**Báo cáo kiến trúc máy tính**

*Người thực hiện:*

1. Nguyễn Thị Thu Uyên (20176904)

2.Phạm Trọng Huấn (20176771)

Mục lục

[**Phần 1:** Báo cáo Mini-Project-9](#_Toc40605766)

[I. Đề bài](#_Toc40605767)

[II. Code](#_Toc40605768)

[III. Phân tích cách thực hiện](#_Toc40605769)

[IV. Ý nghĩa các thanh ghi](#_Toc40605769)

V. Ý nghĩa chương trình con

[**Phần 2:** Báo cáo Mini-Project-10](#_Toc40605770)

[I. Đề bài](#_Toc40605771)

[II. Code](#_Toc40605772)

[III. Phân tích](#_Toc40605773)

# **Phần 1:** Báo cáo Mini-Project-9

**I.Đề bài:** Viết chương trình:

* Đọc vào số lượng học sinh trong lớp.
* Đọc thông tin về từng học sinh, bao gồm: Tên, điểm Toán.
* Liệt kê tên của tất cả học sinh đã **trượt** bài kiểm tra Toán.

**II.Code**

.data

SV: .space 200

MAX\_OF\_SV: .word 10

Message : .asciiz "Danh sach sinh vien truot la: \n\n"

Message1: .asciiz "so luong SV phai > 0"

Number\_Of\_SV: .asciiz "Nhap so luong sinh vien MAX = 10"

Name\_Of\_SV : .asciiz "Nhap ten SV: "

Check\_Mark: .asciiz "diem >0 and diem <10"

Diem: .asciiz "Nhap diem <= 10"

.text

Main:

jal Nhap

nop

jal Print

nop

end\_Main:

li $v0, 10

syscall

Nhap: #Ham nhap so luong

lw $s1,MAX\_OF\_SV #max of sinh vien doc du lieu word tu bo nho vao thanh ghi s1

#---------------nhap n( Nhap so luong sinh vien )------------------------------

li $v0,51# goi hop thoai nhap vao so luong sinh

la $a0,Number\_Of\_SV #$a0 tieu de: nhap vao oso luong sinh vien Max=10

syscall#$a0 Nhap vao so luong sinh vien

slt $t0,$s1,$a0 # so sánh $s1 < $a0 ? neu dung thi $t0=1( so sanh so sinh vien max < so sinh vien nhap vao ko ??)

bne $t0,$0,Main #neu($t0!=0) thì nhay lai lenh main (neu < thi lap lai ham main)

slt $t0,$0,$a0# con neu ($t0!=0) sai t?c là $a0<$s1 thì thuc hien so sanh tiep $0<$a0??( so sanh so sinh vien nhap vao>0 ko ??)

bne $t0,$0,Continue #neu ($t0!=$0)hay ($0<$a0) dung thi nhay lenh Countinue( neu so sinh vien nhap vao >0 thi nhay sang lenh Continue)

li $v0,55# neu 0<$a0 goi hop thong bao

la $a0, Message1# $a0 tieu de :"so luong SV phai > 0"

syscall

j Nhap

Continue:

#khoi tao gia tri các giá tri

add $s0,$a0,$0 # $ gan s0 = n( =a0= so luong sv)

la $s1,SV # nap dia chi cua mang s1

add $t1,$0,$s0 # Khoi tao i = n

addi $s3,$s1,0 #Khoi tao s3=sv[0]

NhapMang:

#nhap ten

beq $t1,0,Done #neu $t1==0 thi nhay sang Done

li $v0,54 # con neu ko thi goi hop thoai nhap ten sinh vien

la $a0,Name\_Of\_SV# tieu de Name\_Of\_SV

la $a1,0($s3)# chi ra vi tri luu ten

la $a2,16# gioi han do dai la 16 ki tu

syscall# nhap ten

addi $s3,$s3,16 # chuyen sang vi tri tiep theo

Mark:

#nhap diem

li $v0,51# goi hop thoai

la $a0,Diem# tieu de : Diem

syscall# Nhap diem

slti $t0,$a0,11# so sánh diem nhap vao a0<11 dung thi t0=1

beq $t0,$0,Mark# neu t0==0 thì nhay lai lenh Mark( nghia la khi nay a0>10)

slt $t0,$0,$a0# con neu ko thi so sánh 0<a0,neu dung t0=1

bne $t0,$0,Continue1#neu t0!=0 thì nhay sang lenh Continue1

li $v0,55 # neu ko thi goi hop thong bao

la $a0, Check\_Mark# tieu de:"diem >0 and diem <10"

syscall# nhan ok

j Mark #va nhay lai lenh main

Continue1:

