ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KĨ THUẬT MÁY TÍNH

MÔN HỌC: KIẾN TRÚC MÁY TÍNH (CO2007)

Chủ đề 5:

KIẾN TRÚC TẬP LỆNH MIPS

HIỆN THỰC GIẢI THUẬT QUICKSORT

GV. Hướng dẫn:	Thầy Võ Tấn Phương	
	Thầy Trần Thanh Bình	
Sinh viên thực hiện:	Nguyễn Duy Uyên	1912410.
	Nguyễn Hoài Thương	1912184.
	Nguyễn Hồng Dân	1910916.

Ngày 6 tháng 12 năm 2020

Nội dung

1 Đề tài

- Sắp xếp chuỗi
- Cho một chuỗi số nguyên gồm 20 phần tử. Sử dụng hợp ngữ assembly MIPS, viết thủ tục sắp xếp chuỗi đó theo thứ tự tăng dần theo giải thuật quicksort.
  Yêu cầu xuất ra từng bước trong quá trình demo.

Yêu cầu:

- Thống kê số lệnh, loại lệnh của chương trình của nhóm
- Tính và trình bày cách tính thời gian chạy của chương trình trên máy tính MIPS có tần số 2GHz.

2 Code C++

- **Hàm Partition**

```
int Partition(int* start, int low, int high) {
  int pivot = start[high];
  int l = low - 1;
  for(int i = low; i < high; i++){
    if(start[i] < pivot){
      l++;
      swap(start[l], start[i]);
    }
  }
  swap(start[high], start[l + 1]);
  return l + 1;
}
```

- **Hàm Quicksort**

```
void QuickSort(int* arr, int low, int high, int size) {
  if(low < high){
    int t = Partition(arr, low, high);
    QuickSort(arr, low, t - 1, size);
    QuickSort(arr, t + 1, high, size);
  }
}
```

3 Code hợp ngữ assembly MIPS

```
###############################################################################
### ------------------------------------------------------------------- ###
###                   CODE THUC THI GIAI THUAT QUICKSORT                  ###
###                                L08                                    ###
### ------------------------------------------------------------------- ###
###############################################################################

# .data segment
.data
    # Initialize the array
    array: .word 15,5,12,20,7,3,13,19,4,8,14,2,1,16,6,9,11,17,18,10
    # Strings for array printing
    space:    .asciiz " "
    P:        .asciiz "partition: "
    high:     .asciiz "high: "
    low:      .asciiz "low: "
    newline:  .asciiz "\n"

# .text segment
.text
.globl main

main:
    la $t0, array
    addi $a0, $t0, 0                # Set argument 1 to the array.
    addi $a1, $zero, 0              # low
    addi $a2, $zero, 19             # high
    jal Quicksort                   # Call quick sort
    jal print                       # Print sorted array
    li $v0, 10                      # Terminate program run and
    syscall                         # Exit

print:
    # -------------------- Fuction to Print here  ----------------------#
    # Make room for 3 argument                                          #
    # Print array with " "                                              #
    # End by "\n"                                                       #
    # ------------------------------------------------------------------ #

Quicksort:
    # Make stack room for 5 here ...

    bge $a1, $a2, endQS          # if low >= high, jump to endQS

    jal print                    # Call print

    jal partition
    move $s0, $v0                # s0 = partition

    jal print_partition

```

```
52      sw $s0, 16($sp)              # Store partition
53      lw $a1, 4($sp)
54      addi $a2, $s0, -1
55      jal Quicksort                # Quicksort (low, partition - 1)
56
57      lw $s0, 16($sp)              # Load partition
58      addi $a1, $s0, 1
59      lw $a2, 8($sp)
60      jal Quicksort                # Quicksort (partition + 1, high)
61
62  endQS:
63      # Restore stack room for 5 here ...
64      jr $ra
65
66  partition:
67      # ---------------------- Function for partition ------------------- #
68      # Make stack room for 4                                             #
69      # Assigns the low, hight value to register $s1, $s2 respectively    #
70      # ------------------------------------------------------------------ #
71
72      sll $t3, $s2, 2              # t1 = 4*high
73      add $t3, $a0, $t3            # t1 = arr + 4*high
74      lw $t4, 0($t3)               # t2 = arr[high] //pivot
75
76      # Set l = low - 1 in $t5
77      # i = low in $t6 ...
78
79      loop:
80          # Condition 1 for loop: i < high
81          addi $t8, $t6, 1
82          slt $t8, $s2, $t8         # high <= i
83          beq $t8, 1, endloop
84              # Calculate arr[i] here and assign to $t7 here ...
85          # Condition 2 for loop: arr[i] < pivot
86          addi $t8, $t7, 1
87          slt $t8, $t4, $t8         # pivot <= arr[i]
88          beq $t8, 1, endif
89
90          addi $t5, $t5, 1          # l++
91
92          addi $a1, $t5, 0
93          addi $a2, $t6, 0
94          jal swap                  # swap (arr[l], arr[i])
95
96          endif:
97              addi $t6, $t6, 1             # i++
98              j loop
99
100     endloop:        # swap (arr[l + 1], arr[high]) ...
101
102     addi $v0, $t5, 1             # return v0 = l + 1
103     # Restore stack room for 4
104     jr $ra
```

```
105
106 swap:
107     # ------------------------ Function for swap ---------------------- #
108     # Make stack room for 4 ...
109
110     sll $t0, $a1, 2
111     add $t0, $a0, $t0
112     lw $t1, 0($t0)              # t1 = arr[i]
113
114     sll $t2, $a2, 2
115     add $t2, $a0, $t2
116     lw $t3, 0($t2)              # t3 = arr[j]
117
118     sw $t1, 0($t2)
119     sw $t3, 0($t0)
120
121     # Restore stack room for 4 ...
122     jr $ra
123
124 print_partition:
125     # ----------------- Function for print-partition ----------------- #
126     # Make stack room for 4                                            #
127     # Print string "partition"                                         #
128     # Print space " "                                                  #
129     # Print low                                                        #
130     # Print space " "                                                  #
131     # Print high                                                       #
132     # Print newline                                                    #
133     # ------------------------------------------------------------------ #
134
135
136     # Restore stack room for 4
137     jr $ra
138
139 ################################################################################
140 ### ------------------------------------------------------------------------##
141 ###                           END  PROGRAM                                  ##
142 ### ------------------------------------------------------------------------##
143 ################################################################################
```

