

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
Khoa Khoa học và Kỹ thuật Máy tính



KIẾN TRÚC MÁY TÍNH

BÀI TẬP LỚN

GVHD:	Trần Thanh Bình	
SV thực hiện:	Nguyễn Phùng Hải Đăng	1910123
	Đặng Nguyễn Xuân Nam	1910351
	Lê Hữu Đức Minh	1910339

Tp. Hồ Chí Minh, Tháng 12/2020

Mục lục

1	Yêu cầu	2
2	Bài làm	2
2.1	Hiện thực bằng ngôn ngữ C++	2
2.2	Hiện thực bằng hợp ngữ MIPS	3
2.2.1	Ý tưởng tổng quát	3
2.2.2	Vùng dữ liệu .data	3
2.2.3	Hàm đọc dữ liệu đầu vào: readInput	3
2.2.4	Hàm xuất dữ liệu: printOutput	3
2.2.5	Hàm partition	3
2.2.6	Hàm quick sort	3
2.2.7	Hàm main	3
2.2.8	Quá trình thực hiện	4
2.3	Thực hiện các thao tác tính toán:	4
2.3.1	Thống kê số lệnh, loại lệnh được sử dụng	4
2.3.2	Tính và trình bày cách tính thời gian chạy của chương trình	8

1 Yêu cầu

1. Sử dụng tập lệnh MIPS để thực thi các thủ tục bên dưới.
2. Thống kê số lệnh, loại lệnh của chương trình bên dưới.
3. Tính và trình bày các tính thời gian chạy trên máy tính MIPS có tần số 2GHz.
4. Code:
 - (a) Code style phải rõ ràng, có comment, phân hoạch công việc theo từng hàm.
 - (b) Truyền nhận và trả kết quả gọi hàm theo quy ước sử dụng thanh ghi ($\$a0 \rightarrow \$a3$ cho argument, $\$v0 \rightarrow \$v1$ cho kết quả trả về)
 - (c) Xuất kết quả để kiểm tra (sử dụng các hàm hệ thống)
5. Các hình thức gian lận nếu bị phát hiện sẽ nhận điểm 0
6. DEADLINE: 23:55 20/11/2020

Đề 5:

1. Sắp xếp chuỗi
2. 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 Quick sort. Yêu cầu xuất ra từng bước trong quá trình demo.

2 Bài làm

2.1 Hiện thực bằng ngôn ngữ C++

```
int partition(int* arr, int low, int high)
{
    int pivot = low;
    int i = low + 1;
    int j = high;
    while (i <= j)
    {
        while (i <= j && arr[i] < arr[pivot]) i++;
        while (i <= j && arr[j] > arr[pivot]) j--;
        if (i <= j) swap(arr[i], arr[j]);
    }
    swap(arr[j], arr[pivot]);
    return j;
}

void quickSort(int* arr, int low, int high)
{
    if (low < high)
    {
        int q = partition(arr, low, high);
        quickSort(arr, low, q - 1);
        quickSort(arr, q + 1, high);
    }
}
```

2.2 Hiện thực bằng hợp ngữ MIPS

2.2.1 Ý tưởng tổng quát

Ngoài hai hàm partition và quickSort thì ta cần hiện thực thêm hàm nhập vào 20 số nguyên và hàm xuất mảng sao mỗi lần thực hiện quick sort.

2.2.2 Vùng dữ liệu .data

1. arr: nơi lưu mảng số nguyên 20 phần tử.
2. size: giá trị của kích cỡ mảng
3. mess1, mess2, mess3, mess4: các message được in ra màn hình để người dùng dễ dàng theo dõi chương trình.

2.2.3 Hàm đọc dữ liệu đầu vào: readInput

Ý tưởng: Sử dụng vòng lặp để đọc giá trị người dùng nhập lần lượt từng giá trị vào và lưu các giá trị này vào arr ở vùng .data. Để thực hiện, ta sử dụng lệnh rẽ nhánh kết hợp với syscall để đọc và ghi dữ liệu vào arr.

2.2.4 Hàm xuất dữ liệu: printOutput

Ý tưởng: Tương tự phương thức nhập dữ liệu, in dữ liệu cũng được hiện thực bằng vòng lặp. Để hiện thực, ta cũng sử dụng lệnh rẽ nhánh kết hợp với syscall để in dữ liệu của arr ra màn hình. Tuy nhiên, vì đây là hàm được gọi trong lúc đệ quy quick sort nên ta cần phải đẩy các giá trị vào stack để tránh việc thất thoát địa chỉ cũng như dữ liệu. Sau mỗi bước quicksort, hàm printOutput sẽ in ra dãy gồm 20 phần tử sau mỗi lần sắp xếp bằng quicksort.

