**BÁO CÁO THỰC HÀNH KIẾN TRÚC MÁY TÍNH**

**LAB 5**

Họ và tên: Phạm Vân Anh

MSSV: 20214988

Mã lớp: 139365

***ASSIGNMENT 1:***

1. **Code**

**Graphical user interface, text, application, email

Description automatically generated**

*Thực hiện gõ chương trình vào công cụ MARS*

1. **Thực hiện dòng lệnh số 6:**

* Gán giá trị $v0 = 4 🡪 option của SYSCALL là PRINT STRING

****

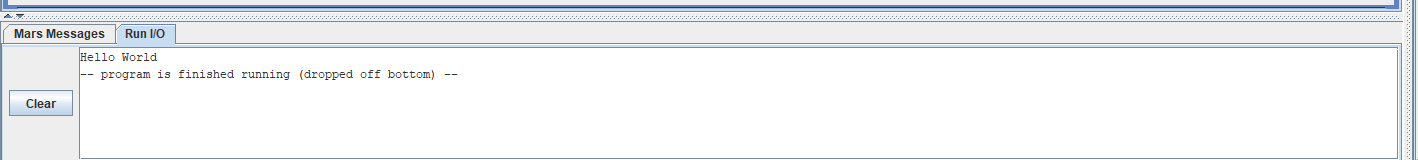
1. **Thực hiện dòng lệnh số 7:**

* Gán $a0 là giá trị địa chỉ của *test:*

**A picture containing graphical user interface

Description automatically generated**

1. **In chuỗi**

****

***ASSIGNMENT 2:***

1. **Code**

Text

Description automatically generated

*Thực hiện gõ chương trình vào công cụ MARS*

1. **Thực hiện câu lệnh số 8 và 9**

* Gán giá trị của $s0 = 10, $s1 = 12

****

1. **Thực hiện câu lệnh số 12, 13, 14: In ra chuỗi “The sum of”**

* Câu lệnh 12: Gán giá trị $v0 = 4 🡪 option của SYSCALL là *PRINT STRING*
* Gán địa chỉ của Message1 vào $a0

**A picture containing rectangle

Description automatically generated**

1. **Thực hiện câu lệnh số 16, 17, 18: In ra “s0”**

* Câu lệnh 16: Gán giá trị của $v0 = 1 🡪 option của SYSCALL là *RINT DECIMAL INTEGER*
* Câu lệnh số 17: a0 = 0 + s0 = s0

**A picture containing icon

Description automatically generated**

1. **Thực hiện câu lệnh số 20, 21, 22: In ra “and”**

* Câu lệnh 20: Gán giá trị $v0 = 4 🡪 option của SYSCALL là *PRINT STRING*
* Gán địa chỉ của Message2 vào $a0

**A picture containing background pattern

Description automatically generated**

1. **Thực hiện câu lệnh số 24, 25, 26: In ra “$s1”**

* Câu lệnh 24: Gán giá trị của $v0 = 1 🡪 option của SYSCALL là *RINT DECIMAL INTEGER*
* Câu lệnh số 25: a0 = 0 + s1 = s1

****

1. **Thực hiện câu lệnh số 28, 29, 30: In ra “is”**

* Câu lệnh 28: Gán giá trị $v0 = 4 🡪 option của SYSCALL là *PRINT STRING*
* Gán địa chỉ của Message3 vào $a0

**A picture containing rectangle

Description automatically generated**

1. **Thực hiện câu lệnh số 32, 33, 34: In ra “result”**

* Câu lệnh 32: Gán giá trị của $v0 = 1 🡪 option của SYSCALL là *RINT DECIMAL INTEGER*
* Câu lệnh số 33: a0 = s0 + s1

**A picture containing background pattern

Description automatically generated**

1. **In ra kết quả**

* $a0 = $s0 + $s1 = 10 + 12 = 22

**Graphical user interface, text, application, email

Description automatically generated**

***ASSIGNMENT 3:***

* + - 1. **Code**

***Text

Description automatically generated with medium confidence***

*Thực hiện gõ chương trình vào công cụ MARS*

* + - 1. **Thực hiện câu lệnh số 9, 10, 11**
* Câu lệnh số 9: Gán $s0 = 0

****

* Câu lệnh số 10: Gán địa chỉ của x vào $a0

**A picture containing rectangle

Description automatically generated**

* Câu lệnh số 11: Gán địa chỉ của y vào $a1

****

* + - 1. **Thực hiện câu lệnh số 13 🡪 23**
* Câu lệnh số 13: $t1 = $s0 + $a1 = i + y[0] 🡪 $t1 mang địa chỉ của y[i]