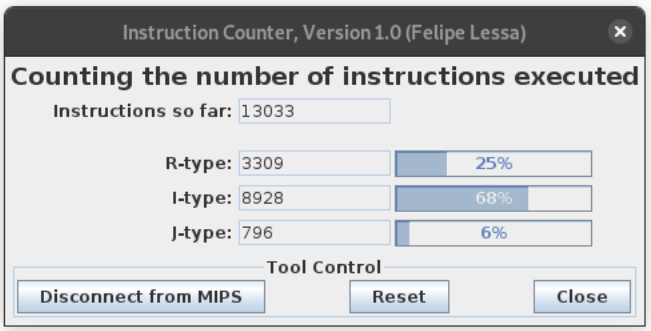
4 Demo

4.1 Test case 1:

[array:] = {20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1}

```
20 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 1
partition: 0 low: 0 high: 19
1 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 20
partition: 19 low: 1 high: 19
1 19 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 2 20
partition: 1 low: 1 high: 18
1 2 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 19 20
partition: 18 low: 2 high: 18
1 2 18 17 16 15 14 13 12 11 10 9 8 7 6 5 4 3 19 20
partition: 2 low: 2 high: 17
1 2 3 17 16 15 14 13 12 11 10 9 8 7 6 5 4 18 19 20
partition: 17 low: 3 high: 17
1 2 3 17 16 15 14 13 12 11 10 9 8 7 6 5 4 18 19 20
partition: 3 low: 3 high: 16
1 2 3 4 16 15 14 13 12 11 10 9 8 7 6 5 17 18 19 20
partition: 16 low: 4 high: 16
1 2 3 4 16 15 14 13 12 11 10 9 8 7 6 5 17 18 19 20
partition: 4 low: 4 high: 15
1 2 3 4 5 15 14 13 12 11 10 9 8 7 6 16 17 18 19 20
partition: 15 low: 5 high: 15
1 2 3 4 5 15 14 13 12 11 10 9 8 7 6 16 17 18 19 20
partition: 5 low: 5 high: 14
1 2 3 4 5 6 14 13 12 11 10 9 8 7 15 16 17 18 19 20
partition: 14 low: 6 high: 14
1 2 3 4 5 6 14 13 12 11 10 9 8 7 15 16 17 18 19 20
partition: 6 low: 6 high: 13
1 2 3 4 5 6 7 13 12 11 10 9 8 14 15 16 17 18 19 20
partition: 13 low: 7 high: 13
1 2 3 4 5 6 7 13 12 11 10 9 8 14 15 16 17 18 19 20
partition: 7 low: 7 high: 12
1 2 3 4 5 6 7 8 12 11 10 9 13 14 15 16 17 18 19 20
partition: 12 low: 8 high: 12
1 2 3 4 5 6 7 8 12 11 10 9 13 14 15 16 17 18 19 20
partition: 8 low: 8 high: 11
1 2 3 4 5 6 7 8 9 11 10 12 13 14 15 16 17 18 19 20
partition: 11 low: 9 high: 11
1 2 3 4 5 6 7 8 9 11 10 12 13 14 15 16 17 18 19 20
partition: 9 low: 9 high: 10
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

-- program is finished running --
```



Tổng số lệnh là: 13033

- R-type: 3309
- I-type: 8928
- J-type: 796

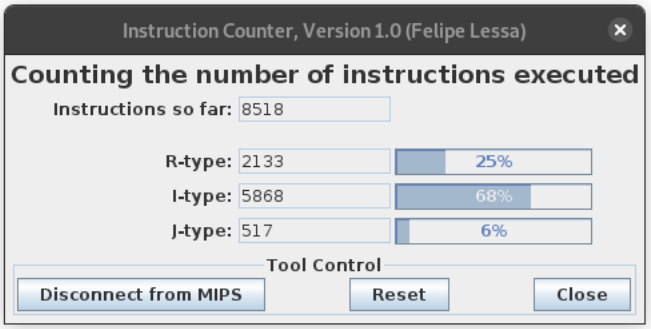
Sử dụng mô hình đơn chu kì.
Thời gian chạy của chương trình = $\frac{CPUClockCycles}{clockRate} = \frac{13033}{2*10^9} = 6516(ns)$

4.2 Test case 2:

[array:] = {15, 5, 12, 20, 7, 3, 13, 19, 4, 8, 14, 2, 1, 16, 6, 9, 11, 17, 18, 10}

```
15 5 12 20 7 3 13 19 4 8 14 2 1 16 6 9 11 17 18 10
partition: 9 low: 0 high: 19
5 7 3 4 8 2 1 6 9 10 14 12 13 16 19 20 11 17 18 15
partition: 8 low: 0 high: 8
5 7 3 4 8 2 1 6 9 10 14 12 13 16 19 20 11 17 18 15
partition: 5 low: 0 high: 7
5 3 4 2 1 6 8 7 9 10 14 12 13 16 19 20 11 17 18 15
partition: 0 low: 0 high: 4
1 3 4 2 5 6 8 7 9 10 14 12 13 16 19 20 11 17 18 15
partition: 4 low: 1 high: 4
1 3 4 2 5 6 8 7 9 10 14 12 13 16 19 20 11 17 18 15
partition: 1 low: 1 high: 3
1 2 4 3 5 6 8 7 9 10 14 12 13 16 19 20 11 17 18 15
partition: 2 low: 2 high: 3
1 2 3 4 5 6 8 7 9 10 14 12 13 16 19 20 11 17 18 15
partition: 6 low: 6 high: 7
1 2 3 4 5 6 7 8 9 10 14 12 13 16 19 20 11 17 18 15
partition: 14 low: 10 high: 19
1 2 3 4 5 6 7 8 9 10 14 12 13 11 15 20 16 17 18 19
partition: 10 low: 10 high: 13
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 20 16 17 18 19
partition: 13 low: 11 high: 13
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 20 16 17 18 19
partition: 12 low: 11 high: 12
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 20 16 17 18 19
partition: 18 low: 15 high: 19
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
partition: 17 low: 15 high: 17
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
partition: 16 low: 15 high: 16
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

-- program is finished running --
```