2.2.5 Hàm partition

Ý tưởng: Hàm partition được hiện thực dựa trên giải thuật quen thuộc trên code C++. Cụ thể hơn, ta sẽ thực hiện việc lấy pivot (ở đây ta lấy pivot là phần tử cuối cùng của dãy cần xét). Sau đó, thực hiện phép so sánh các phần tử trong dãy với pivot để thực hiện việc đổi chỗ 2 phần tử nếu thỏa mãn điều kiện. Việc đổi chỗ được hiện thực bằng hàm swap, dùng thay đổi vị trí của 2 phần tử trong dãy. Để hiện thực hàm partition, ta dùng các lệnh rẽ nhánh và so sánh, đồng thời dùng jal để gọi hàm swap khi cần dùng; để hiện thực hàm swap, ta sử dụng lệnh lw và sw để thay đổi vị trí hai phần tử cần thay đổi vị trí trong arr.

2.2.6 Hàm quick sort

Ý tưởng: Cũng như partition, hàm quick sort được hiện thực dựa trên giải thuật quen thuộc trên code C++. Cụ thể hơn, hàm quicksort sẽ gọi ra hàm partition, từ đó đổi chỗ một số phần tử trong hàm partition. Tiếp đến, tìm ra pivot mới rồi tiếp tục đệ quy hàm quicksort trên 2 dãy con tạo ra bằng cách chia dãy đó theo phần tử pivot. Đến cuối cùng, ta sẽ thu được dãy đã sắp xếp. Qua mỗi lần gọi hàm quicksort, ta sẽ gọi hàm printOutput để in ra dãy sau mỗi bước thực hiện. Hiện thực hàm quick sort, ta dùng jal để gọi các hàm partition, sau đó printOutput và tiếp tục đệ quy với hai hàm quicksort trên hai mảng dữ liệu được chia đôi bởi pivot.

2.2.7 Hàm main

Ý tưởng: đây là hàm chính của chương trình. Hàm này sử dụng lệnh jal để gọi ra các hàm readInput, partition và quicksort; từ đó giúp ta thực hiện phương pháp sắp xếp quicksort trên dãy gồm 20 phần tử được người dùng nhập vào theo thứ tự.

2.2.8 Quá trình thực hiện

- Hàm main gọi hàm readInput bằng lệnh jal để người dùng nhập lần lượt 20 phần tử vào. Sau đó, 20 phần tử này sẽ được lưu trong arr theo đúng thứ tự đã được nhập vào.
- Hàm main gọi hàm quicksort bằng lệnh jal để bắt đầu việc dùng phương pháp sắp xếp quicksort sắp xếp 20 phần tử trong arr theo thứ tự từ bé đến lớn.
- Hàm quicksort sẽ kiểm tra xem đã dừng việc quicksort được hay chưa. Việc quicksort sẽ dừng lại khi chỉ số pivot lớn hơn hoặc bằng chỉ số phần tử cuối cùng của phân mảng. Nếu chưa, hàm quicksort sẽ gọi hàm partition để thực hiện việc sắp xếp theo từng bước và lấy pivot; tiếp tục gọi hàm printOutput để in ra mảng 20 phần tử sau mỗi lần sắp xếp; sau đó gọi đệ quy hàm quicksort đối với 2 mảng được chia theo pivot từ mảng lớn đang xét. Nếu đủ điều kiện, quicksort dừng, ta lấy các dữ liệu ra khỏi stack và trở về hàm main, sau đó in ra các phần tử theo thứ tự từ bé đến lớn (tức là hoàn thành việc sắp xếp).
- Trong hàm partition còn có các hàm hỗ trợ dùng để so sánh và đổi chỗ hai phần tử khi cần thiết
- Cuối cùng, trở về hàm main và in ra các phần tử đã được sắp xếp với lệnh printOutput.