****

* Câu lệnh số 15: $t2 = x[i] = y[i] 🡪 lấy giá trị của $t1 lưu vào thanh ghi $t2

****

* Câu lệnh số 16: $t3 = $s0 + $a0 = i + x[0] 🡪 $t3 mang địa chỉ của x[i]

****

* Câu lệnh số 18: $t2 = x[i] = y[i] 🡪 lấy giá trị của $t2 lưu vào thanh ghi $t3

****

* Câu lệnh số 19: so sánh $t2 và 0
* $t2 = 0 🡪 end
* $t2 != 0 🡪 chạy tiếp
* Câu lệnh số 21: $s0 = $s0 + 1 (i=i+1) (tăng giá trị của $s0 )

****

* Câu lệnh số 22: Vòng lặp L1 tới khi hết chuỗi
  + - 1. **Thực hiện câu lệnh số 25, 26, 27**
* Câu lệnh 25: Gán giá trị $v0 = 4 🡪 option của SYSCALL là *PRINT STRING*

****

* Câu lệnh 26: Gán địa chỉ của x vào $t1

****

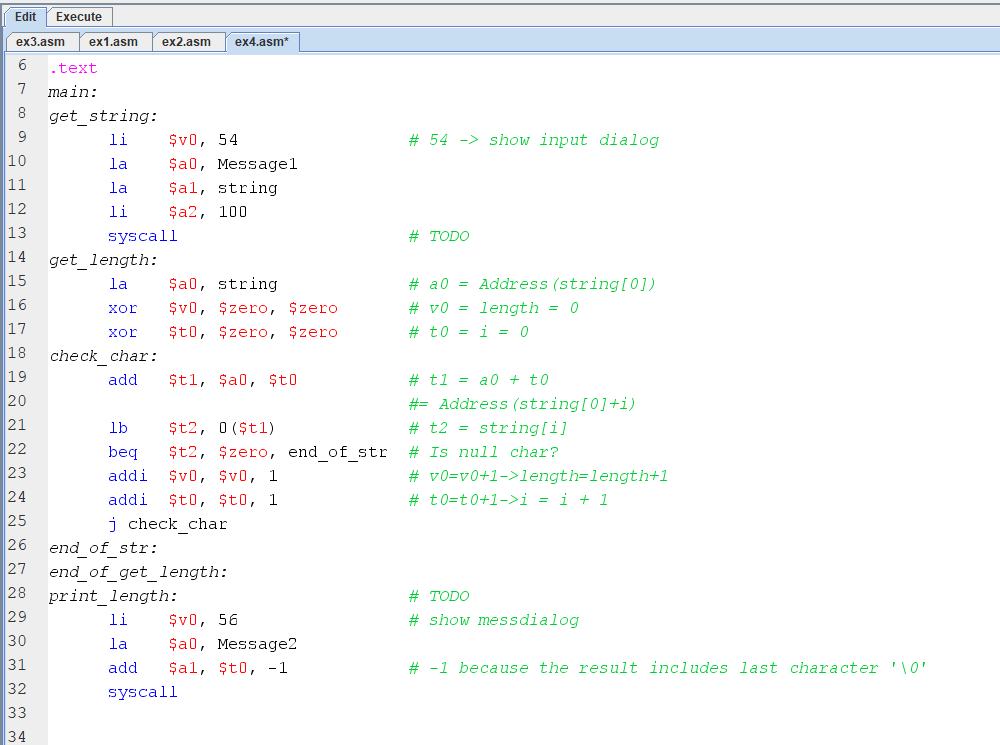
* In ra:

**Graphical user interface, application

Description automatically generated**

***ASSIGNMENT 4:***

**Code**



*Thực hiện gõ chương trình vào công cụ MARS*

**Thực hiện câu lệnh số 10, 11, 12, 13, 14: Nhập xâu**

* Câu lệnh số 10: Gán giá trị $v0 = 54 🡪 option của SYSCALL là *InputDialogString*

****

* Câu lệnh số 11: Gán địa chỉ của Message1 vào $a0

****

* Câu lệnh số 12: Gán địa chỉ của String vào $a1

****

* Câu lệnh số 13: $a2 là max\_string 🡪 $a2 = 100

****

* Kết quả:

***Graphical user interface, application, Word

Description automatically generated***

**Table

Description automatically generated**

**Thực hiện câu lệnh số 16, 17, 18**

* Câu lệnh số 16: Gán địa chỉ của *string* vào $a0

****

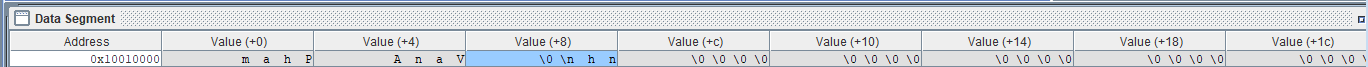
* Câu lệnh số 17: Gán giá trị $v0 = 0

****

* Câu lệnh số 18: Gán giá trị của $t0 = 0

****

* Kết quả:

****

**Thực hiện câu lệnh số 20 🡪 26: Check char**

* Câu lệnh số 20: $t1 = $a0 + $t0 (Address(string[0]+i))

****

* Câu lệnh số 22: Lấy giá trị từ địa chỉ $t1 vào $t2

****

* Câu lệnh số 23: So sánh t2 và 0
* Nếu t2 = 0 🡪 end
* Nếu t2 != 0 🡪 chạy tiếp
* Câu lệnh số 24: Tăng giá trị của v0 lên v0 + 1 (v0 là độ dài chuỗi)

****

* Câu lệnh số 25: Tăng giá trị của t0 lên t0 + 1 (t0: bước nhảy index)

****

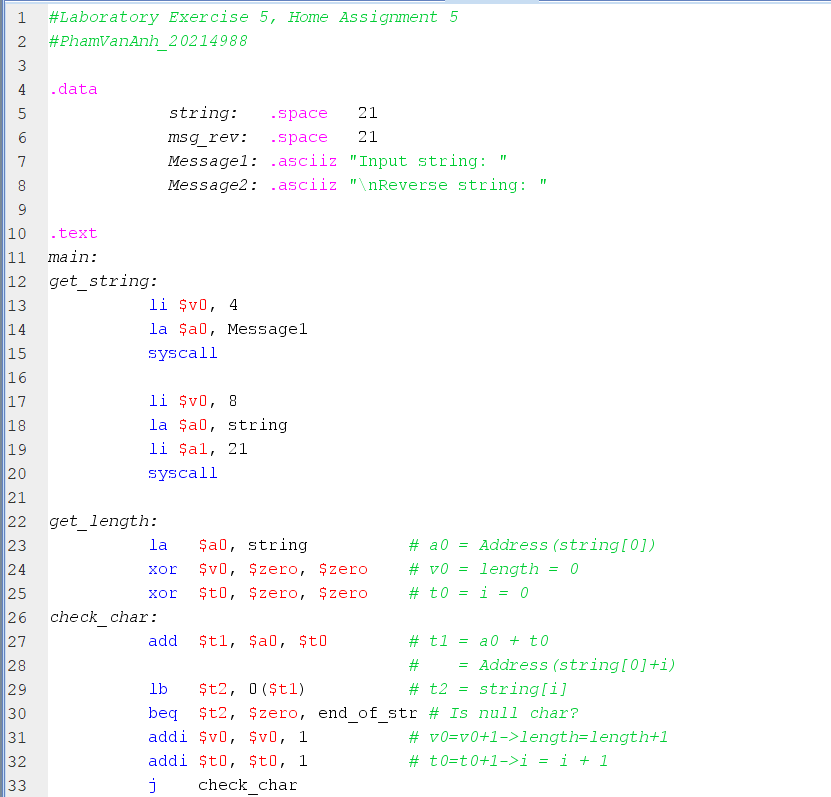
* Câu lệnh số 26: quay lại vòng lặp chevk\_char tới khi nào hết chuỗi
  + - 1. **Thực hiện câu lệnh số 27: end\_of\_str: dừng chuỗi**
      2. **Thực hiện câu lệnh số 28: end\_of\_get\_length: dừng lấy độ dài chuỗi**
      3. **Thực hiện câu lệnh số 29 🡪 33: In ra độ dài chuỗi**
* Thực hiện câu lệnh số 30: Gán giá trị $v0 = 6 🡪 option của SYSCALL là *MessageDialogInt*
* Thực hiện câu lệnh số 31: Gán địa chỉ của Message2 vào $a0
* Thực hiện câu lệnh số 32: Phải -1 của $a1 vì trừ đi kí tự cuối cùng là \0
  + - 1. **Kết quả khi thử chuỗi “PhamVanAnh” 🡪 kết quả đúng**

**Graphical user interface, application, Word

Description automatically generated**

***ASSIGNMENT 5:***

Code



Text

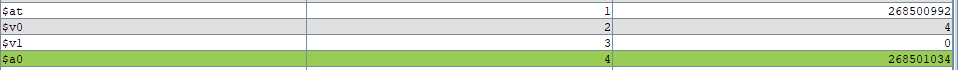
Description automatically generated

**Thực hiện câu lệnh 12 🡪 20: get\_string**

1. Câu lệnh số 13: Gán giá trị $v0 = 4 🡪 option của SYSCALL là *PRINT STRING*



1. Câu lệnh số 14: gán địa chỉ của Message1 vào a0



1. Câu lệnh số 17: gán v0 = 8 🡪 option của SYSCALL là *READ STRING*



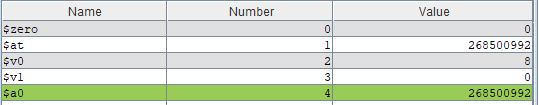
1. Câu lệnh số 18: gán địa chỉ của string vào a0

A picture containing background pattern

Description automatically generated

**Thực hiện câu lệnh 22 🡪 25: get\_length**

1. Câu lệnh số 23: gán địa chỉ của string vào a0



1. Câu lệnh số 24: v0 =0



1. Câu lệnh số 25: t0 = 0



**Thực hiện câu lệnh 27 🡪 33: check\_char**

1. Câu lệnh số 27: $t1 = $a0 + $t0 (Address(string[0]+i))

****

1. Câu lệnh số 29: Lấy giá trị từ địa chỉ $t1 vào $t2

****

* Câu lệnh số 30: So sánh t2 và 0
* Nếu t2 = 0 🡪 end
* Nếu t2 != 0 🡪 chạy tiếp
* Câu lệnh số 31: Tăng giá trị của v0 lên v0 + 1 (v0 là độ dài chuỗi)

****

* Câu lệnh số 32: Tăng giá trị của t0 lên t0 + 1 (t0: bước nhảy index)

****

* Câu lệnh số 33: quay lại vòng lặp check\_char tới khi nào hết chuỗi

**Thực hiện câu lệnh 37 🡪 43**

* Câu lệnh số 38: t4 = v0 – 1

****

* Câu lệnh số 39: s0 = a0 + t4

****

* Câu lệnh số 40: Lấy giá trị từ thanh ghi s0 sang thanh ghi t5

****

* Câu lệnh số 41: t8 = \n
* Câu lệnh số 42: so sánh t5 và t8

+ t5 = t8 🡪 add v0 = v0 – 1

+ t5 != t8 🡪 reverse\_string

**Thực hiện câu lệnh 45 🡪 48: reverse\_string**

* Câu lệnh số 46: gán địa chỉ của msg\_rev vào a1

**A picture containing background pattern

Description automatically generated**

* Câu lệnh số 47: v0 = v0 – 1

****

* Câu lệnh số 48: v1 = 0

**Thực hiện câu lệnh 49 🡪 58**

* Câu lệnh số 50: t3 = v0 + 1

****

* Câu lệnh số 51: So sánh t3 và 0

+ t3 = 0 🡪 end

+ t3 != 0

* Câu lệnh số 52: t1 = a0 + v0



* Câu lệnh số 53: Lấy giá trị từ thanh ghi t1 sang thanh ghi t2



* Câu lệnh số 54: t3 = a1 + v1



* Câu lệnh số 55: lấy giá trị của t3 lưu vào t2



* Câu lệnh số 56: v1 = v1 + 1



* Câu lệnh số 57: v0 = v0 – 1



* Câu lệnh số 58: vòng lặp *rev* tới hết chuỗi
  + - 1. **Thực hiện câu lệnh 61 🡪 67: In chuỗi đảo**
* Câu lệnh số 61: v0 = 4

****

* Câu lệnh số 62: Gán địa chỉ của Message2 và a0

****

* Câu lệnh số 65: v0 = 4

****

* Câu lệnh số 66: Gán địa chỉ của reverse\_String vào $a0

****

* + - 1. **Kết quả:**
* Thử với chuỗi “PhamVanAnh”
* Kết quả:

**Graphical user interface, text, application, email

Description automatically generated**