

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC & KỸ THUẬT MÁY TÍNH



KIẾN TRÚC MÁY TÍNH (CO2007)

BÁO CÁO BÀI TẬP LỚN - ĐỀ 5 - QUICK SORT

GVHD: Võ Tấn Phương
Trần Thanh Bình
Lớp: L05 - L09
Thành viên nhóm: 1911900 - Đinh Gia Quang
1911217 - Trần Phạm Thái Hòa
1911478 - Trần Nguyên Lâm



Mục lục

1	Cơ sở lý thuyết	2
1.1	Ý tưởng	2
1.2	Cách chọn pivot	2
1.3	Cách phân vùng (Partition)	2
2	Hiện thực chương trình sắp xếp mảng 20 số nguyên	2
2.1	Chương trình mô phỏng bằng ngôn ngữ C++	2
2.2	Thực thi chương trình trên Mars	4
3	Kết quả thực thi chương trình và thời gian chạy của chương trình	4
4	Tài liệu tham khảo	14

1 Cơ sở lý thuyết

1.1 Ý tưởng

- Thuật toán sắp xếp quick sort là một thuật toán chia để trị (Divide and Conquer algorithm).
- Ý tưởng của quick sort như sau:
 - Chọn một phần tử để so sánh, gọi là phần tử chốt (pivot), từ trong mảng đầu tiên.
 - Phân vùng và sort mảng con trong phân vùng làm sao cho các phần tử lớn hơn từ phần tử pivot nằm sau (bên phải) và các phần tử bé hơn phần tử pivot nằm trước (bên trái). Đây được gọi là quá trình phân vùng (partition).
 - Cuối cùng là đệ quy sử dụng các bước trên cho các mảng với phần tử bé hơn và phân tách với các phần tử lớn hơn sau khi phân vùng.
- Độ phức tạp thuật toán:
 - Tốt: $O(N \log N)$
 - Xấu: $O(N^2)$
 - Trung Bình: $O(N \log N)$
- Không gian bộ nhớ sử dụng: $O(\log N)$

1.2 Cách chọn pivot

Kỹ thuật chọn pivot ảnh hưởng khá nhiều đến khả năng rơi vào các vòng lặp vô hạn đối với các trường hợp đặc biệt. Tốt nhất là chọn pivot là trung vị của danh sách. Khi đó sau $\log_2(N)$ lần phân chia ta sẽ đạt tới kích thước danh sách bằng 1. Tuy nhiên điều đó rất khó. Có các cách chọn phần tử chốt như sau:

- Chọn phần tử đứng đầu hoặc đứng cuối làm phần tử chốt.
- Chọn phần tử đứng giữa danh sách làm phần tử chốt.
- Chọn phần tử trung vị trong 3 phần tử đứng đầu, đứng giữa và đứng cuối làm phần tử chốt.
- Chọn phần tử ngẫu nhiên làm phần tử chốt. (Cách này có thể dẫn đến khả năng rơi vào các trường hợp đặc biệt)

1.3 Cách phân vùng (Partition)

- Một giải pháp đơn giản nhất cho vấn đề này là duyệt từ đầu đến cuối lần lượt so sánh các phần tử của danh sách với phần tử chốt.
- Theo cách này, ta phải tiến hành n phép so sánh, ngoài ra còn phải dành n đơn vị bộ nhớ để lưu giữ các giá trị trung gian.
- Một giải pháp khác được đề nghị là duyệt theo hai đường. Một đường từ đầu danh sách, một đường từ cuối danh sách. Theo cách này, ta tìm phần tử đầu tiên tính từ trái lớn hơn phần tử chốt và phần tử đầu tiên phía phải nhỏ hơn hoặc bằng phần tử chốt rồi đổi chỗ cho nhau. Tiếp tục như vậy cho đến khi hai đường gặp nhau.
- Để có thể gọi đệ quy ta xét bài toán phân chia một danh sách con của $a: a[k_1, k_2]$ thành hai danh sách.

2 Hiện thực chương trình sắp xếp mảng 20 số nguyên

2.1 Chương trình mô phỏng bằng ngôn ngữ C++

```
#include <iostream>
using namespace std;

int* Partition(int* start, int* end){
    int size = end - start;
    int pivot = start[0];
    int i = 1, j = size - 1;

    if (size == 2){
        if (start[0] < start[1]) j = 0;
        else{
            swap(start[0], start[1]);
            j = 1;
        }
    }
    else {
        while (i < j)
        {
            while (start[i] < pivot && i <= size)
                i += 1;
            while (start[j] > pivot && j > 0)
                j -= 1;
            if (i < j) swap(start[i], start[j]);
        }
        if (i > j) swap(start[0], start[j]);
    }
    return start + j;
}

void QuickSort(int* start, int* end){
    static int count = 0;
    int size = end - start;
    if (end - start == 0) return;

    cout << "Partition lan " << ++count << ": ";

    int* pivot = Partition(start, end);

    for (int i = 0; i < size; i++)
        cout << start[i] << " ";
    cout << endl;
    cout << "PivotIndex: " << pivot - start << endl;

    QuickSort(start, pivot);
    QuickSort(pivot + 1, end);
}

int main(){
    int array[] = { 19, 17, 18, 16, 13, 14, 12, 13, 11, 10, 9, 5, 6, 8 , 7, 4, 3, 1 , 2, 0 };
    cout << "Mang cac phan tu ban dau la: " ;
    for (int i = 0; i < 20; i++)
        cout << array[i] << " ";
    cout << endl;

    QuickSort(&array[0], &array[20]);
    cout << "Mang cac phan tu sau khi sap xep la: ";
    for (int i : array) cout << i << " ";
    return 0;
}
```