2.3 Thực hiện các thao tác tính toán:

Yêu cầu: Máy tính có kiến trúc MIPS có tần số $2GHz$

2.3.1 Thống kê số lệnh, loại lệnh được sử dụng

Để thống kê số loại lệnh được sử dụng, ta sử dụng Tool "Instruction Counter" tích hợp sẵn trong Mars. Dòng cuối cùng trong output của mỗi testcase chính là dãy gồm 20 phần tử được sắp xếp từ bé đến lớn sau khi đã sắp xếp bằng phương pháp Quicksort. **TestCase 1:** Các phần tử được sắp xếp sẵn theo thứ tự giảm dần

Input: 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 1

Output: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20

Quy trình thực hiện:

```
Current array: 1, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 20
Current array: 1, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 2, 20
Current array: 1, 2, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 19, 20
Current array: 1, 2, 18, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 3, 19, 20
Current array: 1, 2, 3, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 18, 19, 20
Current array: 1, 2, 3, 17, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 4, 18, 19, 20
Current array: 1, 2, 3, 4, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 17, 18, 19, 20
Current array: 1, 2, 3, 4, 16, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 5, 17, 18, 19, 20
Current array: 1, 2, 3, 4, 5, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 16, 17, 18, 19, 20
Current array: 1, 2, 3, 4, 5, 15, 14, 13, 12, 11, 10, 9, 8, 7, 6, 16, 17, 18, 19, 20
Current array: 1, 2, 3, 4, 5, 6, 14, 13, 12, 11, 10, 9, 8, 7, 15, 16, 17, 18, 19, 20
Current array: 1, 2, 3, 4, 5, 6, 14, 13, 12, 11, 10, 9, 8, 7, 15, 16, 17, 18, 19, 20
Current array: 1, 2, 3, 4, 5, 6, 7, 13, 12, 11, 10, 9, 8, 14, 15, 16, 17, 18, 19, 20
Current array: 1, 2, 3, 4, 5, 6, 7, 13, 12, 11, 10, 9, 8, 14, 15, 16, 17, 18, 19, 20
Current array: 1, 2, 3, 4, 5, 6, 7, 8, 12, 11, 10, 9, 13, 14, 15, 16, 17, 18, 19, 20
Current array: 1, 2, 3, 4, 5, 6, 7, 8, 12, 11, 10, 9, 13, 14, 15, 16, 17, 18, 19, 20
Current array: 1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 10, 12, 13, 14, 15, 16, 17, 18, 19, 20
Current array: 1, 2, 3, 4, 5, 6, 7, 8, 9, 11, 10, 12, 13, 14, 15, 16, 17, 18, 19, 20
Current array: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20
Current array: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20
```

Có tổng cộng 9186 lệnh được thực hiện. Trong đó:

R-type: 2434 lệnh.

I-type: 5662 lệnh.

J-type: 1090 lệnh.

TestCase 2: Các phần tử được sắp xếp sẵn theo thứ tự tăng dần

Input: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20

Output: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20

Quy trình thực hiện:

[illegible]

Có tổng cộng 9236 lệnh được thực hiện. Trong đó:

R-type: 2464 lệnh.

I-type: 5682 lệnh.

J-type: 1090 lệnh.

TestCase 3: Dãy bao gồm số dương và số âm sắp xếp xen kẽ nhau

Input: 49, 86, 2, 64, -32, 29, 18, -52, -45, 56, -47, 4, 92, 95, 53, -27, 51, 81, 98, 56

Output: -52, -47, -45, -32, -27, 2, 4, 18, 29, 49, 51, 53, 56, 56, 64, 81, 86, 92, 95, 98

Quy trình thực hiện:

Current array:	-47,	-27,	2,	4,	-32,	29,	18,	-52,	-45,	49,	56,	64,	92,	95,	53,	86,	51,	81,	98,	56
Current array:	-52,	-47,	2,	4,	-32,	29,	18,	-27,	-45,	49,	56,	64,	92,	95,	53,	86,	51,	81,	98,	56
Current array:	-52,	-47,	-27,	-45,	-32,	2,	18,	29,	4,	49,	56,	64,	92,	95,	53,	86,	51,	81,	98,	56
Current array:	-52,	-47,	-32,	-45,	-27,	2,	18,	29,	4,	49,	56,	64,	92,	95,	53,	86,	51,	81,	98,	56
Current array:	-52,	-47,	-45,	-32,	-27,	2,	18,	29,	4,	49,	56,	64,	92,	95,	53,	86,	51,	81,	98,	56
Current array:	-52,	-47,	-45,	-32,	-27,	2,	4,	18,	29,	49,	56,	64,	92,	95,	53,	86,	51,	81,	98,	56
Current array:	-52,	-47,	-45,	-32,	-27,	2,	4,	18,	29,	49,	53,	51,	56,	95,	92,	86,	64,	81,	98,	56
Current array:	-52,	-47,	-45,	-32,	-27,	2,	4,	18,	29,	49,	51,	53,	56,	95,	92,	86,	64,	81,	98,	56
Current array:	-52,	-47,	-45,	-32,	-27,	2,	4,	18,	29,	49,	51,	53,	56,	56,	92,	86,	64,	81,	95,	98
Current array:	-52,	-47,	-45,	-32,	-27,	2,	4,	18,	29,	49,	51,	53,	56,	56,	92,	86,	64,	81,	95,	98
Current array:	-52,	-47,	-45,	-32,	-27,	2,	4,	18,	29,	49,	51,	53,	56,	56,	81,	86,	64,	92,	95,	98
Current array:	-52,	-47,	-45,	-32,	-27,	2,	4,	18,	29,	49,	51,	53,	56,	56,	64,	81,	86,	92,	95,	98
Current array:	-52,	-47,	-45,	-32,	-27,	2,	4,	18,	29,	49,	51,	53,	56,	56,	64,	81,	86,	92,	95,	98

Có tổng cộng 5773 lệnh được thực hiện. Trong đó:

R-type: 1503 lệnh.

I-type: 3604 lệnh.

J-type: 666 lệnh.

TestCase 4: Dãy bao gồm cả số dương, số âm và số 0 sắp xếp ngẫu nhiên

Input: 74, 5, 78, 0, -89, 62, -48, 44, -63, 91, 53, -60, -21, -33, 35, 77, 48, 98, 22, -90

Output: -90, -89, -63, -60, -48, -33, -21, 0, 5, 22, 35, 44, 48, 53, 62, 74, 77, 78, 91, 98

Quy trình thực hiện:

Current array:	48,	5,	-90,	0,	-89,	62,	-48,	44,	-63,	22,	53,	-60,	-21,	-33,	35,	74,	77,	98,	91,	78
Current array:	-21,	5,	-90,	0,	-89,	35,	-48,	44,	-63,	22,	-33,	-60,	48,	53,	62,	74,	77,	98,	91,	78
Current array:	-48,	-60,	-90,	-33,	-89,	-63,	-21,	44,	35,	22,	0,	5,	48,	53,	62,	74,	77,	98,	91,	78
Current array:	-89,	-60,	-90,	-63,	-48,	-33,	-21,	44,	35,	22,	0,	5,	48,	53,	62,	74,	77,	98,	91,	78
Current array:	-90,	-89,	-60,	-63,	-48,	-33,	-21,	44,	35,	22,	0,	5,	48,	53,	62,	74,	77,	98,	91,	78
Current array:	-90,	-89,	-63,	-60,	-48,	-33,	-21,	44,	35,	22,	0,	5,	48,	53,	62,	74,	77,	98,	91,	78
Current array:	-90,	-89,	-63,	-60,	-48,	-33,	-21,	5,	35,	22,	0,	44,	48,	53,	62,	74,	77,	98,	91,	78
Current array:	-90,	-89,	-63,	-60,	-48,	-33,	-21,	0,	5,	22,	35,	44,	48,	53,	62,	74,	77,	98,	91,	78
Current array:	-90,	-89,	-63,	-60,	-48,	-33,	-21,	0,	5,	22,	35,	44,	48,	53,	62,	74,	77,	98,	91,	78
Current array:	-90,	-89,	-63,	-60,	-48,	-33,	-21,	0,	5,	22,	35,	44,	48,	53,	62,	74,	77,	98,	91,	78
Current array:	-90,	-89,	-63,	-60,	-48,	-33,	-21,	0,	5,	22,	35,	44,	48,	53,	62,	74,	77,	98,	91,	78
Current array:	-90,	-89,	-63,	-60,	-48,	-33,	-21,	0,	5,	22,	35,	44,	48,	53,	62,	74,	77,	78,	91,	98
Current array:	-90,	-89,	-63,	-60,	-48,	-33,	-21,	0,	5,	22,	35,	44,	48,	53,	62,	74,	77,	78,	91,	98

Có tổng cộng 6194 lệnh được thực hiện. Trong đó:

R-type: 1611 lệnh.

I-type: 3868 lệnh.

J-type: 715 lệnh.