#l?u thong tin vua nhap và nhay sang phan tu tiep theo cua mang

sw $a0,0($s3)# dua du lieu tu a0 ra dia chi co so cua s3

addi $s3, $s3, 4# nhay sang phan tu tiep theo cua s3

addi $t1,$t1,-1# Khoi tao lai t1=t1-1

j NhapMang# nhay lai lenh NhapMang

Done:

# ket thuc viec nhap

li $v0,4# thong bao ket qua ra console

la $a0,Message # tieu de: "Danh sach sinh vien truot la \n"

syscall

jr $ra

Print:

#B?t ??u in

addi $t3, $0,0 # i = 0

add $s4,$0,$s1 # dia chi sv[0]

for:

#vong lap nap diem va check diem

add $s5,$s4, $0 #s5=s4

addi $t5,$s4,16 #sv[0]+16

lw $t6,0($t5) #nap sv[0].diem

slti $t7,$t6,5# ?diem check là 5, so sánh diem nhap vao voi 5, neu < thì t7=1

bne $t7,$0,print\_char # neu t7!=0(hay t7=1) thì nhay sang lenh print\_char

j next# còn neu không thì nhay sang lenh next

print\_char: #in ten

lb $t2,0($s5) # load byte

beq $t2,10, end\_for # = \n. neu t2==10 thì nhay lenh end\_for

addi $a0,$t2,0 # neu khong thi a0 = t2

li $v0, 11 # print char

syscall

addi $s5, $s5, 1 # j++

j print\_char

end\_for:

#ket thuc vong lap

li $v0,11# in char

li $a0,10# thoat chuong trinh

syscall

next:

#chuyen toi vi tri tiep theo

addi $t3, $t3,1 #sv[i++]

beq $t3, $s0, end\_print #i = n? thi nhay sang lenh end\_print

mul $t4,$t3,20 #i\*20

add $s4,$s1,$t4 # dia chi sv[i]

j for# nhay sang lenh for

end\_print:

jr $ra

**III.Phân tích cách thực hiện:**

* Phân tích đề bài:
* Input: Số sinh viên và danh sách sinh viên.
* Output: Danh sách sinh viên bị trượt bài thi toán
* Ý tưởng thực hiện:
* Nhập số lượng sinh viên ( số nguyên dương)
* Nhập lần lượt tên và điểm sinh viên với điều kiện: 0 <= điểm <= 10
* So sánh lần lượt từng điểm sinh viên với 5, bé hơn thì trượt
* Các bước thực hiện:
* Bước 1: Nhập n( số lượng sinh viên ) có max =10

Nếu N không phải số tự nhiên < = max và >0thì nhập lại.

* Bước 2: Nhập tên và điểm sinh viên:

Nếu điểm nhập vào không thỏa mãn: 0<= điểm <=10 thì nhập lại.

* Bước 3: Kiểm tra điểm và đưa ra danh sách thí sinh trượt, nếu điểm bé hơn 5 là trượt

**IV.Ý nghĩa của các thanh ghi được sử dụng:**

* $s1 : Lưu số lượng sinh viên MAX, sau đó thì được gán là địa chỉ cơ sở mảng sinh viên
* $t1: các giá trị tạm thời
* $s3: lưu địa chỉ cơ sở của mảng
* $a0 : số sinh viên
* $a1 : chỉ ra vị trí lưu tên.
* $a2: giới hạn tên
* $t1: biến i
* $s3: Lưu dánh sách sinh viên
* $t0 biến gán tạm thời

**V.Ý nghĩa của các chương trình con nếu có:**

* count: Nhập số sinh viên.
* NhapMang: nhập tên sinh viên với giới hạn là 16 kí tự.
* Mark: nhập điểm sinh viên với điều kiện 0 <= điểm <=10.
* Continue: Khởi tạo các giá trị ban đầu
* Countinue1: lưu thông tin vừa nhập và nhảy sang địa chỉ tiếp theo của mảng
* Done : In ra dánh sách trượt
* Print\_name : khởi tạo i=0, s4 là địa chủ cơ sở SV[0]
* For: vòng lặp nạp điểm và check điểm
* Print\_char: in tên
* End\_for: kết thúc vòng lặp
* Next: in ra các sinh viên tiếp theo