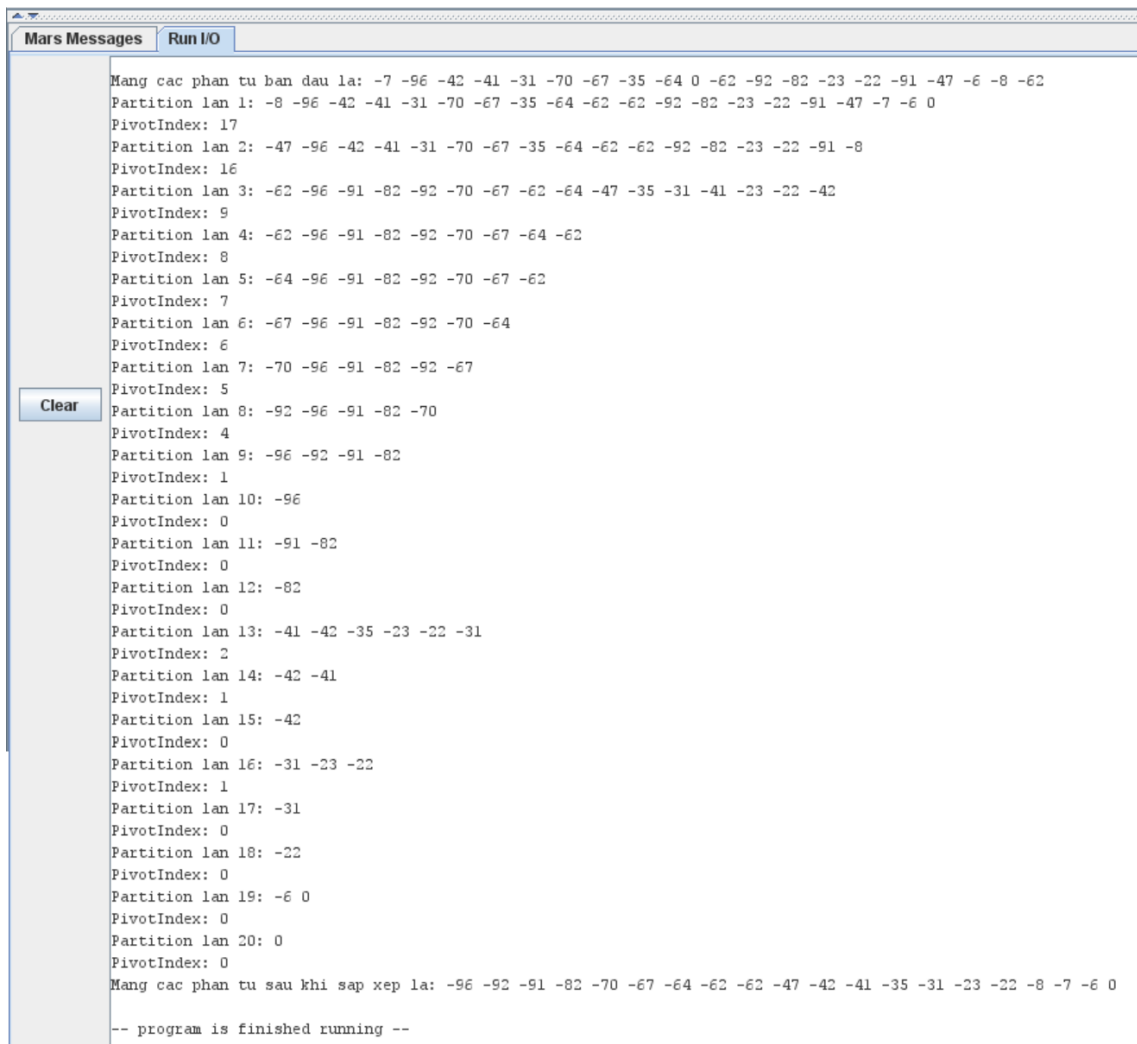
2.2 Thực thi chương trình trên Mars

- File code được đính kèm cùng với bài báo cáo
- Các lệnh thực thi trong chương trình
 - Lệnh R-Type: add, sll, jr, srl, sub, slt, sle, and.
 - Lệnh I-Type: la, li, sw, addi, slti, bne, lw, beq, move, beqz.
 - Lệnh J-Type: jal, j.

3 Kết quả thực thi chương trình và thời gian chạy của chương trình

1. Test case 1

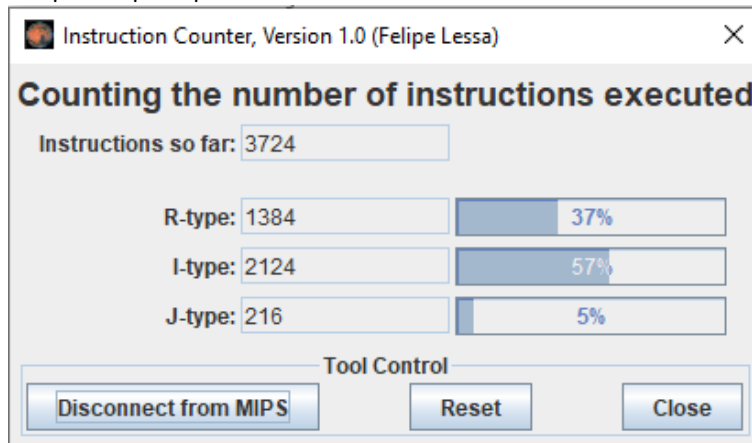
Dãy nhập vào là: -7, -96, -42, -41, -31, -70, -67, -35, -64, 0, -62, -92, -82, -23, -22, -91, -47, -6, -8, -62.
Kết quả:



```
Mang cac phan tu ban dau la: -7 -96 -42 -41 -31 -70 -67 -35 -64 0 -62 -92 -82 -23 -22 -91 -47 -6 -8 -62
Partition lan 1: -8 -96 -42 -41 -31 -70 -67 -35 -64 -62 -92 -82 -23 -22 -91 -47 -7 -6 0
PivotIndex: 17
Partition lan 2: -47 -96 -42 -41 -31 -70 -67 -35 -64 -62 -92 -82 -23 -22 -91 -8
PivotIndex: 16
Partition lan 3: -62 -96 -91 -82 -92 -70 -67 -62 -64 -47 -35 -31 -41 -23 -22 -42
PivotIndex: 9
Partition lan 4: -62 -96 -91 -82 -92 -70 -67 -64 -62
PivotIndex: 8
Partition lan 5: -64 -96 -91 -82 -92 -70 -67 -62
PivotIndex: 7
Partition lan 6: -67 -96 -91 -82 -92 -70 -64
PivotIndex: 6
Partition lan 7: -70 -96 -91 -82 -92 -67
PivotIndex: 5
Partition lan 8: -92 -96 -91 -82 -70
PivotIndex: 4
Partition lan 9: -96 -92 -91 -82
PivotIndex: 1
Partition lan 10: -96
PivotIndex: 0
Partition lan 11: -91 -82
PivotIndex: 0
Partition lan 12: -82
PivotIndex: 0
Partition lan 13: -41 -42 -35 -23 -22 -31
PivotIndex: 2
Partition lan 14: -42 -41
PivotIndex: 1
Partition lan 15: -42
PivotIndex: 0
Partition lan 16: -31 -23 -22
PivotIndex: 1
Partition lan 17: -31
PivotIndex: 0
Partition lan 18: -22
PivotIndex: 0
Partition lan 19: -6 0
PivotIndex: 0
Partition lan 20: 0
PivotIndex: 0
Mang cac phan tu sau khi sap xep la: -96 -92 -91 -82 -70 -67 -64 -62 -62 -47 -42 -41 -35 -31 -23 -22 -8 -7 -6 0

-- program is finished running --
```

