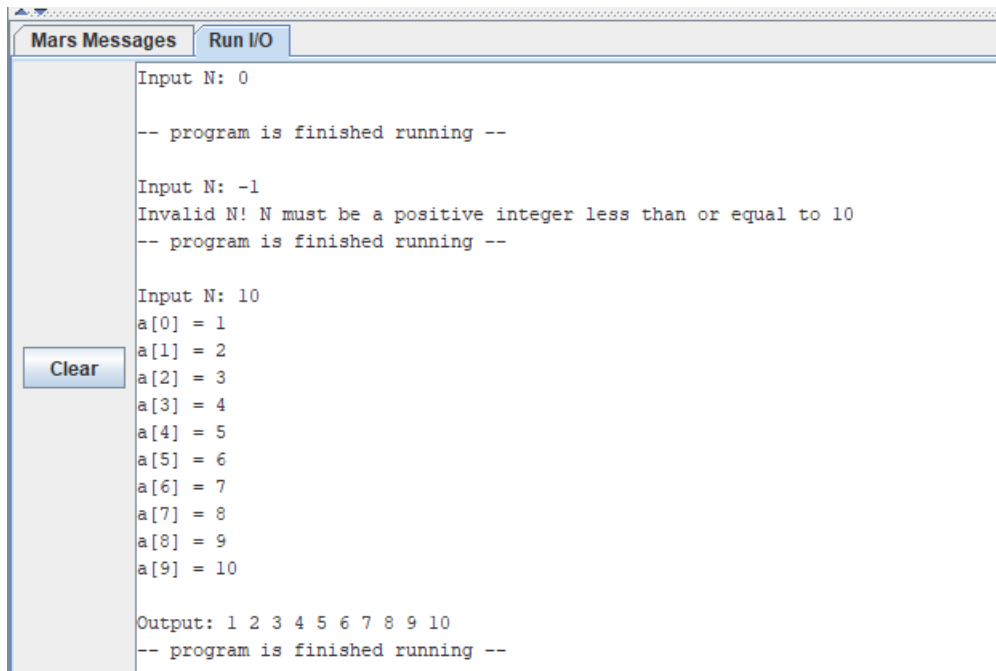
The screenshot shows a window titled "Mars Messages" with a "Run I/O" button. The window contains three sets of test results. Each set starts with "Input:" and "Result:" values, followed by "-- program is finished running --". A "Clear" button is located on the left side of the window.

```
Input: a
Result: lowercase
-- program is finished running --

Input: B
Result: uppercase
-- program is finished running --

Input: 0
Invalid input! Input data must be character
-- program is finished running --
```

- **Bài 3:**



The screenshot shows a window titled "Mars Messages" with a "Run I/O" button. The text area contains the following output:

```
Input N: 0
-- program is finished running --

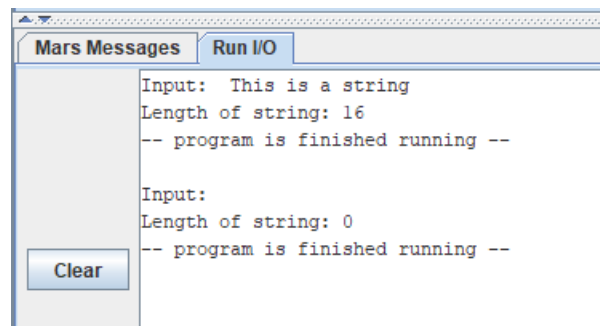
Input N: -1
Invalid N! N must be a positive integer less than or equal to 10
-- program is finished running --

Input N: 10
a[0] = 1
a[1] = 2
a[2] = 3
a[3] = 4
a[4] = 5
a[5] = 6
a[6] = 7
a[7] = 8
a[8] = 9
a[9] = 10

Output: 1 2 3 4 5 6 7 8 9 10
-- program is finished running --
```

A "Clear" button is visible on the left side of the text area.

- **Bài 4:**



The screenshot shows a window titled "Mars Messages" with a "Run I/O" button. The text area contains the following output:

```
Input: This is a string
Length of string: 16
-- program is finished running --

Input:
Length of string: 0
-- program is finished running --
```

A "Clear" button is visible on the left side of the text area.

IV. Link source code

https://drive.google.com/drive/folders/119tSI2b1pYuiQzUdd0tzoJFzNHtsZ_qh?usp=sharing