

Bài 1: Cho mảng a[10], sắp xếp tăng dần.

1. Code mảng trên bằng Code C
2. Dịch ngược sang ASM
3. Tìm đoạn Code ASM thực hiện hoán vị 2 phần tử
4. Thay thế đoạn Code bằng lệnh XCHG
5. Đặt vấn đề sự khác nhau giữa bài học trên lớp và mã ASM đã dịch

Bài 2: Chạy thử lệnh PUSH, POP (đọc trước và làm trên máy)

Giải:**Bài 1:****1. Code mảng trên bằng code C**

// Bai 1: Code lệnh dung thuật toán sắp xếp - Nguyen Van Toan 20172852 - Source: NguyenVanHieu.vn

```
#include <stdio.h>
```

```
#include <conio.h>
```

```
// Array is swapped by Swap
```

```
void swap(char &a, char &b)
```

```
{
```

```
    char t = a;
```

```
    a = b;
```

```
    b = t;
```

```
}
```

```
// Using Quick Sort  $O[n \cdot \log(n)] < T < O[n^2]$  || Memory  $O[\log(n)]$ 
```

```
char partition (char a[10], char low, char high)
```

```
{
```

```
    char pivot = a[high];
```

```
    char left = low;
```

```
    char right = high - 1;
```

```
    while (true)
```

```
    {
```

```
        while(left <= right && a[left] < pivot)
```

```
            left ++;
```

```
        while(right >= left && a[right] > pivot)
```

```
            right--;
```

```
        if(left >= right)
```

```
        break;

        swap(a[left], a[right]);

        left ++;

        right --;

    }

    swap(a[left], a[high]);

    return left;

}

// The function perform array

void quicksort(char a[10], char low, char high)

{

    if (low < high)

    {

        // Array is divided by PI

        char PI = partition (a, low, high);

        // Recursively sorts 2 arrays

        quicksort(a, low, PI-1);

        quicksort(a, PI+1, high);

    }

}

// The function print

void printArray(char a[10], char size)

{

    for(char i = 0; i<size; i++)

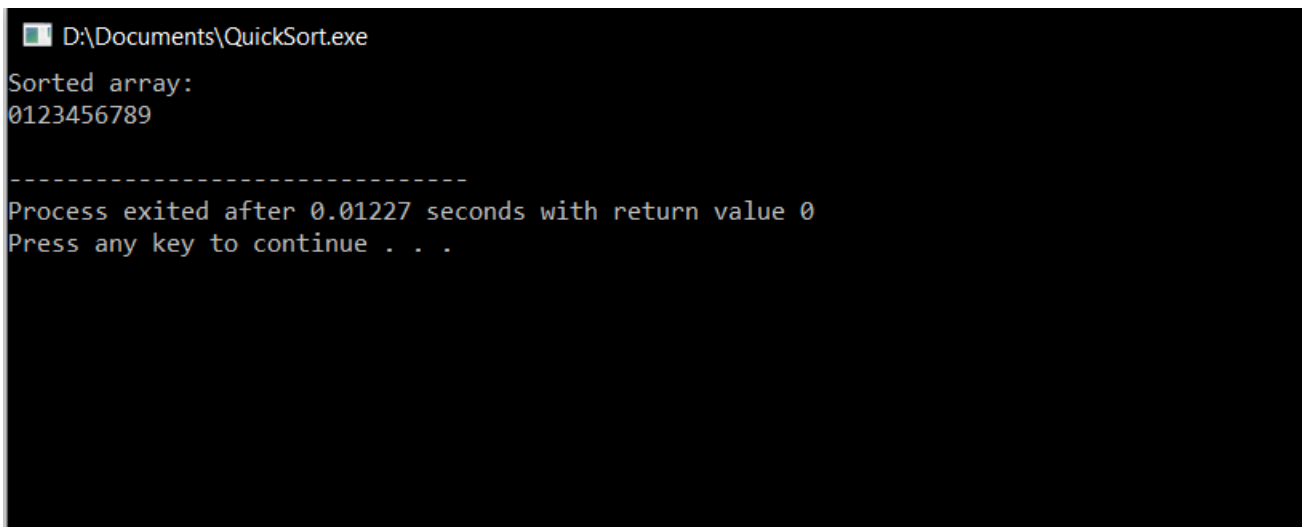
        printf("%d",a[i]);

    printf ("\n");

}
```

```
int main()
{
    char a[10]={1, 4, 6, 0, 8, 7, 5, 2, 3, 9};
    quicksort(a, 0, 9);
    printf("Sorted array: \n");
    printArray(a, 10);
    return 0;
}
```