Số lệnh được thực thi:



$$\text{Execution Time} = \frac{\text{InstructionCount} * \text{CPI}}{\text{ClockRate}} = \frac{3724 * 1}{2 * 10^9} = 1.862 * 10^{-6} s$$

2. Test case 2

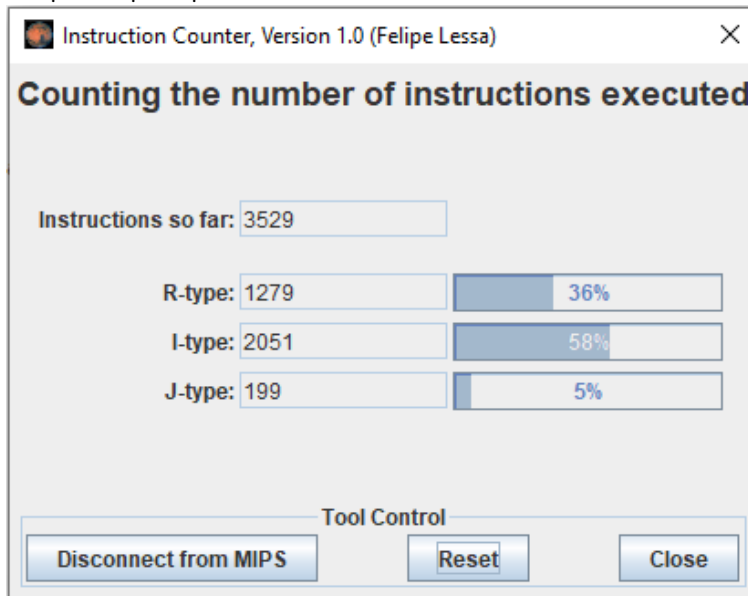
Dãy nhập vào là: -35, 27, 47, 17, 36, -28, -4, 36, -40, 7, 45, -22, 9, 25, 46, 11, 12, -35, -19, -41

```

Mars Messages  Run I/O
Mang cac phan tu ban dau la: -35 27 47 17 36 -28 -4 36 -40 7 45 -22 9 25 46 11 12 -35 -19 -41
Partition lan 1: -35 -41 -40 -35 36 -28 -4 36 17 7 45 -22 9 25 46 11 12 47 -19 27
PivotIndex: 3
Partition lan 2: -40 -41 -35
PivotIndex: 2
Partition lan 3: -41 -40
PivotIndex: 1
Partition lan 4: -41
PivotIndex: 0
Partition lan 5: 36 -28 -4 27 17 7 -19 -22 9 25 12 11 36 47 46 45
PivotIndex: 12
Partition lan 6: 11 -28 -4 27 17 7 -19 -22 9 25 12 36
PivotIndex: 11
Partition lan 7: -19 -28 -4 9 -22 7 11 17 27 25 12
PivotIndex: 6
Partition lan 8: -22 -28 -19 9 -4 7
PivotIndex: 2
Partition lan 9: -28 -22
PivotIndex: 1
Partition lan 10: -28
PivotIndex: 0
Partition lan 11: 7 -4 9
PivotIndex: 2
Partition lan 12: -4 7
PivotIndex: 1
Partition lan 13: -4
PivotIndex: 0
Partition lan 14: 12 17 25 27
PivotIndex: 1
Partition lan 15: 12
PivotIndex: 0
Partition lan 16: 25 27
PivotIndex: 0
Partition lan 17: 27
PivotIndex: 0
Partition lan 18: 45 46 47
PivotIndex: 2
Partition lan 19: 45 46
PivotIndex: 0
Partition lan 20: 46
PivotIndex: 0
Mang cac phan tu sau khi sap xep la: -41 -40 -35 -35 -28 -22 -19 -4 7 9 11 12 17 25 27 36 36 45 46 47
-- program is finished running --

```

Số lệnh được thực thi:



$$\text{Execution Time} = \frac{\text{InstructionCount} * \text{CPI}}{\text{ClockRate}} = \frac{3529 * 1}{2 * 10^9} = 1.7645 * 10^{-6} s$$

3. Test case 3

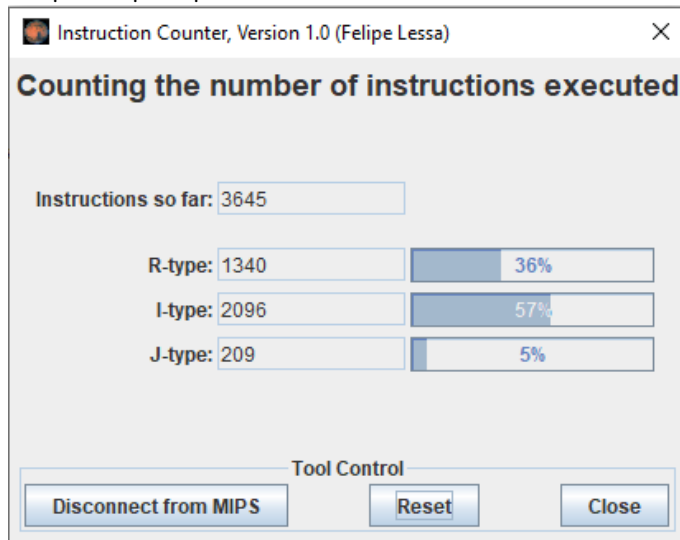
Dãy nhập vào là: 15, 24, -33, -12, 14, 56, 130, 209, 89, 421, 378, 14, 31, 95, 27, 1, 3, 31, 9, 10.

```

Mars Messages  Run I/O
Mang cac phan tu ban dau la: 15 24 -33 -12 14 56 130 209 89 421 378 14 31 95 27 1 3 31 9 10
Partition lan 1: 14 10 -33 -12 14 9 3 1 15 421 378 89 31 95 27 209 130 31 56 24
PivotIndex: 8
Partition lan 2: 14 10 -33 -12 1 9 3 14
PivotIndex: 7
Partition lan 3: 3 10 -33 -12 1 9 14
PivotIndex: 6
Partition lan 4: -12 1 -33 3 10 9
PivotIndex: 3
Partition lan 5: -33 -12 1
PivotIndex: 1
Partition lan 6: -33
PivotIndex: 0
Partition lan 7: 1
PivotIndex: 0
Partition lan 8: 9 10
PivotIndex: 1
Partition lan 9: 9
PivotIndex: 0
Partition lan 10: 24 378 89 31 95 27 209 130 31 56 421
PivotIndex: 10
Partition lan 11: 24 378 89 31 95 27 209 130 31 56
PivotIndex: 0
Partition lan 12: 56 89 31 95 27 209 130 31 378
PivotIndex: 8
Partition lan 13: 27 31 31 56 95 209 130 89
PivotIndex: 3
Partition lan 14: 27 31 31
PivotIndex: 0
PivotIndex: 1
Partition lan 16: 31
PivotIndex: 0
Partition lan 17: 89 95 130 209
PivotIndex: 1
Partition lan 18: 89
PivotIndex: 0
Partition lan 19: 130 209
PivotIndex: 0
Partition lan 20: 209
PivotIndex: 0
Mang cac phan tu sau khi sap xep la: -33 -12 1 3 9 10 14 14 15 24 27 31 31 56 89 95 130 209 378 421
-- program is finished running --