Tổng số lệnh là: 8518

- R-type: 2133
- I-type: 5868
- J-type: 517

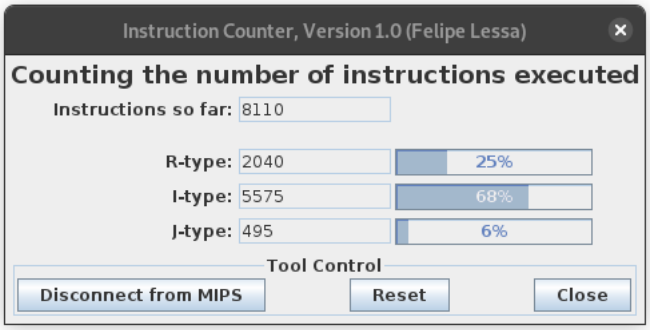
Sử dụng mô hình đơn chu kì.
Thời gian chạy của chương trình $= \frac{CPUClockCycles}{clockRate} = \frac{8518}{2*10^9} = 4259(ns)$

4.3 Test case 3:

[array:] = {-5, 6, 9, -20, 7, 1, 8, 100, 0, 3, 40, 45, 7, 1, 8, 16, 5, 18, -19, 20}

```
-5 6 9 -20 7 1 8 100 0 3 40 45 7 1 8 16 5 18 -19 20
partition: 16 low: 0 high: 19
-5 6 9 -20 7 1 8 0 3 7 1 8 16 5 18 -19 20 45 100 40
partition: 1 low: 0 high: 15
-20 -19 9 -5 7 1 8 0 3 7 1 8 16 5 18 6 20 45 100 40
partition: 8 low: 2 high: 15
-20 -19 -5 1 0 3 1 5 6 7 8 8 16 7 18 9 20 45 100 40
partition: 7 low: 2 high: 7
-20 -19 -5 1 0 3 1 5 6 7 8 8 16 7 18 9 20 45 100 40
partition: 4 low: 2 high: 6
-20 -19 -5 0 1 3 1 5 6 7 8 8 16 7 18 9 20 45 100 40
partition: 3 low: 2 high: 3
-20 -19 -5 0 1 3 1 5 6 7 8 8 16 7 18 9 20 45 100 40
partition: 5 low: 5 high: 6
-20 -19 -5 0 1 1 3 5 6 7 8 8 16 7 18 9 20 45 100 40
partition: 13 low: 9 high: 15
-20 -19 -5 0 1 1 3 5 6 7 8 8 7 9 18 16 20 45 100 40
partition: 9 low: 9 high: 12
-20 -19 -5 0 1 1 3 5 6 7 8 8 7 9 18 16 20 45 100 40
partition: 10 low: 10 high: 12
-20 -19 -5 0 1 1 3 5 6 7 7 8 8 9 18 16 20 45 100 40
partition: 11 low: 11 high: 12
-20 -19 -5 0 1 1 3 5 6 7 7 8 8 9 18 16 20 45 100 40
partition: 14 low: 14 high: 15
-20 -19 -5 0 1 1 3 5 6 7 7 8 8 9 16 18 20 45 100 40
partition: 17 low: 17 high: 19
-20 -19 -5 0 1 1 3 5 6 7 7 8 8 9 16 18 20 40 100 45
partition: 18 low: 18 high: 19
-20 -19 -5 0 1 1 3 5 6 7 7 8 8 9 16 18 20 40 45 100

-- program is finished running --
```



Tổng số lệnh là: 8110

- R-type: 2040
- I-type: 5575
- J-type: 495

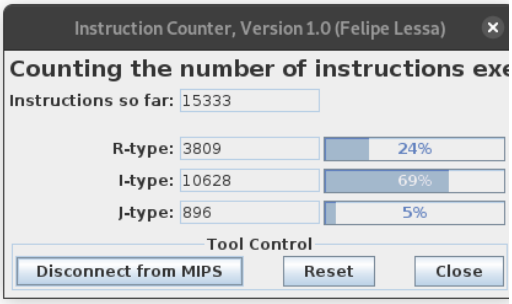
Sử dụng mô hình đơn chu kì.
Thời gian chạy của chương trình $= \frac{CPUClockCycles}{clockRate} = \frac{8110}{2*10^9} = 4055(ns)$

4.4 Test case 4:

[array:] = {-95, -80, -78, -74, -53, -43, -40,-38, -30, -25, -19, -12, -1, 13, 49, 50, 60, 71, 90, 94}

```
-95 -80 -78 -74 -53 -43 -40 -38 -30 -25 -19 -12 -1 13 49 50 60 71 90 94
partition: 19 low: 0 high: 19
-95 -80 -78 -74 -53 -43 -40 -38 -30 -25 -19 -12 -1 13 49 50 60 71 90 94
partition: 18 low: 0 high: 18
-95 -80 -78 -74 -53 -43 -40 -38 -30 -25 -19 -12 -1 13 49 50 60 71 90 94
partition: 17 low: 0 high: 17
-95 -80 -78 -74 -53 -43 -40 -38 -30 -25 -19 -12 -1 13 49 50 60 71 90 94
partition: 16 low: 0 high: 16
-95 -80 -78 -74 -53 -43 -40 -38 -30 -25 -19 -12 -1 13 49 50 60 71 90 94
partition: 15 low: 0 high: 15
-95 -80 -78 -74 -53 -43 -40 -38 -30 -25 -19 -12 -1 13 49 50 60 71 90 94
partition: 14 low: 0 high: 14
-95 -80 -78 -74 -53 -43 -40 -38 -30 -25 -19 -12 -1 13 49 50 60 71 90 94
partition: 13 low: 0 high: 13
-95 -80 -78 -74 -53 -43 -40 -38 -30 -25 -19 -12 -1 13 49 50 60 71 90 94
partition: 12 low: 0 high: 12
-95 -80 -78 -74 -53 -43 -40 -38 -30 -25 -19 -12 -1 13 49 50 60 71 90 94
partition: 11 low: 0 high: 11
-95 -80 -78 -74 -53 -43 -40 -38 -30 -25 -19 -12 -1 13 49 50 60 71 90 94
partition: 10 low: 0 high: 10
-95 -80 -78 -74 -53 -43 -40 -38 -30 -25 -19 -12 -1 13 49 50 60 71 90 94
partition: 9 low: 0 high: 9
-95 -80 -78 -74 -53 -43 -40 -38 -30 -25 -19 -12 -1 13 49 50 60 71 90 94
partition: 8 low: 0 high: 8
-95 -80 -78 -74 -53 -43 -40 -38 -30 -25 -19 -12 -1 13 49 50 60 71 90 94
partition: 7 low: 0 high: 7
-95 -80 -78 -74 -53 -43 -40 -38 -30 -25 -19 -12 -1 13 49 50 60 71 90 94
partition: 6 low: 0 high: 6
-95 -80 -78 -74 -53 -43 -40 -38 -30 -25 -19 -12 -1 13 49 50 60 71 90 94
partition: 5 low: 0 high: 5
-95 -80 -78 -74 -53 -43 -40 -38 -30 -25 -19 -12 -1 13 49 50 60 71 90 94
partition: 4 low: 0 high: 4
-95 -80 -78 -74 -53 -43 -40 -38 -30 -25 -19 -12 -1 13 49 50 60 71 90 94
partition: 3 low: 0 high: 3
-95 -80 -78 -74 -53 -43 -40 -38 -30 -25 -19 -12 -1 13 49 50 60 71 90 94
partition: 2 low: 0 high: 2
-95 -80 -78 -74 -53 -43 -40 -38 -30 -25 -19 -12 -1 13 49 50 60 71 90 94
partition: 1 low: 0 high: 1
-95 -80 -78 -74 -53 -43 -40 -38 -30 -25 -19 -12 -1 13 49 50 60 71 90 94

-- program is finished running --
```



Tổng số lệnh là: 15333

- R-type: 3809
- I-type: 10628
- J-type: 896

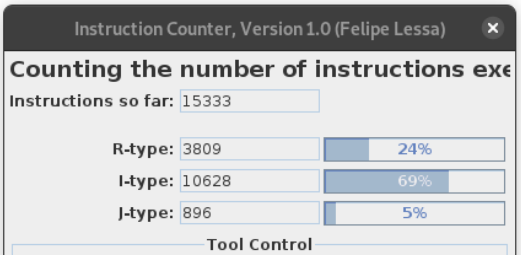
Sử dụng mô hình đơn chu kì.
Thời gian chạy của chương trình $= \frac{CPUClockCycles}{clockRate} = \frac{15333}{2*10^9} = 7666(ns)$

4.5 Test case 5:

[array:] = {-78443, -76863, -73970, -63648, -54472, -42792, -31733, -23956, -20561, -9765, -2202, 5583, 9451, 22731, 44872, 47958, 53344, 64902, 79039, 85313}

```
-78443 -76863 -73970 -63648 -54472 -42792 -31733 -23956 -20561 -9765 -2202 5583 9451 22731 44872 47958 53344 64902 79039 85313
partition: 19 low: 0 high: 19
-78443 -76863 -73970 -63648 -54472 -42792 -31733 -23956 -20561 -9765 -2202 5583 9451 22731 44872 47958 53344 64902 79039 85313
partition: 18 low: 0 high: 18
-78443 -76863 -73970 -63648 -54472 -42792 -31733 -23956 -20561 -9765 -2202 5583 9451 22731 44872 47958 53344 64902 79039 85313
partition: 17 low: 0 high: 17
-78443 -76863 -73970 -63648 -54472 -42792 -31733 -23956 -20561 -9765 -2202 5583 9451 22731 44872 47958 53344 64902 79039 85313
partition: 16 low: 0 high: 16
-78443 -76863 -73970 -63648 -54472 -42792 -31733 -23956 -20561 -9765 -2202 5583 9451 22731 44872 47958 53344 64902 79039 85313
partition: 15 low: 0 high: 15
-78443 -76863 -73970 -63648 -54472 -42792 -31733 -23956 -20561 -9765 -2202 5583 9451 22731 44872 47958 53344 64902 79039 85313
partition: 14 low: 0 high: 14
-78443 -76863 -73970 -63648 -54472 -42792 -31733 -23956 -20561 -9765 -2202 5583 9451 22731 44872 47958 53344 64902 79039 85313
partition: 13 low: 0 high: 13
-78443 -76863 -73970 -63648 -54472 -42792 -31733 -23956 -20561 -9765 -2202 5583 9451 22731 44872 47958 53344 64902 79039 85313
partition: 12 low: 0 high: 12
-78443 -76863 -73970 -63648 -54472 -42792 -31733 -23956 -20561 -9765 -2202 5583 9451 22731 44872 47958 53344 64902 79039 85313
partition: 11 low: 0 high: 11
-78443 -76863 -73970 -63648 -54472 -42792 -31733 -23956 -20561 -9765 -2202 5583 9451 22731 44872 47958 53344 64902 79039 85313
partition: 10 low: 0 high: 10
-78443 -76863 -73970 -63648 -54472 -42792 -31733 -23956 -20561 -9765 -2202 5583 9451 22731 44872 47958 53344 64902 79039 85313
partition: 9 low: 0 high: 9
-78443 -76863 -73970 -63648 -54472 -42792 -31733 -23956 -20561 -9765 -2202 5583 9451 22731 44872 47958 53344 64902 79039 85313
partition: 8 low: 0 high: 8
-78443 -76863 -73970 -63648 -54472 -42792 -31733 -23956 -20561 -9765 -2202 5583 9451 22731 44872 47958 53344 64902 79039 85313
partition: 7 low: 0 high: 7
-78443 -76863 -73970 -63648 -54472 -42792 -31733 -23956 -20561 -9765 -2202 5583 9451 22731 44872 47958 53344 64902 79039 85313
partition: 6 low: 0 high: 6
-78443 -76863 -73970 -63648 -54472 -42792 -31733 -23956 -20561 -9765 -2202 5583 9451 22731 44872 47958 53344 64902 79039 85313
partition: 5 low: 0 high: 5
-78443 -76863 -73970 -63648 -54472 -42792 -31733 -23956 -20561 -9765 -2202 5583 9451 22731 44872 47958 53344 64902 79039 85313
partition: 4 low: 0 high: 4
-78443 -76863 -73970 -63648 -54472 -42792 -31733 -23956 -20561 -9765 -2202 5583 9451 22731 44872 47958 53344 64902 79039 85313
partition: 3 low: 0 high: 3
-78443 -76863 -73970 -63648 -54472 -42792 -31733 -23956 -20561 -9765 -2202 5583 9451 22731 44872 47958 53344 64902 79039 85313
partition: 2 low: 0 high: 2
-78443 -76863 -73970 -63648 -54472 -42792 -31733 -23956 -20561 -9765 -2202 5583 9451 22731 44872 47958 53344 64902 79039 85313
partition: 1 low: 0 high: 1
-78443 -76863 -73970 -63648 -54472 -42792 -31733 -23956 -20561 -9765 -2202 5583 9451 22731 44872 47958 53344 64902 79039 85313

-- program is finished running --
```



Tổng số lệnh là: 15333

- R-type: 3809
- I-type: 10628
- J-type: 896

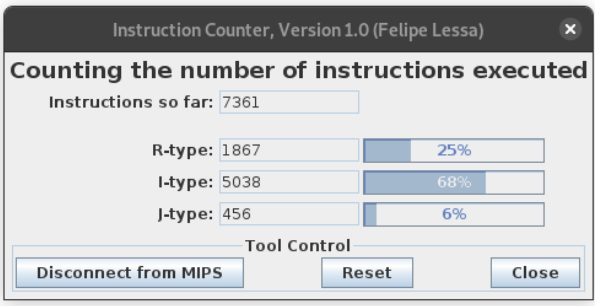
Sử dụng mô hình đơn chu kì.
Thời gian chạy của chương trình = $\frac{CPUClockCycles}{clockRate} = \frac{15333}{2*10^9} = 7666(ns)$

4.6 Test case 6:

[array:] = {86, -49, 22, -74, 17, -21, -19, 84, 69, -60, -39, -42, 57, 40, -16, -85, 71, -75, 92, -78}

```
86 -49 22 -74 17 -21 -19 84 69 -60 -39 -42 57 40 -16 -85 71 -75 92 -78
partition: 1 low: 0 high: 19
-85 -78 22 -74 17 -21 -19 84 69 -60 -39 -42 57 40 -16 86 71 -75 92 -49
partition: 5 low: 2 high: 19
-85 -78 -74 -60 -75 -49 -19 84 69 22 -39 -42 57 40 -16 86 71 17 92 -21
partition: 2 low: 2 high: 4
-85 -78 -75 -60 -74 -49 -19 84 69 22 -39 -42 57 40 -16 86 71 17 92 -21
partition: 3 low: 3 high: 4
-85 -78 -75 -74 -60 -49 -19 84 69 22 -39 -42 57 40 -16 86 71 17 92 -21
partition: 8 low: 6 high: 19
-85 -78 -75 -74 -60 -49 -39 -42 -21 22 -19 84 57 40 -16 86 71 17 92 69
partition: 6 low: 6 high: 7
-85 -78 -75 -74 -60 -49 -42 -39 -21 22 -19 84 57 40 -16 86 71 17 92 69
partition: 15 low: 9 high: 19
-85 -78 -75 -74 -60 -49 -42 -39 -21 22 -19 57 40 -16 17 69 71 84 92 86
partition: 11 low: 9 high: 14
-85 -78 -75 -74 -60 -49 -42 -39 -21 -19 -16 17 40 22 57 69 71 84 92 86
partition: 10 low: 9 high: 10
-85 -78 -75 -74 -60 -49 -42 -39 -21 -19 -16 17 40 22 57 69 71 84 92 86
partition: 14 low: 12 high: 14
-85 -78 -75 -74 -60 -49 -42 -39 -21 -19 -16 17 40 22 57 69 71 84 92 86
partition: 12 low: 12 high: 13
-85 -78 -75 -74 -60 -49 -42 -39 -21 -19 -16 17 22 40 57 69 71 84 92 86
partition: 18 low: 16 high: 19
-85 -78 -75 -74 -60 -49 -42 -39 -21 -19 -16 17 22 40 57 69 71 84 86 92
partition: 17 low: 16 high: 17
-85 -78 -75 -74 -60 -49 -42 -39 -21 -19 -16 17 22 40 57 69 71 84 86 92

-- program is finished running --
```



Tổng số lệnh là: 7361

- R-type: 1867
- I-type: 5038
- J-type: 456

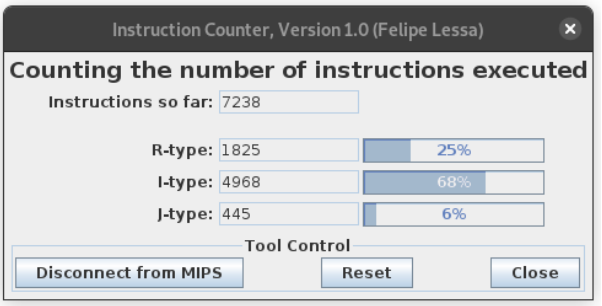
Sử dụng mô hình đơn chu kì.
Thời gian chạy của chương trình $= \frac{CPUClockCycles}{clockRate} = \frac{7361}{2*10^9} = 3680(ns)$

4.7 Test case 7:

[array:] = {42, -78, -40, 41, 84, 59, 76, -52, -26, 44, -48, 5, -36, -58, 21, -79, 6, 88, 63, -34}

```
42 -78 -40 41 84 59 76 -52 -26 44 -48 5 -36 -58 21 -79 6 88 63 -34
partition: 7 low: 0 high: 19
-78 -40 -52 -48 -36 -58 -79 -34 -26 44 41 5 84 59 21 76 6 88 63 42
partition: 0 low: 0 high: 6
-79 -40 -52 -48 -36 -58 -78 -34 -26 44 41 5 84 59 21 76 6 88 63 42
partition: 1 low: 1 high: 6
-79 -78 -52 -48 -36 -58 -40 -34 -26 44 41 5 84 59 21 76 6 88 63 42
partition: 5 low: 2 high: 6
-79 -78 -52 -48 -58 -40 -36 -34 -26 44 41 5 84 59 21 76 6 88 63 42
partition: 2 low: 2 high: 4
-79 -78 -58 -48 -52 -40 -36 -34 -26 44 41 5 84 59 21 76 6 88 63 42
partition: 3 low: 3 high: 4
-79 -78 -58 -52 -48 -40 -36 -34 -26 44 41 5 84 59 21 76 6 88 63 42
partition: 13 low: 8 high: 19
-79 -78 -58 -52 -48 -40 -36 -34 -26 41 5 21 6 42 44 76 84 88 63 59
partition: 10 low: 8 high: 12
-79 -78 -58 -52 -48 -40 -36 -34 -26 5 6 21 41 42 44 76 84 88 63 59
partition: 9 low: 8 high: 9
-79 -78 -58 -52 -48 -40 -36 -34 -26 5 6 21 41 42 44 76 84 88 63 59
partition: 12 low: 11 high: 12
-79 -78 -58 -52 -48 -40 -36 -34 -26 5 6 21 41 42 44 76 84 88 63 59
partition: 15 low: 14 high: 19
-79 -78 -58 -52 -48 -40 -36 -34 -26 5 6 21 41 42 44 59 84 88 63 76
partition: 17 low: 16 high: 19
-79 -78 -58 -52 -48 -40 -36 -34 -26 5 6 21 41 42 44 59 63 76 84 88
partition: 19 low: 18 high: 19
-79 -78 -58 -52 -48 -40 -36 -34 -26 5 6 21 41 42 44 59 63 76 84 88

-- program is finished running --
```



Tổng số lệnh là: 7238

- R-type: 1825
- I-type: 4968
- J-type: 445

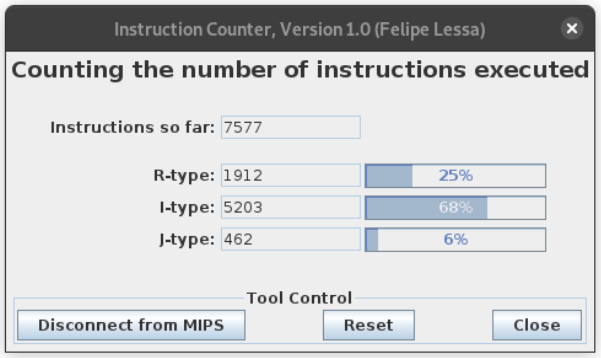
Sử dụng mô hình đơn chu kì.
Thời gian chạy của chương trình $= \frac{CPUClockCycles}{clockRate} = \frac{7238}{2*10^9} = 3619(ns)$

4.8 Test case 8:

[array:] = {-53, 84, 1, 63, -43, -23, -46, -27, 84, 33, 36, 91, -59, 26, 85, 31, 11, 63, -67, -59}

```
-53 84 1 63 -43 -23 -46 -27 84 33 36 91 -59 26 85 31 11 63 -67 -59
partition: 1 low: 0 high: 19
-67 -59 1 63 -43 -23 -46 -27 84 33 36 91 -59 26 85 31 11 63 -53 84
partition: 16 low: 2 high: 19
-67 -59 1 63 -43 -23 -46 -27 33 36 -59 26 31 11 63 -53 84 85 84 91
partition: 3 low: 2 high: 15
-67 -59 -59 -53 -43 -23 -46 -27 33 36 1 26 31 11 63 63 84 85 84 91
partition: 14 low: 4 high: 15
-67 -59 -59 -53 -43 -23 -46 -27 33 36 1 26 31 11 63 63 84 85 84 91
partition: 9 low: 4 high: 13
-67 -59 -59 -53 -43 -23 -46 -27 1 11 33 26 31 36 63 63 84 85 84 91
partition: 8 low: 4 high: 8
-67 -59 -59 -53 -43 -23 -46 -27 1 11 33 26 31 36 63 63 84 85 84 91
partition: 6 low: 4 high: 7
-67 -59 -59 -53 -43 -46 -27 -23 1 11 33 26 31 36 63 63 84 85 84 91
partition: 4 low: 4 high: 5
-67 -59 -59 -53 -46 -43 -27 -23 1 11 33 26 31 36 63 63 84 85 84 91
partition: 13 low: 10 high: 13
-67 -59 -59 -53 -46 -43 -27 -23 1 11 33 26 31 36 63 63 84 85 84 91
partition: 11 low: 10 high: 12
-67 -59 -59 -53 -46 -43 -27 -23 1 11 26 31 33 36 63 63 84 85 84 91
partition: 19 low: 17 high: 19
-67 -59 -59 -53 -46 -43 -27 -23 1 11 26 31 33 36 63 63 84 85 84 91
partition: 17 low: 17 high: 18
-67 -59 -59 -53 -46 -43 -27 -23 1 11 26 31 33 36 63 63 84 84 85 91

-- program is finished running --
```



Tổng số lệnh là: 7577

- R-type: 1912
- I-type: 5203
- J-type: 462

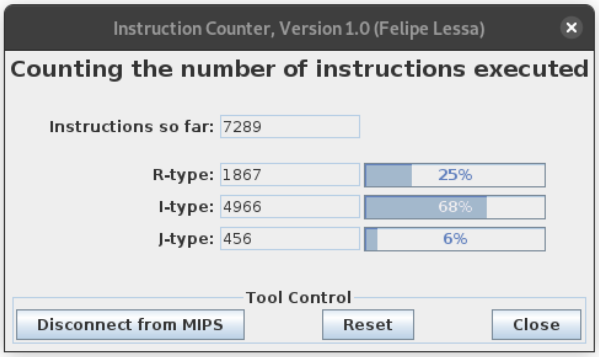
Sử dụng mô hình đơn chu kì.
Thời gian chạy của chương trình $= \frac{CPUClockCycles}{clockRate} = \frac{7577}{2*10^9} = 3789(ns)$

4.9 Test case 9:

[array:] = {1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 18, 16, 14, 12, 10, 8, 6, 4, 2, 0}

```
1 3 5 7 9 11 13 15 17 19 18 16 14 12 10 8 6 4 2 0
partition: 0 low: 0 high: 19
0 3 5 7 9 11 13 15 17 19 18 16 14 12 10 8 6 4 2 1
partition: 1 low: 1 high: 19
0 1 5 7 9 11 13 15 17 19 18 16 14 12 10 8 6 4 2 3
partition: 3 low: 2 high: 19
0 1 2 3 9 11 13 15 17 19 18 16 14 12 10 8 6 4 5 7
partition: 7 low: 4 high: 19
0 1 2 3 6 4 5 7 17 19 18 16 14 12 10 8 9 11 13 15
partition: 5 low: 4 high: 6
0 1 2 3 4 5 6 7 17 19 18 16 14 12 10 8 9 11 13 15
partition: 15 low: 8 high: 19
0 1 2 3 4 5 6 7 14 12 10 8 9 11 13 15 17 19 18 16
partition: 13 low: 8 high: 14
0 1 2 3 4 5 6 7 12 10 8 9 11 13 14 15 17 19 18 16
partition: 11 low: 8 high: 12
0 1 2 3 4 5 6 7 10 8 9 11 12 13 14 15 17 19 18 16
partition: 9 low: 8 high: 10
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 17 19 18 16
partition: 16 low: 16 high: 19
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 19 18 17
partition: 17 low: 17 high: 19
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19
partition: 19 low: 18 high: 19
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19

-- program is finished running --
```



Tổng số lệnh là: 7289

- R-type: 1867
- I-type: 4966
- J-type: 456

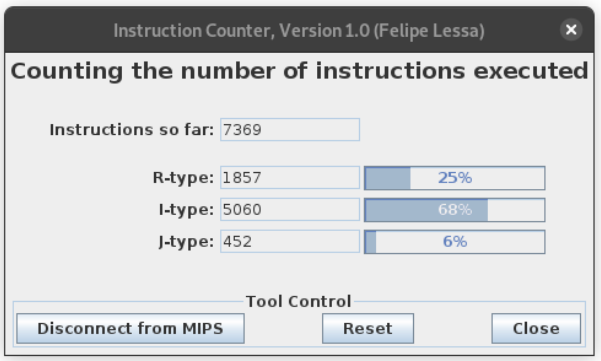
Sử dụng mô hình đơn chu kì.
Thời gian chạy của chương trình $= \frac{CPUClockCycles}{clockRate} = \frac{7289}{2*10^9} = 3644(ns)$