TestCase 5: Dãy bao gồm số dương và số âm sắp xếp xe kẽ nhau

Input: 57, -64, -36, 42, -71, 18, -85, 18, 96, -64, -30, -31, 18, -71, -12, 82, 82, 1, 89, -32

Output: -85, -71, -71, -64, -64, -36, -32, -31, -30, -12, 1, 18, 18, 18, 42, 57, 82, 82, 89, 96

Quy trình thực hiện:

```
Current array: 1, -64, -36, 42, -71, 18, -85, 18, -32, -64, -30, -31, 18, -71, -12, 57, 82, 82, 89, 96
Current array: -30, -64, -36, -12, -71, -71, -85, -31, -32, -64, 1, 18, 18, 18, 42, 57, 82, 82, 89, 96
Current array: -32, -64, -36, -64, -71, -71, -85, -31, -30, -12, 1, 18, 18, 18, 42, 57, 82, 82, 89, 96
Current array: -85, -64, -36, -64, -71, -71, -32, -31, -30, -12, 1, 18, 18, 18, 42, 57, 82, 82, 89, 96
Current array: -85, -64, -36, -64, -71, -71, -32, -31, -30, -12, 1, 18, 18, 18, 42, 57, 82, 82, 89, 96
Current array: -85, -71, -71, -64, -64, -36, -32, -31, -30, -12, 1, 18, 18, 18, 42, 57, 82, 82, 89, 96
Current array: -85, -71, -71, -64, -64, -36, -32, -31, -30, -12, 1, 18, 18, 18, 42, 57, 82, 82, 89, 96
Current array: -85, -71, -71, -64, -64, -36, -32, -31, -30, -12, 1, 18, 18, 18, 42, 57, 82, 82, 89, 96
Current array: -85, -71, -71, -64, -64, -36, -32, -31, -30, -12, 1, 18, 18, 18, 42, 57, 82, 82, 89, 96
Current array: -85, -71, -71, -64, -64, -36, -32, -31, -30, -12, 1, 18, 18, 18, 42, 57, 82, 82, 89, 96
Current array: -85, -71, -71, -64, -64, -36, -32, -31, -30, -12, 1, 18, 18, 18, 42, 57, 82, 82, 89, 96
Current array: -85, -71, -71, -64, -64, -36, -32, -31, -30, -12, 1, 18, 18, 18, 42, 57, 82, 82, 89, 96
```

Có tổng cộng 5354 lệnh được thực hiện. Trong đó:

R-type: 1391 lệnh.

I-type: 3340 lệnh.

J-type: 623 lệnh.

TestCase 6: Dãy bao gồm nhiều số dương và một ít số âm sắp xếp ngẫu nhiên

Input: -180, 149, 140, 196, 57, 36, 38, 49, -24, 183, 103, 47, 38, 172, 124, 139, 2, 74, 21, 88

Output: -180, -24, 2, 21, 36, 38, 38, 47, 49, 57, 74, 88, 103, 124, 139, 140, 149, 172, 183, 196

Quy trình thực hiện:

```
Current array: -180, 149, 140, 196, 57, 36, 38, 49, -24, 183, 103, 47, 38, 172, 124, 139, 2, 74, 21, 88
Current array: -180, 2, 140, 88, 57, 36, 38, 49, -24, 21, 103, 47, 38, 74, 124, 139, 149, 172, 183, 196
Current array: -180, -24, 2, 88, 57, 36, 38, 49, 140, 21, 103, 47, 38, 74, 124, 139, 149, 172, 183, 196
Current array: -180, -24, 2, 47, 57, 36, 38, 49, 74, 21, 38, 88, 103, 140, 124, 139, 149, 172, 183, 196
Current array: -180, -24, 2, 21, 38, 36, 38, 47, 74, 49, 57, 88, 103, 140, 124, 139, 149, 172, 183, 196
Current array: -180, -24, 2, 21, 38, 36, 38, 47, 74, 49, 57, 88, 103, 140, 124, 139, 149, 172, 183, 196
Current array: -180, -24, 2, 21, 38, 36, 38, 47, 74, 49, 57, 88, 103, 140, 124, 139, 149, 172, 183, 196
Current array: -180, -24, 2, 21, 36, 38, 38, 47, 74, 49, 57, 88, 103, 140, 124, 139, 149, 172, 183, 196
Current array: -180, -24, 2, 21, 36, 38, 38, 47, 49, 57, 74, 88, 103, 140, 124, 139, 149, 172, 183, 196
Current array: -180, -24, 2, 21, 36, 38, 38, 47, 49, 57, 74, 88, 103, 140, 124, 139, 149, 172, 183, 196
Current array: -180, -24, 2, 21, 36, 38, 38, 47, 49, 57, 74, 88, 103, 140, 124, 139, 149, 172, 183, 196
Current array: -180, -24, 2, 21, 36, 38, 38, 47, 49, 57, 74, 88, 103, 140, 124, 139, 149, 172, 183, 196
Current array: -180, -24, 2, 21, 36, 38, 38, 47, 49, 57, 74, 88, 103, 140, 124, 139, 149, 172, 183, 196
Current array: -180, -24, 2, 21, 36, 38, 38, 47, 49, 57, 74, 88, 103, 140, 124, 139, 149, 172, 183, 196
Current array: -180, -24, 2, 21, 36, 38, 38, 47, 49, 57, 74, 88, 103, 140, 124, 139, 149, 172, 183, 196
Current array: -180, -24, 2, 21, 36, 38, 38, 47, 49, 57, 74, 88, 103, 140, 124, 139, 149, 172, 183, 196
```