- **Kết quả**



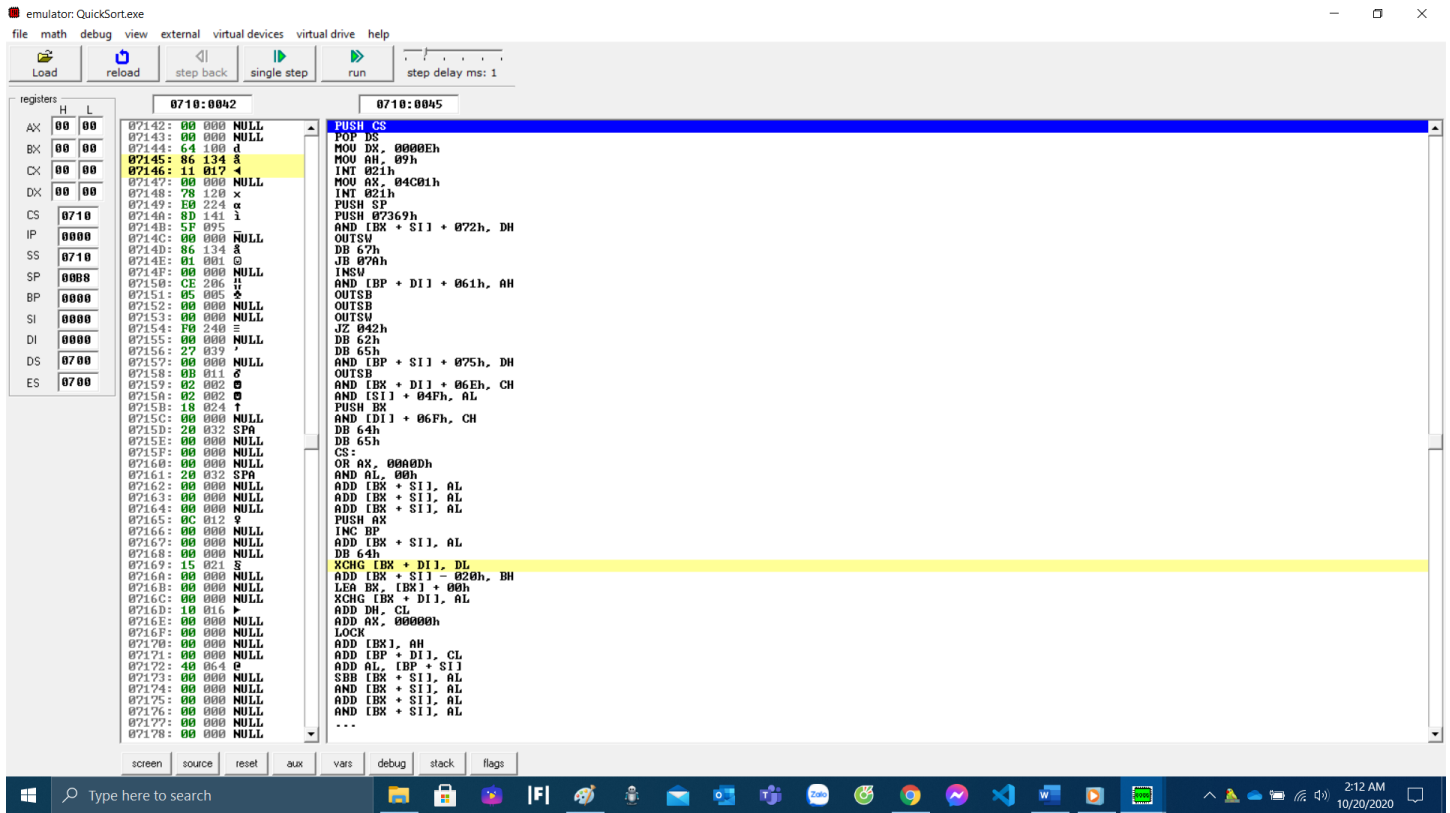
```
D:\Documents\QuickSort.exe
Sorted array:
0123456789

-----
Process exited after 0.01227 seconds with return value 0
Press any key to continue . . .
```