```

Số lệnh được thực thi:



$$\text{Execution Time} = \frac{\text{InstructionCount} * \text{CPI}}{\text{ClockRate}} = \frac{3645 * 1}{2 * 10^9} = 1.8225 * 10^{-6} s$$

4. Test case 4

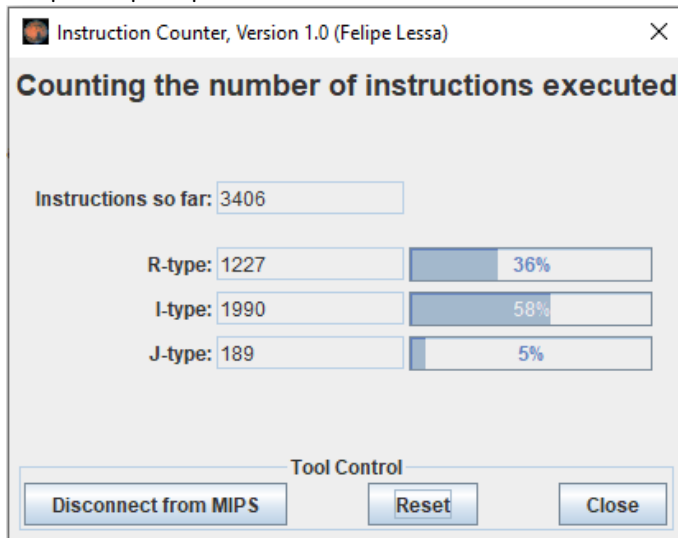
Dãy nhập vào là: 16, 29, -21, 43, -8, -7, 69, -9, 16, -18, 51, 46, 6, 10, 3, 64, 21, 12, -8, 25

```

Mars Messages  Run I/O
Mang cac phan tu ban dau la: 16 29 -21 43 -8 -7 69 -9 16 -18 51 46 6 10 3 64 21 12 -8 25
Partition lan 1: 16 -8 -21 12 -8 -7 3 -9 10 -18 6 16 46 51 69 64 21 43 29 25
PivotIndex: 11
Partition lan 2: 6 -8 -21 12 -8 -7 3 -9 10 -18 16
PivotIndex: 10
Partition lan 3: -9 -8 -21 -18 -8 -7 3 6 10 12
PivotIndex: 7
Partition lan 4: -21 -18 -9 -8 -8 -7 3
PivotIndex: 2
Partition lan 5: -21 -18
PivotIndex: 0
Partition lan 6: -18
PivotIndex: 0
Partition lan 7: -8 -8 -7 3
PivotIndex: 1
Partition lan 8: -8
PivotIndex: 0
Partition lan 9: -7 3
PivotIndex: 0
Partition lan 10: 3
PivotIndex: 0
Partition lan 11: 10 12
PivotIndex: 0
Partition lan 12: 12
PivotIndex: 0
Partition lan 13: 21 25 29 43 46 64 69 51
PivotIndex: 4
Partition lan 14: 21 25 29 43
PivotIndex: 0
Partition lan 15: 25 29 43
PivotIndex: 0
Partition lan 16: 29 43
PivotIndex: 0
Partition lan 17: 43
PivotIndex: 0
Partition lan 18: 51 64 69
PivotIndex: 1
Partition lan 19: 51
PivotIndex: 0
Partition lan 20: 69
PivotIndex: 0
Mang cac phan tu sau khi sap xep la: -21 -18 -9 -8 -8 -7 3 6 10 12 16 16 21 25 29 43 46 51 64 69
-- program is finished running --

```


Số lệnh được thực thi:



$$\text{Execution Time} = \frac{\text{InstructionCount} * \text{CPI}}{\text{ClockRate}} = \frac{3406 * 1}{2 * 10^9} = 1.703 * 10^{-6} s$$

5. Test case 5

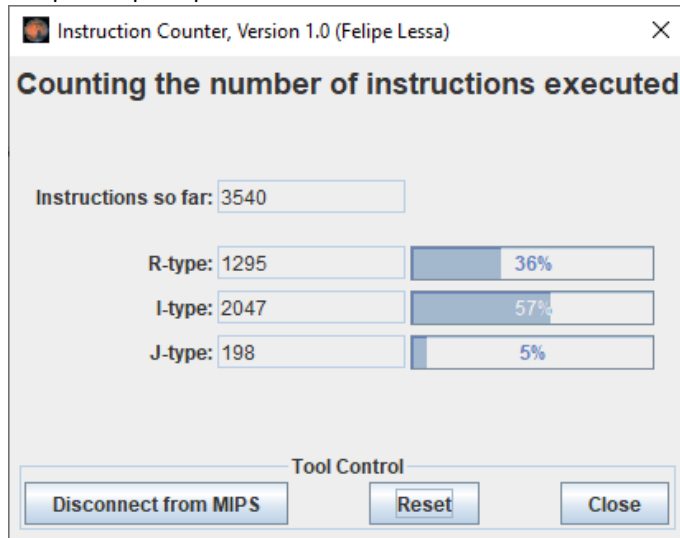
Dãy nhập vào là: 25, 58, 19, -36, -9, 1, -34, -17, -23, 19, 55, -28, 16, -39, -12, 3, -1, -29, -36, 28

```

Mars Messages  Run I/O
Mang cac phan tu ban dau la: 25 58 19 -36 -9 1 -34 -17 -23 19 55 -28 16 -39 -12 3 -1 -29 -36 28
Partition lan 1: -1 -36 19 -36 -9 1 -34 -17 -23 19 -29 -28 16 -39 -12 3 25 55 58 28
PivotIndex: 16
Partition lan 2: -29 -36 -12 -36 -9 -39 -34 -17 -23 -28 -1 19 16 1 19 3
PivotIndex: 10
Partition lan 3: -39 -36 -34 -36 -29 -9 -12 -17 -23 -28
PivotIndex: 4
Partition lan 4: -39 -36 -34 -36
PivotIndex: 0
Partition lan 5: -36 -36 -34
PivotIndex: 1
Partition lan 6: -36
PivotIndex: 0
Partition lan 7: -34
PivotIndex: 0
Partition lan 8: -28 -12 -17 -23 -9
PivotIndex: 4
Partition lan 9: -28 -12 -17 -23
PivotIndex: 0
Partition lan 10: -23 -17 -12
PivotIndex: 2
Partition lan 11: -23 -17
PivotIndex: 0
Partition lan 12: -17
PivotIndex: 0
Partition lan 13: 19 16 1 3 19
PivotIndex: 4
Partition lan 14: 3 16 1 19
PivotIndex: 3
Partition lan 15: 1 3 16
PivotIndex: 1
Partition lan 16: 1
PivotIndex: 0
Partition lan 17: 16
PivotIndex: 0
Partition lan 18: 28 55 58
PivotIndex: 1
Partition lan 19: 28
PivotIndex: 0
Partition lan 20: 58
PivotIndex: 0
Mang cac phan tu sau khi sap xep la: -39 -36 -36 -34 -29 -28 -23 -17 -12 -9 -1 1 3 16 19 19 25 28 55
-- program is finished running --