4.10 Test case 10:

[array:] = {-100, 65, 26, 0, 47, -25, -59, 51, 15, -94, 40, 19, 88, 5, -58, 81, 93, -1, -46, -76}

```
-100 65 26 0 47 -25 -59 51 15 -94 40 19 88 5 -58 81 93 -1 -46 -76
partition: 2 low: 0 high: 19
-100 -94 -76 0 47 -25 -59 51 15 65 40 19 88 5 -58 81 93 -1 -46 26
partition: 1 low: 0 high: 1
-100 -94 -76 0 47 -25 -59 51 15 65 40 19 88 5 -58 81 93 -1 -46 26
partition: 12 low: 3 high: 19
-100 -94 -76 0 -25 -59 15 19 5 -58 -1 -46 26 47 65 81 93 40 51 88
partition: 5 low: 3 high: 11
-100 -94 -76 -59 -58 -46 15 19 5 -25 -1 0 26 47 65 81 93 40 51 88
partition: 4 low: 3 high: 4
-100 -94 -76 -59 -58 -46 15 19 5 -25 -1 0 26 47 65 81 93 40 51 88
partition: 8 low: 6 high: 11
-100 -94 -76 -59 -58 -46 -25 -1 0 15 19 5 26 47 65 81 93 40 51 88
partition: 7 low: 6 high: 7
-100 -94 -76 -59 -58 -46 -25 -1 0 15 19 5 26 47 65 81 93 40 51 88
partition: 9 low: 9 high: 11
-100 -94 -76 -59 -58 -46 -25 -1 0 5 19 15 26 47 65 81 93 40 51 88
partition: 10 low: 10 high: 11
-100 -94 -76 -59 -58 -46 -25 -1 0 5 15 19 26 47 65 81 93 40 51 88
partition: 18 low: 13 high: 19
-100 -94 -76 -59 -58 -46 -25 -1 0 5 15 19 26 47 65 81 40 51 88 93
partition: 15 low: 13 high: 17
-100 -94 -76 -59 -58 -46 -25 -1 0 5 15 19 26 47 40 51 65 81 88 93
partition: 13 low: 13 high: 14
-100 -94 -76 -59 -58 -46 -25 -1 0 5 15 19 26 40 47 51 65 81 88 93
partition: 17 low: 16 high: 17
-100 -94 -76 -59 -58 -46 -25 -1 0 5 15 19 26 40 47 51 65 81 88 93

-- program is finished running --
```



Tổng số lệnh là: 7369 chu kì.

- R-type: 1857
- I-type: 5060
- J-type: 452

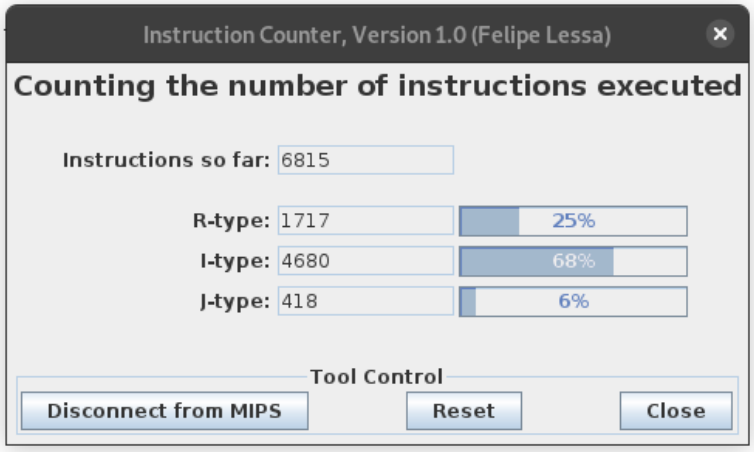
Sử dụng mô hình đơn chu kì.
Thời gian chạy của chương trình = $\frac{CPUClockCycles}{clockRate} = \frac{7369}{2*10^9} = 3685(ns)$

4.11 Test case 11:

[array:] = {7150, 3381, 2237, 4555, 1806, 9, 9252, 4103, 2289, 7072, 8695, 7586, 6800, 2196, 9724, 8463, 5507, 7269, 1260, 2211}

```
7150 3381 2237 4555 1806 9 9252 4103 2289 7072 8695 7586 6800 2196 9724 8463 5507 7269 1260 2211
partition: 4 low: 0 high: 19
1806 9 2196 1260 2211 3381 9252 4103 2289 7072 8695 7586 6800 2237 9724 8463 5507 7269 4555 7150
partition: 1 low: 0 high: 3
9 1260 2196 1806 2211 3381 9252 4103 2289 7072 8695 7586 6800 2237 9724 8463 5507 7269 4555 7150
partition: 2 low: 2 high: 3
9 1260 1806 2196 2211 3381 9252 4103 2289 7072 8695 7586 6800 2237 9724 8463 5507 7269 4555 7150
partition: 13 low: 5 high: 19
9 1260 1806 2196 2211 3381 4103 2289 7072 6800 2237 5507 4555 7150 9724 8463 7586 7269 9252 8695
partition: 9 low: 5 high: 12
9 1260 1806 2196 2211 3381 4103 2289 2237 4555 7072 5507 6800 7150 9724 8463 7586 7269 9252 8695
partition: 5 low: 5 high: 8
9 1260 1806 2196 2211 2237 4103 2289 3381 4555 7072 5507 6800 7150 9724 8463 7586 7269 9252 8695
partition: 7 low: 6 high: 8
9 1260 1806 2196 2211 2237 2289 3381 4103 4555 7072 5507 6800 7150 9724 8463 7586 7269 9252 8695
partition: 11 low: 10 high: 12
9 1260 1806 2196 2211 2237 2289 3381 4103 4555 5507 6800 7072 7150 9724 8463 7586 7269 9252 8695
partition: 17 low: 14 high: 19
9 1260 1806 2196 2211 2237 2289 3381 4103 4555 5507 6800 7072 7150 8463 7586 7269 8695 9252 9724
partition: 14 low: 14 high: 16
9 1260 1806 2196 2211 2237 2289 3381 4103 4555 5507 6800 7072 7150 7269 7586 8463 8695 9252 9724
partition: 16 low: 15 high: 16
9 1260 1806 2196 2211 2237 2289 3381 4103 4555 5507 6800 7072 7150 7269 7586 8463 8695 9252 9724
partition: 19 low: 18 high: 19
9 1260 1806 2196 2211 2237 2289 3381 4103 4555 5507 6800 7072 7150 7269 7586 8463 8695 9252 9724

-- program is finished running --
```



Tổng số lệnh là: 6815

- R-type: 1717
- I-type: 3680
- J-type: 418

Sử dụng mô hình đơn chu kì.
Thời gian chạy của chương trình $= \frac{CPUClockCycles}{clockRate} = \frac{6815}{2*10^9} = 3705(ns)$