Có tổng cộng 7020 lệnh được thực hiện. Trong đó:

R-type: 1808 lệnh.

I-type: 4392 lệnh.

J-type: 820 lệnh.

TestCase 7: Dãy bao gồm các số dương, các số âm và số 0

Input: -62, 76, -126, 157, 0, -376, 498, 126, -299, -471, -47, 310, 96, -105, -231, -19, 358, -400, 296, 453

Output: -471, -400, -376, -299, -231, -126, -105, -62, -47, -19, 0, 76, 96, 126, 157, 296, 310, 358, 453, 498

Quy trình thực hiện:

```
Current array: -299, -400, -126, -231, -105, -376, -471, -62, 126, 498, -47, 310, 96, 0, 157, -19, 358, 76, 296, 453
Current array: -376, -400, -471, -299, -105, -231, -126, -62, 126, 498, -47, 310, 96, 0, 157, -19, 358, 76, 296, 453
Current array: -471, -400, -376, -299, -105, -231, -126, -62, 126, 498, -47, 310, 96, 0, 157, -19, 358, 76, 296, 453
Current array: -471, -400, -376, -299, -105, -231, -126, -62, 126, 498, -47, 310, 96, 0, 157, -19, 358, 76, 296, 453
Current array: -471, -400, -376, -299, -231, -126, -105, -62, 126, 498, -47, 310, 96, 0, 157, -19, 358, 76, 296, 453
Current array: -471, -400, -376, -299, -231, -126, -105, -62, 0, 76, -47, -19, 96, 126, 157, 310, 358, 498, 296, 453
Current array: -471, -400, -376, -299, -231, -126, -105, -62, -47, -19, 0, 76, 96, 126, 157, 310, 358, 498, 296, 453
Current array: -471, -400, -376, -299, -231, -126, -105, -62, -47, -19, 0, 76, 96, 126, 157, 310, 358, 498, 296, 453
Current array: -471, -400, -376, -299, -231, -126, -105, -62, -47, -19, 0, 76, 96, 126, 157, 310, 358, 498, 296, 453
Current array: -471, -400, -376, -299, -231, -126, -105, -62, -47, -19, 0, 76, 96, 126, 157, 310, 358, 498, 296, 453
Current array: -471, -400, -376, -299, -231, -126, -105, -62, -47, -19, 0, 76, 96, 126, 157, 296, 310, 453, 358, 498
Current array: -471, -400, -376, -299, -231, -126, -105, -62, -47, -19, 0, 76, 96, 126, 157, 296, 310, 358, 453, 498
Current array: -471, -400, -376, -299, -231, -126, -105, -62, -47, -19, 0, 76, 96, 126, 157, 296, 310, 358, 453, 498
```

Có tổng cộng 6483 lệnh được thực hiện. Trong đó:

R-type: 1658 lệnh.

I-type: 4074 lệnh.

J-type: 751 lệnh.

