# Báo cáo Thực hành KTMT buổi 11

**Họ và tên:** Nguyễn Đức Phú

**MSSV:** 20215116

## Assignment 1:

1. Vẽ hình tam giác đều:
   - Code:

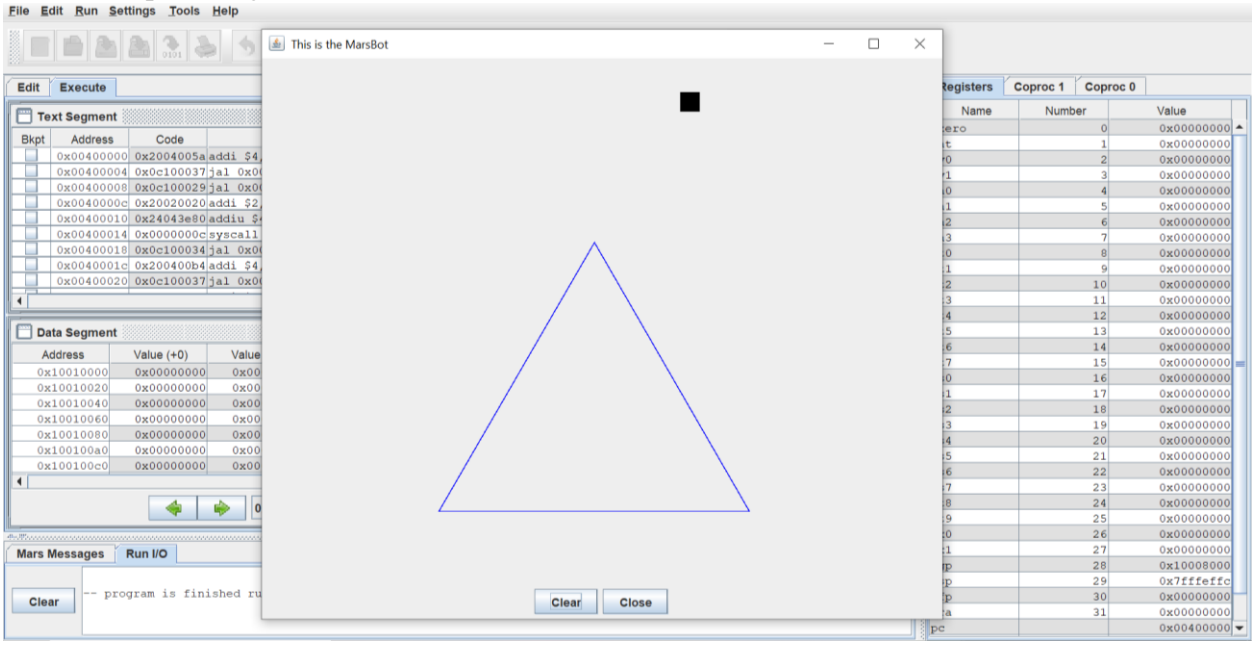
```
.eqv  HEADING    0xffff8010
.eqv  MOVING     0xffff8050
.eqv  LEAVETRACK 0xffff8020
.eqv  WHEREX     0xffff8030
.eqv  WHEREY     0xffff8040
.text
main:
  addi     $a0, $zero, 90
  jal      ROTATE
  jal      GO
sleep1:
  addi     $v0,$zero,32
  li       $a0,16000
  syscall
  jal      UNTRACK
goDOWN:
  addi$a0, $zero, 180
  jal      ROTATE
sleep2:
  addi     $v0,$zero,32
  li       $a0,9000
  syscall
  jal      UNTRACK
  jal      TRACK

go120:
  addi     $a0, $zero, 150
  jal      ROTATE
sleep3:
```

```
  addi     $v0,$zero,32
  li       $a0,15000
  syscall
  jal      UNTRACK
  jal      TRACK
goLEFT:
  addi     $a0, $zero, 270
  jal      ROTATE
sleep4:
  addi     $v0,$zero,32
  li       $a0,15000
  syscall
  jal UNTRACK
  jal TRACK
go30:
  addi$a0, $zero, 30
  jal ROTATE
sleep5:
  addi$v0,$zero,32
  li  $a0,15000
  syscall
  jal UNTRACK
goLEFT1:
  addi$a0,$zero,270
  li  $a0,10000
  syscall
  jal       STOP
end_main:
  li $v0,10
  syscall
#-----------------------------------------------
-----------
# GO procedure, to start running
# param[in]    none
#-----------------------------------------------
-----------
GO:        li       $at, MOVING
```

```
  addi     $k0, $zero,1     # to  logic 1,
  sb       $k0, 0($at)      # to start running
  jr       $ra
#-----------------------------------------------
-----------
# STOP procedure, to stop running
# param[in]    none
#-----------------------------------------------
-----------
STOP:      li       $at, MOVING
  sb       $zero, 0($at)    # to stop
  jr       $ra
#-----------------------------------------------
-----------
# TRACK procedure, to start drawing line
# param[in]    none
#-----------------------------------------------
-----------
TRACK:     li       $at, LEAVETRACK
           addi     $k0, $zero,1
           sb       $k0, 0($at)
           jr       $ra
#-----------------------------------------------
-----------
# UNTRACK procedure, to stop drawing line
# param[in]    none
#-----------------------------------------------
-----------
UNTRACK:   li       $at, LEAVETRACK
           sb       $zero, 0($at)
           jr       $ra
#-----------------------------------------------
-----------
# ROTATE procedure, to rotate the robot
# param[in]    $a0, An angle between 0 and 359
#                   0 : North (up)
# 90: East (right)
```

```
# 180: South (down)
# 270: West (left)
#-----------------------------------------------
-----------
ROTATE:    li   $at, HEADING
           sw   $a0, 0($at)
           jr   $ra
```