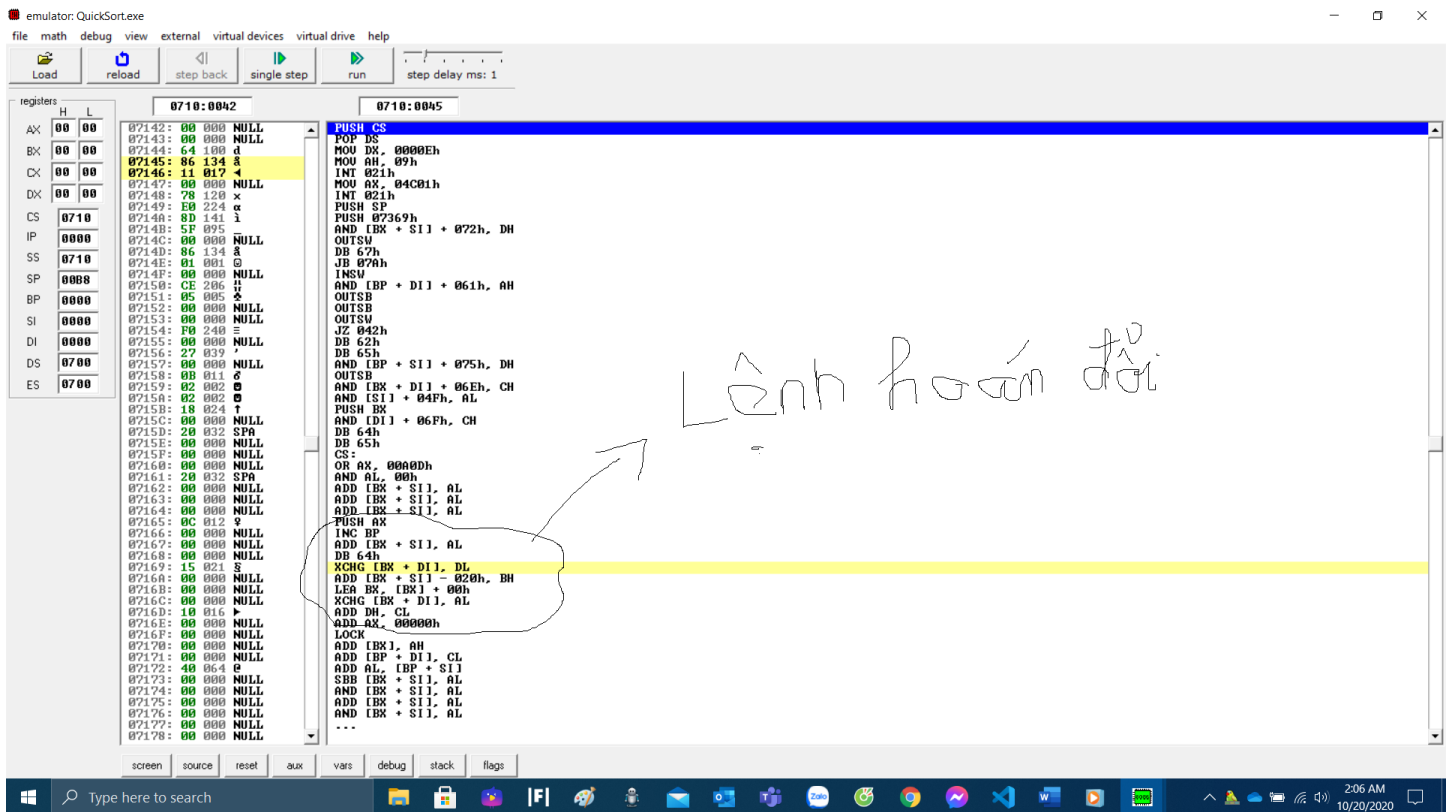
Size file: 129.9052734375 KiB

Compilation time: 0.64s

2. Chuyển code sang ASM (demo 1 đoạn)



3. Tìm đoạn code ASM cho đảo 2 phần tử



4. Thay thế đoạn code bằng lệnh XCHG (Hình trên)

5. Đặt vấn đề

- Khi debug lần đầu ra được file như EXE như trên nhưng file trên lại không chạy được trên nền DOS, trong khi code C của nó vẫn chạy trên DOS bình thường. Tại sao lại vậy?
- File trên là bản debug lần 1, lần 2 ra bản debug toàn MOV với ADD không thấy XCHG nên rất khó xác định. Vì sao lại vậy?