TestCase 8: Dãy bao gồm các số nhỏ và số rất lớn (trên 1000) và các số rất nhỏ (dưới 1000)

Input: 319, 1027, -43, 349, 833, 1330, -1641, -132, 153, -714, -1237, 342, -882, 1221, -1839, 1110, 325, 511, 478, 1708

Output: -1839, -1641, -1237, -882, -714, -132, -43, 153, 319, 325, 342, 349, 478, 511, 833, 1027, 1110, 1221, 1330, 1708

Quy trình thực hiện:

Current array:	153,	-1839,	-43,	-82,	-1237,	-714,	-1641,	-132,	319,	1330,	833,	342,	349,	1221,	1027,	1110,	325,	511,	478,	1708
Current array:	-132,	-1839,	-43,	-882,	-1237,	-714,	-1641,	153,	319,	1330,	833,	342,	349,	1221,	1027,	1110,	325,	511,	478,	1708
Current array:	-714,	-1839,	-1641,	-882,	-1237,	-132,	-43,	153,	319,	1330,	833,	342,	349,	1221,	1027,	1110,	325,	511,	478,	1708
Current array:	-1237,	-1839,	-1641,	-882,	-714,	-132,	-43,	153,	319,	1330,	833,	342,	349,	1221,	1027,	1110,	325,	511,	478,	1708
Current array:	-1641,	-1839,	-1237,	-882,	-714,	-132,	-43,	153,	319,	1330,	833,	342,	349,	1221,	1027,	1110,	325,	511,	478,	1708
Current array:	-1839,	-1641,	-1237,	-882,	-714,	-132,	-43,	153,	319,	1330,	833,	342,	349,	1221,	1027,	1110,	325,	511,	478,	1708
Current array:	-1839,	-1641,	-1237,	-882,	-714,	-132,	-43,	153,	319,	478,	833,	342,	349,	1221,	1027,	1110,	325,	511,	1330,	1708
Current array:	-1839,	-1641,	-1237,	-882,	-714,	-132,	-43,	153,	319,	349,	325,	342,	478,	1221,	1027,	1110,	833,	511,	1330,	1708
Current array:	-1839,	-1641,	-1237,	-882,	-714,	-132,	-43,	153,	319,	342,	325,	342,	478,	1221,	1027,	1110,	833,	511,	1330,	1708
Current array:	-1839,	-1641,	-1237,	-882,	-714,	-132,	-43,	153,	319,	325,	342,	349,	478,	1221,	1027,	1110,	833,	511,	1330,	1708
Current array:	-1839,	-1641,	-1237,	-882,	-714,	-132,	-43,	153,	319,	325,	342,	349,	478,	511,	1027,	1110,	833,	1221,	1330,	1708
Current array:	-1839,	-1641,	-1237,	-882,	-714,	-132,	-43,	153,	319,	325,	342,	349,	478,	511,	1027,	1110,	833,	1221,	1330,	1708
Current array:	-1839,	-1641,	-1237,	-882,	-714,	-132,	-43,	153,	319,	325,	342,	349,	478,	511,	833,	1027,	1110,	1221,	1330,	1708
Current array:	-1839,	-1641,	-1237,	-882,	-714,	-132,	-43,	153,	319,	325,	342,	349,	478,	511,	833,	1027,	1110,	1221,	1330,	1708

Có tổng cộng 6089 lệnh được thực hiện. Trong đó:

R-type: 1554 lệnh.

I-type: 3822 lệnh.

J-type: 713 lệnh.

TestCase 9: Dãy bao gồm tất cả các số bằng nhau

Input: 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3, 3

Output: 3, 3

Quy trình thực hiện:

[illegible]

Có tổng cộng 9141 lệnh được thực hiện. Trong đó:

R-type: 2407 lệnh.

I-type: 5644 lệnh.

J-type: 1090 lệnh.