- Kết quả chạy:




2. Vẽ hình vuông:
   - Code:

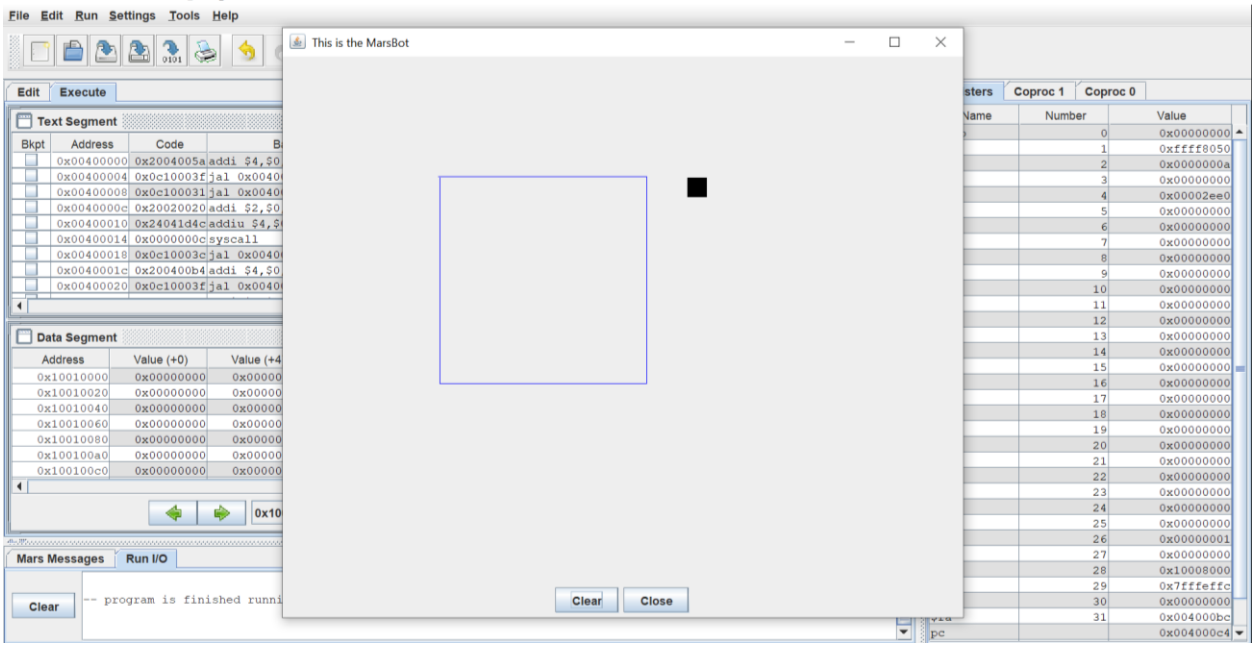
```
.eqv  HEADING    0xffff8010
.eqv  MOVING     0xffff8050
.eqv  LEAVETRACK 0xffff8020
.eqv  WHEREX     0xffff8030
.eqv  WHEREY     0xffff8040
.text
main:
  addi     $a0, $zero, 90
  jal      ROTATE
  jal      GO
sleep1:    addi     $v0,$zero,32
```

```
  li       $a0,7500
  syscall
  jal      UNTRACK
goDOWN:    addi$a0, $zero, 180
  jal      ROTATE
sleep2:    addi     $v0,$zero,32
  li       $a0,6000
  syscall
  jal      UNTRACK
  jal      TRACK

goRIGHT:   addi     $a0, $zero, 90
  jal      ROTATE
sleep3:    addi     $v0,$zero,32
  li       $a0,10000
  syscall
  jal      UNTRACK
  jal      TRACK
goDOWN1:   addi     $a0, $zero, 180
  jal      ROTATE
sleep4:    addi     $v0,$zero,32
  li       $a0,10000
  syscall
  jal UNTRACK
  jal TRACK
goLEFT:    addi$a0, $zero, 270
  jal ROTATE
sleep5:    addi$v0,$zero,32
  li  $a0,10000
  syscall
  jal UNTRACK
  jal TRACK
goUP: addi$a0,$zero,0
  jal ROTATE
sleep6:    addi$v0,$zero,32
  li  $a0,10000
  syscall
```

```
  jal      UNTRACK
goRIGHT1:  addi     $a0, $zero, 90
  jal      ROTATE
sleep7:    addi$v0,$zero,32
  li   $a0,12000
  syscall
  jal       STOP

end_main:
  li        $v0,10
  syscall

#-----------------------------------------------
-----------
# GO procedure, to start running
# param[in]    none
#-----------------------------------------------