```

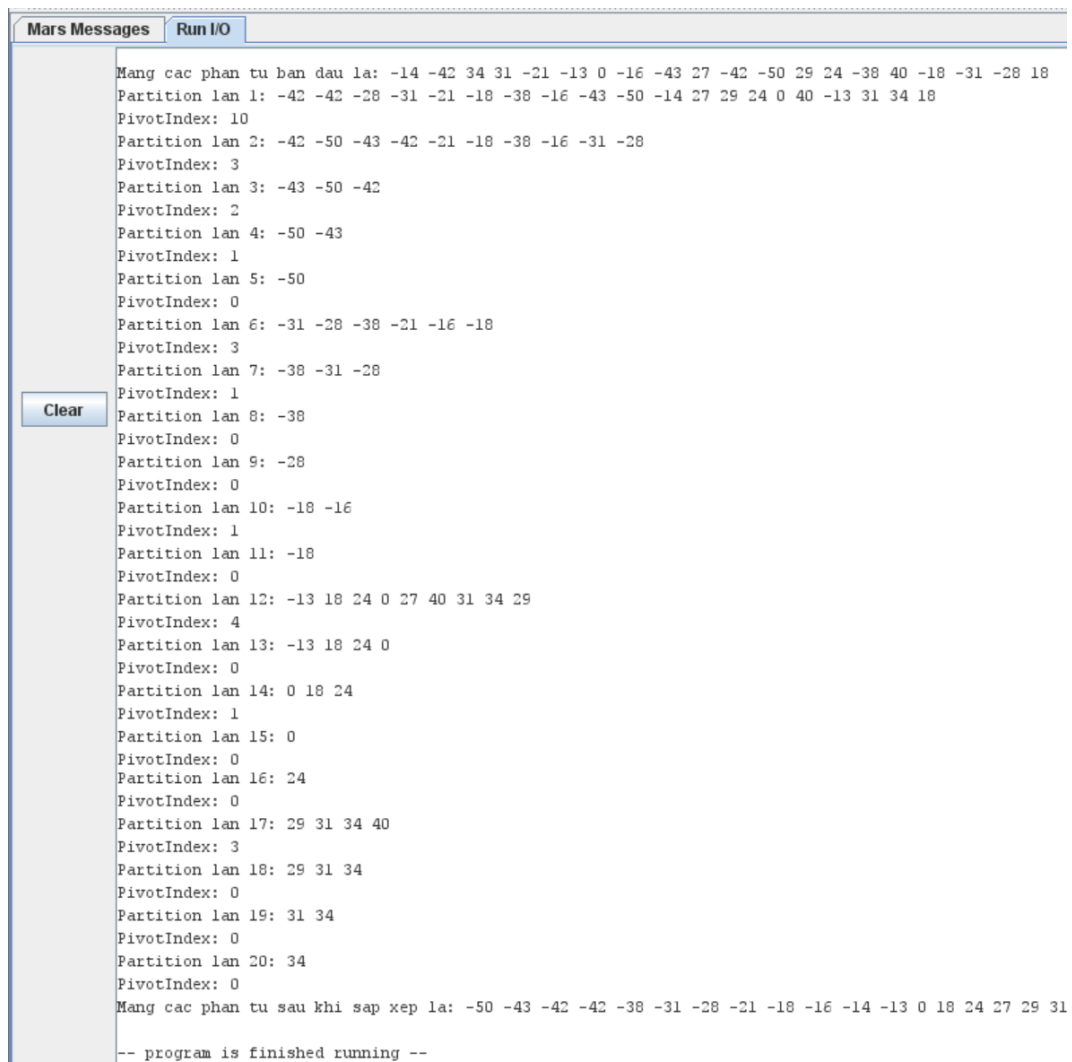
Số lệnh được thực thi:



$$\text{Execution Time} = \frac{\text{InstructionCount} * \text{CPI}}{\text{ClockRate}} = \frac{3540 * 1}{2 * 10^9} = 1.77 * 10^{-6} s$$

6. Test case 6

Dãy nhập vào là: -14, -42, 34, 31, -21, -13, 0, -16, -43, 27, -42, -50, 29, 24, -38, 40, -18, -31, -28, 18