TestCase 10: Dãy bao gồm các số dương, các số âm và số 0

Input: -6, 7, -9, 7, -3, 1, -6, 5, 0, 7, -1, 1, -8, 5, -3, 1, -2, 4, -4, 9

Output: -9, -8, -6, -6, -4, -3, -3, -2, -1, 0, 1, 1, 1, 4, 5, 5, 7, 7, 7, 9

Quy trình thực hiện:


```
Current array: -9, -8, -6, 7, -3, 1, -6, 5, 0, 7, -1, 1, 7, 5, -3, 1, -2, 4, -4, 9
Current array: -9, -8, -6, 7, -3, 1, -6, 5, 0, 7, -1, 1, 7, 5, -3, 1, -2, 4, -4, 9
Current array: -9, -8, -6, -4, -3, 1, -6, 5, 0, 7, -1, 1, 7, 5, -3, 1, -2, 4, 7, 9
Current array: -9, -8, -6, -6, -4, 1, -3, 5, 0, 7, -1, 1, 7, 5, -3, 1, -2, 4, 7, 9
Current array: -9, -8, -6, -6, -4, 1, -3, -2, 0, -3, -1, 1, 7, 5, 7, 1, 5, 4, 7, 9
Current array: -9, -8, -6, -6, -4, -1, -3, -2, 0, -3, 1, 1, 7, 5, 7, 1, 5, 4, 7, 9
Current array: -9, -8, -6, -6, -4, -3, -3, -2, -1, 0, 1, 1, 7, 5, 7, 1, 5, 4, 7, 9
Current array: -9, -8, -6, -6, -4, -3, -3, -2, -1, 0, 1, 1, 7, 5, 7, 1, 5, 4, 7, 9
Current array: -9, -8, -6, -6, -4, -3, -3, -2, -1, 0, 1, 1, 4, 5, 7, 1, 5, 7, 7, 9
Current array: -9, -8, -6, -6, -4, -3, -3, -2, -1, 0, 1, 1, 1, 4, 7, 5, 5, 7, 7, 9
Current array: -9, -8, -6, -6, -4, -3, -3, -2, -1, 0, 1, 1, 1, 4, 5, 5, 7, 7, 7, 9
Current array: -9, -8, -6, -6, -4, -3, -3, -2, -1, 0, 1, 1, 1, 4, 5, 5, 7, 7, 7, 9
Current array: -9, -8, -6, -6, -4, -3, -3, -2, -1, 0, 1, 1, 1, 4, 5, 5, 7, 7, 7, 9
```

Có tổng cộng 5836 lệnh được thực hiện. Trong đó:

R-type: 1521 lệnh.

I-type: 3632 lệnh.

J-type: 683 lệnh.

2.3.2 Tính và trình bày cách tính thời gian chạy của chương trình

Cách tính thời gian chạy theo công thức sau:

$$CPU_{time} = \frac{CPI \times IC}{ClockCycle}$$

TestCase 1: Có tổng tất cả 9186 lệnh. Áp dụng công thức trên:

$$CPU_{time} = \frac{1 \times 9186}{2 \times 10^9} = 4.593 \times 10^{-6} \text{ (s)}$$

TestCase 2: Có tổng tất cả 9236 lệnh. Áp dụng công thức trên:

$$CPU_{time} = \frac{1 \times 9236}{2 \times 10^9} = 4.618 \times 10^{-6} \text{ (s)}$$

TestCase 3: Có tổng tất cả 5773 lệnh. Áp dụng công thức trên:

$$CPU_{time} = \frac{1 \times 5773}{2 \times 10^9} = 2.8865 \times 10^{-6} \text{ (s)}$$

TestCase 4: Có tổng tất cả 6194 lệnh. Áp dụng công thức trên:

$$CPU_{time} = \frac{1 \times 6194}{2 \times 10^9} = 3.097 \times 10^{-6} \text{ (s)}$$

TestCase 5: Có tổng tất cả 5354 lệnh. Áp dụng công thức trên:

$$CPU_{time} = \frac{1 \times 5354}{2 \times 10^9} = 2.677 \times 10^{-6} \text{ (s)}$$

TestCase 6: Có tổng tất cả 7020 lệnh. Áp dụng công thức trên:

$$CPU_{time} = \frac{1 \times 7020}{2 \times 10^9} = 3.51 \times 10^{-6} \text{ (s)}$$

TestCase 7: Có tổng tất cả 6483 lệnh. Áp dụng công thức trên:

$$CPU_{time} = \frac{1 \times 6483}{2 \times 10^9} = 3.2415 \times 10^{-6} \text{ (s)}$$

TestCase 8: Có tổng tất cả 6089 lệnh. Áp dụng công thức trên:

$$CPU_{time} = \frac{1 \times 6089}{2 \times 10^9} = 3.0445 \times 10^{-6} \text{ (s)}$$

TestCase 9: Có tổng tất cả 9141 lệnh. Áp dụng công thức trên:

$$CPU_{time} = \frac{1 \times 9141}{2 \times 10^9} = 4.5705 \times 10^{-6} \text{ (s)}$$

TestCase 10: Có tổng tất cả 5836 lệnh. Áp dụng công thức trên:

$$CPU_{time} = \frac{1 \times 5836}{2 \times 10^9} = 2.918 \times 10^{-6} \text{ (s)}$$