-----------
GO:        li       $at, MOVING
  addi     $k0, $zero,1     # to  logic 1,
  sb       $k0, 0($at)      # to start running
  jr       $ra
#-----------------------------------------------
-----------
# STOP procedure, to stop running
# param[in]    none
#-----------------------------------------------
-----------
STOP:      li       $at, MOVING
  sb       $zero, 0($at)    # to stop
  jr       $ra
#-----------------------------------------------
-----------
# TRACK procedure, to start drawing line
# param[in]    none
#-----------------------------------------------
-----------
```

```
TRACK:     li       $at, LEAVETRACK
           addi     $k0, $zero,1
           sb       $k0, 0($at)
           jr       $ra
#-----------------------------------------------
-----------
# UNTRACK procedure, to stop drawing line
# param[in]    none
#-----------------------------------------------
-----------
UNTRACK:   li       $at, LEAVETRACK
           sb       $zero, 0($at)
           jr       $ra
#-----------------------------------------------
-----------
# ROTATE procedure, to rotate the robot
# param[in]    $a0, An angle between 0 and 359
#                   0 : North (up)
# 90: East (right)
# 180: South (down)
# 270: West (left)
#-----------------------------------------------
-----------
ROTATE:    li   $at, HEADING
           sw   $a0, 0($at)
           jr   $ra
```

- Kết quả chạy:




3. Vẽ ngôi sao 5 cánh:
   - Code:

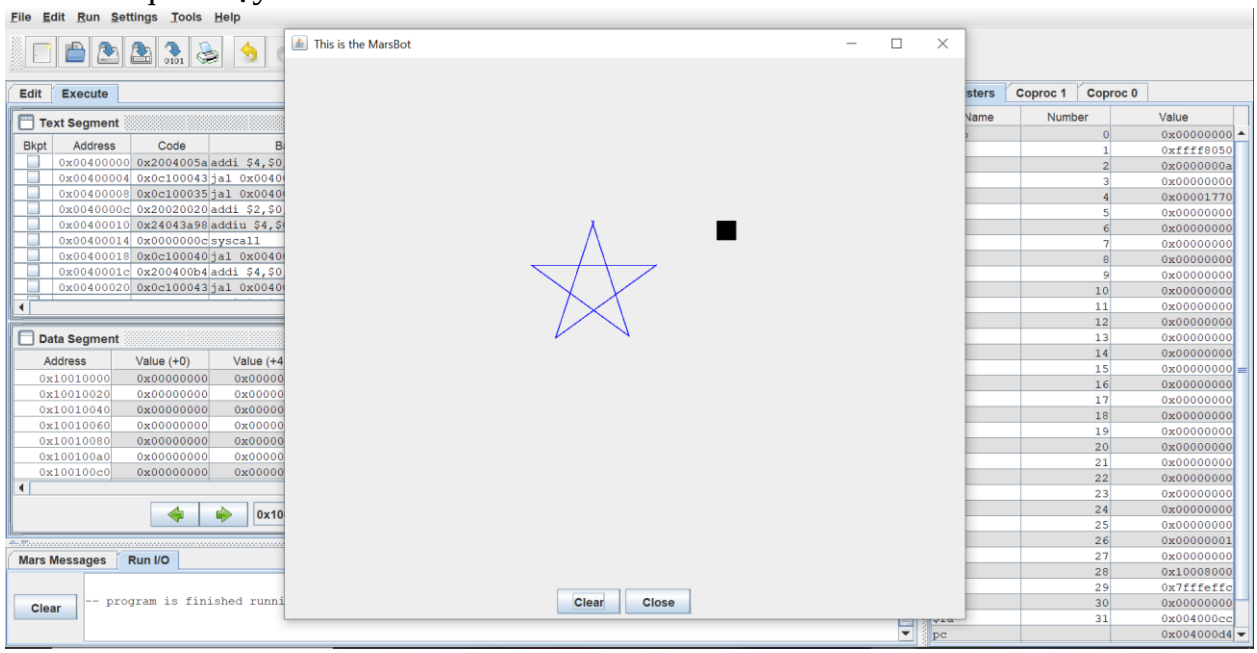
```
.eqv   HEADING     0xffff8010

.eqv   MOVING      0xffff8050

.eqv   LEAVETRACK 0xffff8020
.eqv   WHEREX      0xffff8030
.eqv   WHEREY      0xffff8040
.text
main:
   addi      $a0, $zero, 90
   jal       ROTATE
   jal       GO
sleep1:     addi      $v0,$zero,32
   li        $a0,15000
   syscall
   jal       UNTRACK
goDOWN:     addi$a0, $zero, 180
   jal       ROTATE
sleep2:     addi      $v0,$zero,32
```

```
  li       $a0,8000
  syscall
  jal      UNTRACK
  jal      TRACK

go120:     addi     $a0, $zero, 162
  jal      ROTATE
sleep3:    addi     $v0,$zero,32
  li       $a0,6000
  syscall
  jal      UNTRACK
  jal      TRACK
go30: addi     $a0, $zero, 306
  jal      ROTATE
sleep4:    addi     $v0,$zero,32
  li       $a0,6000
  syscall
  jal UNTRACK
  jal TRACK
goleft:    addi$a0, $zero, 90
  jal ROTATE
sleep5:    addi$v0,$zero,32
  li  $a0,6000
  syscall
  jal UNTRACK
  jal TRACK
goL:   addi$a0,$zero,234
  jal ROTATE
  li  $a0,6000
  syscall
  jal UNTRACK
  jal TRACK
go1:   addi$a0,$zero,18
  jal ROTATE
  li  $a0,6000
  syscall
  jal UNTRACK
```

```
go12:  addi$a0,$zero,90
  jal  ROTATE
  li   $a0,6000
  syscall
  jal       STOP
end_main:
  li        $v0,10
  syscall
#-----------------------------------------------
-----------
# GO procedure, to start running
# param[in]    none
#-----------------------------------------------
-----------
GO:        li       $at, MOVING
  addi     $k0, $zero,1     # to  logic 1,
  sb       $k0, 0($at)      # to start running
  jr       $ra
#-----------------------------------------------
-----------
# STOP procedure, to stop running
# param[in]    none
#-----------------------------------------------
-----------
STOP:      li       $at, MOVING
  sb       $zero, 0($at)    # to stop
  jr       $ra
#-----------------------------------------------
-----------
# TRACK procedure, to start drawing line
# param[in]    none
#-----------------------------------------------
-----------
TRACK:     li       $at, LEAVETRACK
           addi     $k0, $zero,1
           sb       $k0, 0($at)
           jr       $ra
```

```
#-----------------------------------------------
-----------
# UNTRACK procedure, to stop drawing line
# param[in]    none
#-----------------------------------------------
-----------
UNTRACK:    li       $at, LEAVETRACK
            sb       $zero, 0($at)
            jr       $ra
#-----------------------------------------------
-----------
# ROTATE procedure, to rotate the robot
# param[in]    $a0, An angle between 0 and 359
#                   0 : North (up)
# 90: East (right)
# 180: South (down)
# 270: West (left)
#-----------------------------------------------
-----------
ROTATE:     li   $at, HEADING
            sw   $a0, 0($at)
            jr   $ra
```