# **Phần 2:** Báo cáo Mini-Project-10

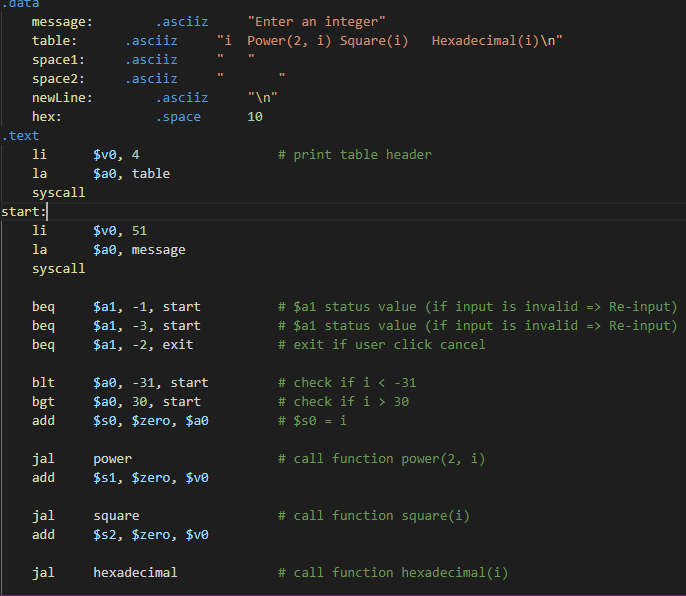
**1.Đề bài**

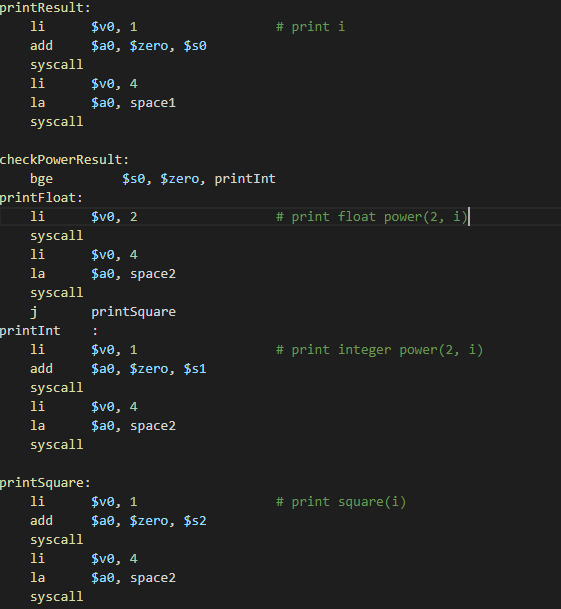
Write a program that gets an integer i from the user and creates the table shown below on the screen (example inputs provided). Subroutines are required for power, square, and hexadecimal (in 32-bit arithmetic, attend to overflowed results). Hint: Hexadecimal can be done with shifts and masks because the size is 32 bits. i power(2,i) square(i) Hexadecimal(i) 10 1024 100 0xA