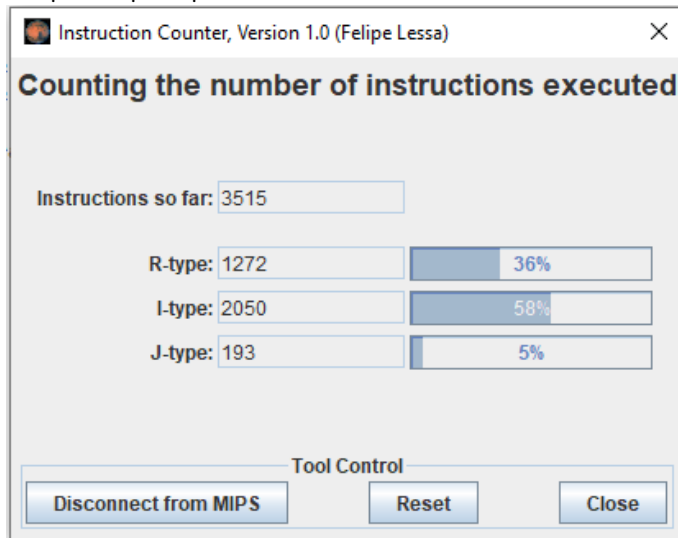


```

Mang cac phan tu ban dau la: -14 -42 34 31 -21 -13 0 -16 -43 27 -42 -50 29 24 -38 40 -18 -31 -28 18
Partition lan 1: -42 -42 -28 -31 -21 -18 -38 -16 -43 -50 -14 27 29 24 0 40 -13 31 34 18
PivotIndex: 10
Partition lan 2: -42 -50 -43 -42 -21 -18 -38 -16 -31 -28
PivotIndex: 3
Partition lan 3: -43 -50 -42
PivotIndex: 2
Partition lan 4: -50 -43
PivotIndex: 1
Partition lan 5: -50
PivotIndex: 0
Partition lan 6: -31 -28 -38 -21 -16 -18
PivotIndex: 3
Partition lan 7: -38 -31 -28
PivotIndex: 1
Partition lan 8: -38
PivotIndex: 0
Partition lan 9: -28
PivotIndex: 0
Partition lan 10: -18 -16
PivotIndex: 1
Partition lan 11: -18
PivotIndex: 0
Partition lan 12: -13 18 24 0 27 40 31 34 29
PivotIndex: 4
Partition lan 13: -13 18 24 0
PivotIndex: 0
Partition lan 14: 0 18 24
PivotIndex: 1
Partition lan 15: 0
PivotIndex: 0
Partition lan 16: 24
PivotIndex: 0
Partition lan 17: 29 31 34 40
PivotIndex: 3
Partition lan 18: 29 31 34
PivotIndex: 0
Partition lan 19: 31 34
PivotIndex: 0
Partition lan 20: 34
PivotIndex: 0
Mang cac phan tu sau khi sap xep la: -50 -43 -42 -42 -38 -31 -28 -21 -18 -16 -14 -13 0 18 24 27 29 31
-- program is finished running --

```

Số lệnh được thực thi:



$$\text{Execution Time} = \frac{\text{InstructionCount} * \text{CPI}}{\text{ClockRate}} = \frac{3515 * 1}{2 * 10^9} = 1.7575 * 10^{-6} s$$

7. Test case 7

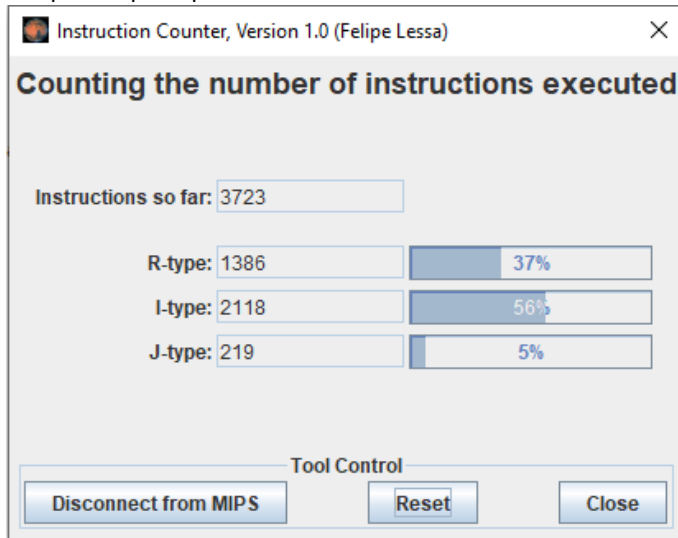
Dãy nhập vào là: -19, 32, 10, 8, 9, 34, 46, 16, -27, -23, 46, 2, -10, 42, 38, -29, 4, 39, 23, 29

```

Mars Messages Run I/O
Mang cac phan tu ban dau la: -19 32 10 8 9 34 46 16 -27 -23 46 2 -10 42 38 -29 4 39 23 29
Partition lan 1: -27 -29 -23 -19 9 34 46 16 8 10 46 2 -10 42 38 32 4 39 23 29
PivotIndex: 3
Partition lan 2: -29 -27 -23
PivotIndex: 1
Partition lan 3: -29
PivotIndex: 0
Partition lan 4: -23
PivotIndex: 0
Partition lan 5: 8 4 -10 2 9 10 46 16 46 42 38 32 34 39 23 29
PivotIndex: 4
Partition lan 6: 2 4 -10 8
PivotIndex: 3
Partition lan 7: -10 2 4
PivotIndex: 1
Partition lan 8: -10
PivotIndex: 0
Partition lan 9: 4
PivotIndex: 0
Partition lan 10: 10 46 16 46 42 38 32 34 39 23 29
PivotIndex: 0
Partition lan 11: 46 16 29 42 38 32 34 39 23 46
PivotIndex: 9
Partition lan 12: 23 16 29 42 38 32 34 39 46
PivotIndex: 8
Partition lan 13: 16 23 29 42 38 32 34 39
PivotIndex: 1
Partition lan 14: 16
PivotIndex: 0
Partition lan 15: 29 42 38 32 34 39
PivotIndex: 0
Partition lan 16: 39 38 32 34 42
PivotIndex: 4
Partition lan 17: 34 38 32 39
PivotIndex: 3
Partition lan 18: 32 34 38
PivotIndex: 1
Partition lan 19: 32
PivotIndex: 0
Partition lan 20: 38
PivotIndex: 0
Mang cac phan tu sau khi sap xep la: -29 -27 -23 -19 -10 2 4 8 9 10 16 23 29 32 34 38 39 42 46 46
-- program is finished running --

```

Số lệnh được thực thi:



$$\text{Execution Time} = \frac{\text{InstructionCount} * \text{CPI}}{\text{ClockRate}} = \frac{3723 * 1}{2 * 10^9} = 1.8615 * 10^{-6} s$$

8. Test case 8

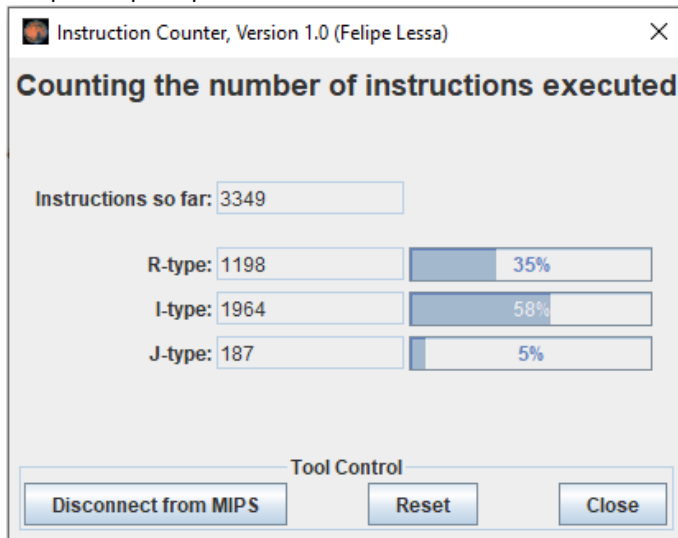
Dãy nhập vào là: 23, 12, 42, 36, 8, 18, 35, 10, 10, 48, 11, 6, 29, 29, 7, 12, 49, 2, 2, 30

```

Mars Messages Run I/O
Mang cac phan tu ban dau la: 23 12 42 36 8 18 35 10 10 48 11 6 29 29 7 12 49 2 2 30
Partition lan 1: 6 12 2 2 8 18 12 10 10 7 11 23 29 29 48 35 49 36 42 30
PivotIndex: 11
Partition lan 2: 2 2 6 12 8 18 12 10 10 7 11
PivotIndex: 2
Partition lan 3: 2 2
PivotIndex: 1
Partition lan 4: 2
PivotIndex: 0
Partition lan 5: 12 8 11 7 10 10 12 18
PivotIndex: 6
Partition lan 6: 10 8 11 7 10 12
PivotIndex: 5
Partition lan 7: 10 8 7 10 11
PivotIndex: 3
Partition lan 8: 7 8 10
PivotIndex: 2
Partition lan 9: 7 8
PivotIndex: 0
Partition lan 10: 8
PivotIndex: 0
Partition lan 11: 11
PivotIndex: 0
Partition lan 12: 18
PivotIndex: 0
Partition lan 13: 29 29 48 35 49 36 42 30
PivotIndex: 1
Partition lan 14: 29
PivotIndex: 0
Partition lan 15: 42 35 30 36 48 49
PivotIndex: 4
Partition lan 16: 36 35 30 42
PivotIndex: 3
Partition lan 17: 30 35 36
PivotIndex: 2
Partition lan 18: 30 35
PivotIndex: 0
Partition lan 19: 35
PivotIndex: 0
Partition lan 20: 49
PivotIndex: 0
Mang cac phan tu sau khi sap xep la: 2 2 6 7 8 10 10 11 12 12 18 23 29 29 30 35 36 42 48 49
-- program is finished running --

```

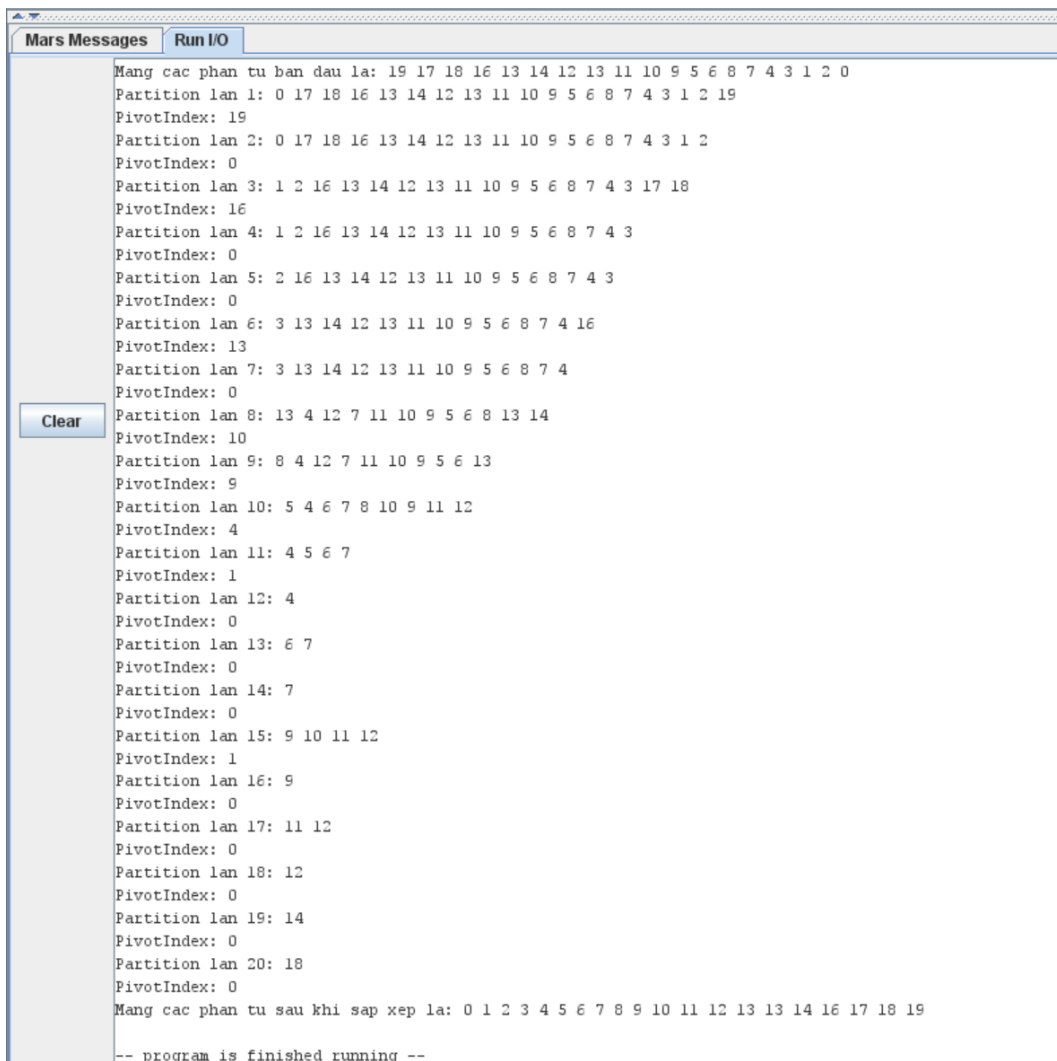
Số lệnh được thực thi:



$$\text{Execution Time} = \frac{\text{InstructionCount} * \text{CPI}}{\text{ClockRate}} = \frac{3349 * 1}{2 * 10^9} = 1.6745 * 10^{-6} s$$

9. Test case 9

Dãy nhập vào là: 19, 17, 18, 16, 13, 14, 12, 13, 11, 10, 9, 5, 6, 8, 7, 4, 3, 1, 2, 0

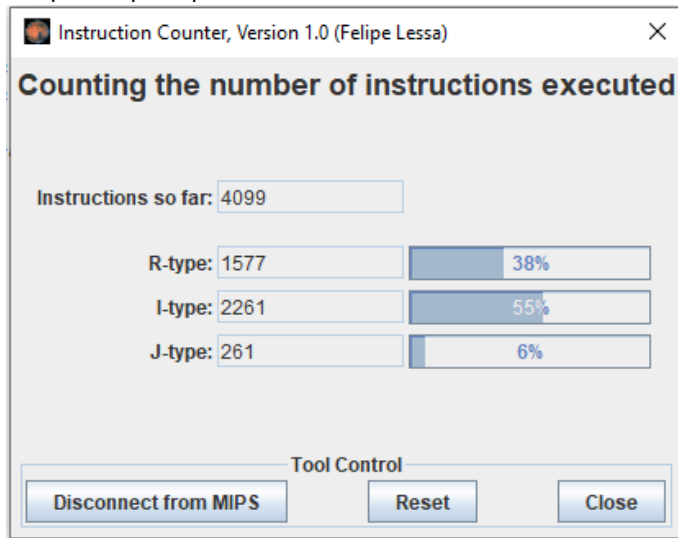


```

Mars Messages Run I/O
Mang cac phan tu ban dau la: 19 17 18 16 13 14 12 13 11 10 9 5 6 8 7 4 3 1 2 0
Partition lan 1: 0 17 18 16 13 14 12 13 11 10 9 5 6 8 7 4 3 1 2 19
PivotIndex: 19
Partition lan 2: 0 17 18 16 13 14 12 13 11 10 9 5 6 8 7 4 3 1 2
PivotIndex: 0
Partition lan 3: 1 2 16 13 14 12 13 11 10 9 5 6 8 7 4 3 17 18
PivotIndex: 16
Partition lan 4: 1 2 16 13 14 12 13 11 10 9 5 6 8 7 4 3
PivotIndex: 0
Partition lan 5: 2 16 13 14 12 13 11 10 9 5 6 8 7 4 3
PivotIndex: 0
Partition lan 6: 3 13 14 12 13 11 10 9 5 6 8 7 4 16
PivotIndex: 13
Partition lan 7: 3 13 14 12 13 11 10 9 5 6 8 7 4
PivotIndex: 0
Partition lan 8: 13 4 12 7 11 10 9 5 6 8 13 14
PivotIndex: 10
Partition lan 9: 8 4 12 7 11 10 9 5 6 13
PivotIndex: 9
Partition lan 10: 5 4 6 7 8 10 9 11 12
PivotIndex: 4
Partition lan 11: 4 5 6 7
PivotIndex: 1
Partition lan 12: 4
PivotIndex: 0
Partition lan 13: 6 7
PivotIndex: 0
Partition lan 14: 7
PivotIndex: 0
Partition lan 15: 9 10 11 12
PivotIndex: 1
Partition lan 16: 9
PivotIndex: 0
Partition lan 17: 11 12
PivotIndex: 0
Partition lan 18: 12
PivotIndex: 0
Partition lan 19: 14
PivotIndex: 0
Partition lan 20: 18
PivotIndex: 0
Mang cac phan tu sau khi sap xep la: 0 1 2 3 4 5 6 7 8 9 10 11 12 13 13 14 16 17 18 19
-- program is finished running --

```

Số lệnh được thực thi:



$$\text{Execution Time} = \frac{\text{InstructionCount} * \text{CPI}}{\text{ClockRate}} = \frac{4099 * 1}{2 * 10^9} = 2.0495 * 10^{-6} s$$

10. Test case 10

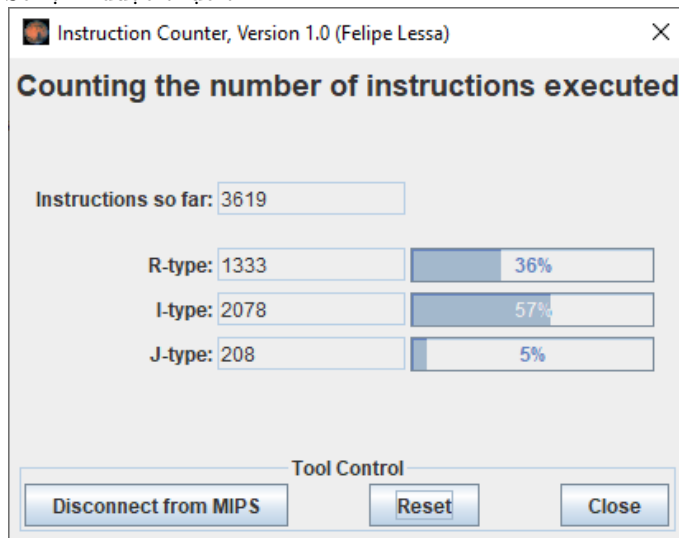
Dãy nhập vào là: 56, 0, 56, 33, 14, -3, -2, 20, 36, 53, 36, 26, 52, 25, 0, 50, 29, 87, 79, -7

```

Mars Messages  Run I/O
Mang cac phan tu ban dau la: 56 0 56 33 14 -3 -2 20 36 53 36 26 52 25 0 50 29 87 79 -7
Partition lan 1: 56 0 -7 33 14 -3 -2 20 36 53 36 26 52 25 0 50 29 56 79 87
PivotIndex: 17
Partition lan 2: 29 0 -7 33 14 -3 -2 20 36 53 36 26 52 25 0 50 56
PivotIndex: 16
Partition lan 3: 26 0 -7 0 14 -3 -2 20 25 29 36 53 52 36 33 50
PivotIndex: 9
Partition lan 4: 25 0 -7 0 14 -3 -2 20 26
PivotIndex: 8
Partition lan 5: 20 0 -7 0 14 -3 -2 25
PivotIndex: 7
Partition lan 6: -2 0 -7 0 14 -3 20
PivotIndex: 6
Partition lan 7: -7 -3 -2 0 14 0
PivotIndex: 2
Partition lan 8: -7 -3
PivotIndex: 0
Partition lan 9: -3
PivotIndex: 0
Partition lan 10: 0 0 14
PivotIndex: 1
Partition lan 11: 0
PivotIndex: 0
Partition lan 12: 14
PivotIndex: 0
Partition lan 13: 36 33 36 52 53 50
PivotIndex: 2
Partition lan 14: 33 36
PivotIndex: 1
Partition lan 15: 33
PivotIndex: 0
Partition lan 16: 50 52 53
PivotIndex: 1
Partition lan 17: 50
PivotIndex: 0
Partition lan 18: 53
PivotIndex: 0
Partition lan 19: 79 87
PivotIndex: 0
Partition lan 20: 87
PivotIndex: 0
Mang cac phan tu sau khi sap xep la: -7 -3 -2 0 0 14 20 25 26 29 33 36 36 50 52 53 56 56 79 87
-- program is finished running --

```

Số lệnh được thực thi:



$$\text{Execution Time} = \frac{\text{InstructionCount} * \text{CPI}}{\text{ClockRate}} = \frac{3619 * 1}{2 * 10^9} = 1.8095 * 10^{-6} s$$

4 Tài liệu tham khảo

- (1) <https://en.wikipedia.org/wiki/Quicksort>
- (2) <https://nguyenvanhieu.vn/thuat-toan-sap-xep-quick-sort/>