- Kết quả chạy:



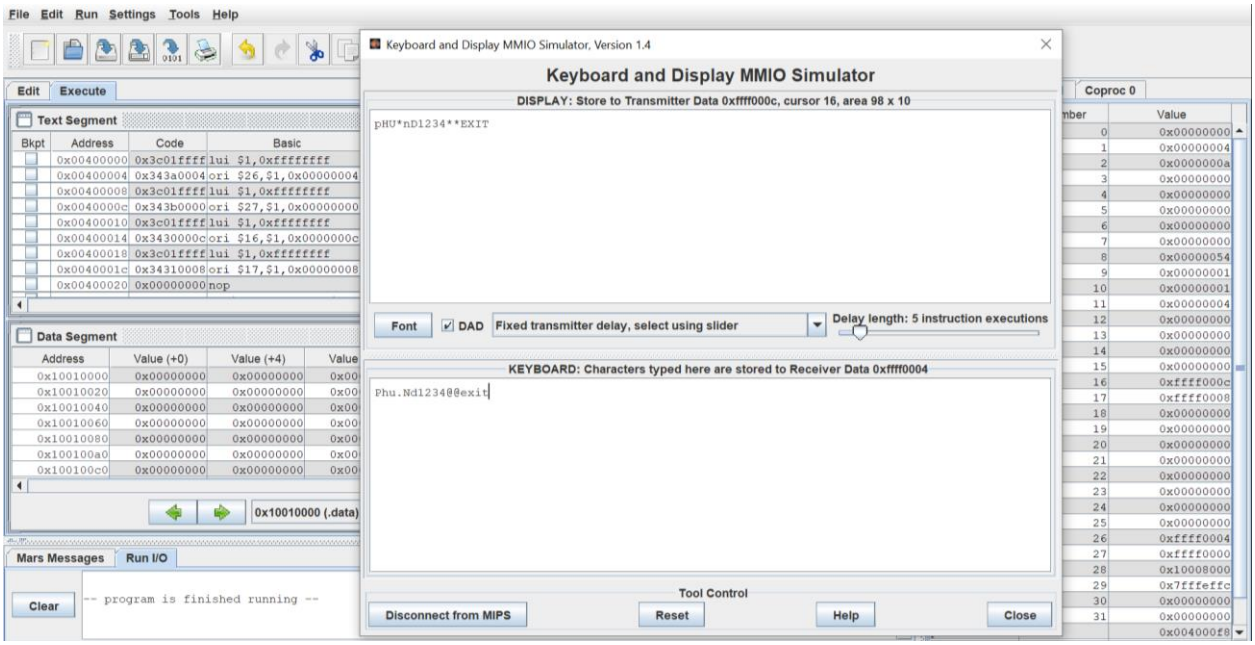
## Assignment 2:

- Code:

```
.eqv KEY_CODE   0xFFFF0004
.eqv KEY_READY  0xFFFF0000
.eqv DISPLAY_CODE   0xFFFF000C
.eqv DISPLAY_READY  0xFFFF0008
.text
  li   $k0,  KEY_CODE
  li   $k1,  KEY_READY
  li   $s0, DISPLAY_CODE
  li   $s1, DISPLAY_READY
loop:      nop

WaitForKey:
  lw   $t1, 0($k1)
  beq  $t1, $zero, WaitForKey

ReadKey:
  lw   $t0, 0($k0)

WaitForDis:
  lw   $t2, 0($s1)
  beq  $t2, $zero, WaitForDis
Kiemtra:
KiemTraE:
  beq $t3, 1, KiemTraX
  beq $t0, 101, Co

KiemTraX:
  beq $t3, 2, KiemTraI
  beq $t0, 120, Co

KiemTraI:
  beq $t3, 3, KiemTraT
  beq $t0, 105, Co
```

```
KiemTraT:
  beq $t3, 4, Encrypt2
  beq $t0, 116, Co

Encrypt:
  addi$t3, $zero, 0
Encrypt2:
ChuHoa:    bgt $t0, 90, ChuThuong
  blt $t0, 65, ChuThuong
  addi$t0, $t0, 32
  j   ShowKey
ChuThuong:
  bgt $t0, 122, ChuSo
  blt $t0, 97, ChuSo
  addi$t0, $t0, -32
  j   ShowKey
ChuSo:bgt $t0, 57, Khac
  blt $t0, 48, Khac
  addi$t0, $t0, 0
  j   ShowKey
Khac:
  addi    $t0, $zero, 42

ShowKey:
  sw  $t0, 0($s0)
  nop
  beq $t3, 4, Exit
  j loop
Co:   addi$t3, $t3, 1
  j   Encrypt2
Exit:
  li  $v0, 10
  syscall
```