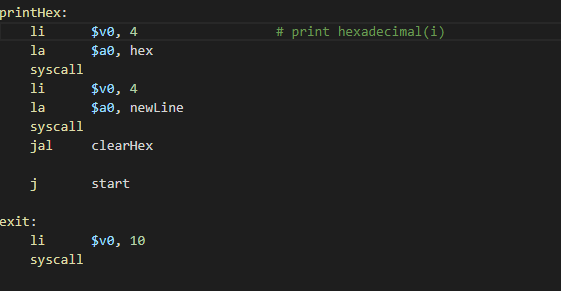
7 128 49 0x7

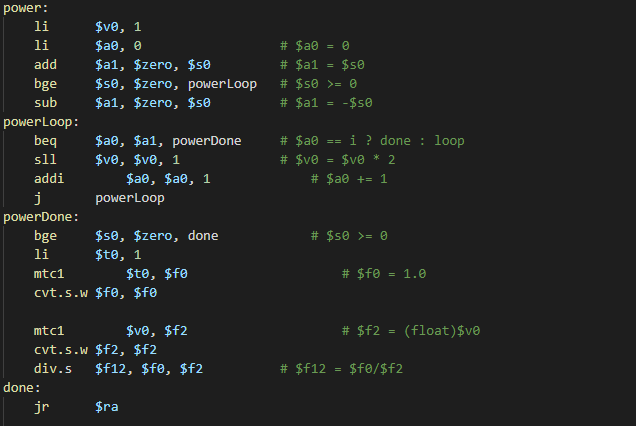
16 65536 256 0x10

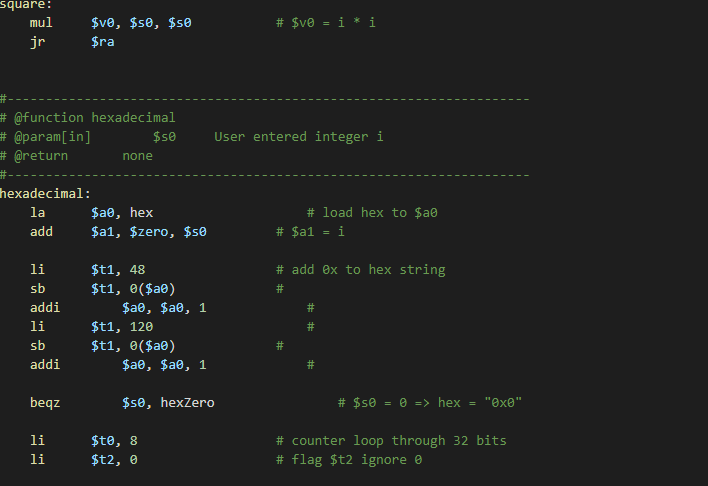
**2. Code**

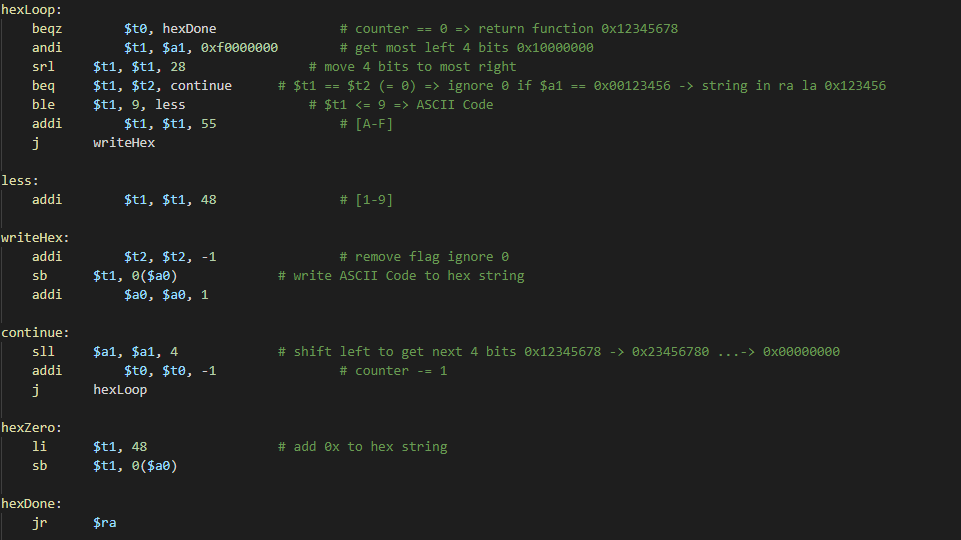


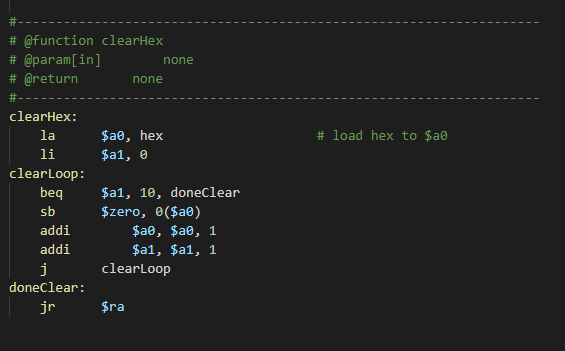












**3. Phân tích**

Hexa:

Các thanh ghi sử dụng

Thanh ghi Mục đích

$a0 Lưu địa chỉ biến Hex

$a1 Lưu giá trị i

$t0 Lưu giá trị couter

$t1 Thanh ghi tạm, lưu giá trị từng byte

$t2 Cờ để xác định in số 0 hay không

Giải thích

Biến Hex khiểu .space dùng để lưu xâu biểu diễn số hexa chuyển đổi từ i.

Nạp địa chỉ của biến hex vào $a0

Ghi xâu “0x” vào biến hex. Nếu i= 0 trả về hex:”0x0”

Khởi tạo biến counter $t0 =8 , biến $t2 =0 làm cờ:

Khi $t2 =0 => không ghi “0” vào xâu

Thực hiện vòng lặp :

Lấy 4bits ngoài cùng trái của i bằng phép AND lưu vào thanh ghi $t1

Kiểm tra giá trị $t1:

Nếu $t1 <= 9 => $t1 += 48 để được ASII của [0-9]

Nếu $t1 > 9 => $t1 +=55 để được ASII của [A-F]

Ghi kí tự vừa lấy được vào xâu hex.

Dịch trái thanh ghi $a1 4bits để lấy 4 bits trái ngoài cùng tiếp theo của i

Vòng lặp kết thúc khi lặp đủ 8 lần .

Biến hex đã chứa xâu biểu diễn số hexa chuyển từ i