- Kết quả chạy thử:




## Assignment 3:

- Code:

```
.eqv   HEADING  0xffff8010
.eqv   MOVING   0xffff8050
.eqv   LEAVETRACK   0xffff8020
.eqv   WHEREX       0xffff8030
.eqv   WHEREY       0xffff8040

.eqv KEY_CODE       0xFFFF0004
.eqv KEY_READY      0xFFFF0000
.eqv DISPLAY_CODE       0xFFFF000C
.eqv DISPLAY_READY      0xFFFF0008

.text
main:
  li        $s2,  KEY_CODE
  li        $s3,  KEY_READY
  li        $s0, DISPLAY_CODE
```

```
  li  $s1, DISPLAY_READY
start:
WaitForKey_start:
  lw  $t1, 0($s3)
  beq     $t1, $zero, WaitForKey_start

ReadKey_start:
  lw      $t0, 0($s2)

  beq $t0, 32, loop    # dau space thi bat dau
  j start  #neu khong thi lap lai

loop:

WaitForKey:
  lw  $t1, 0($s3)
  beq     $t1, $zero, WaitForKey

ReadKey:
  lw      $t0, 0($s2)

WaitForDis:
  lw      $t2, 0($s1)
          beq     $t2, $zero, WaitForDis

Input:
InputW:   beq $t0, 119, Len
  beq $t0, 87, Len

InputS:   beq $t0, 115, Xuong
  beq $t0, 83, Xuong

InputA:   beq $t0, 97, Trai
  beq $t0, 65, Trai

InputD:   beq $t0, 100, Phai
  beq $t0, 68, Phai
```

```
  j   InputExit

Len:
  addi    $a0, $zero, 0
          jal     ROTATE
  jal     GO
          jal     UNTRACK
          jal     TRACK
  addi    $v0,$zero,32
  li      $a0,100
  syscall
  j   ShowKey

Xuong:
  addi    $a0, $zero, 180
          jal     ROTATE
  jal     GO
          jal     UNTRACK
          jal     TRACK
  addi    $v0,$zero,32
  li      $a0,100
  syscall
  j   ShowKey

Trai:
  addi    $a0, $zero, 270
  jal     ROTATE
  jal     GO
  jal     UNTRACK
  jal     TRACK
  addi    $v0,$zero,32
  li      $a0,100
  syscall
  j   ShowKey

Phai:
  addi    $a0, $zero, 90
```

```
          jal     ROTATE
  jal     GO
          jal     UNTRACK
          jal     TRACK
  addi    $v0,$zero,32
  li      $a0,100
  syscall
  j   ShowKey

InputExit:
  beq $t0, 32, end_main
  j   ShowKey

ShowKey:
  sw  $t0, 0($s0)
  nop
      j   loop

end_main:
  jal     UNTRACK
  jal STOP
  li  $v0, 10
  syscall

GO:
  li    $at, MOVING
  addi  $k0, $zero,1
  sb    $k0, 0($at)
  jr    $ra

ROTATE:
  li    $at, HEADING
  sw    $a0, 0($at)
  jr    $ra

STOP:
  li    $at, MOVING
```

```
    sb      $zero, 0($at)
    jr      $ra

TRACK:
    li      $at, LEAVETRACK
    addi    $k0, $zero,1
    sb      $k0, 0($at)
    jr      $ra

UNTRACK:
    li      $at, LEAVETRACK
    sb      $zero, 0($at)
    jr      $ra
```

- Kết quả chạy thử: