

Cấu trúc dữ liệu và giải thuật

Nhóm 15

BÀI TẬP VỀ NHÀ BUỔI 1

Họ và tên
Vũ Công Tuấn Dương

Mã sinh viên
B22DCKH024

Giảng viên:
TS Đỗ Thị Liên

Ngày nộp : 23/02/2024

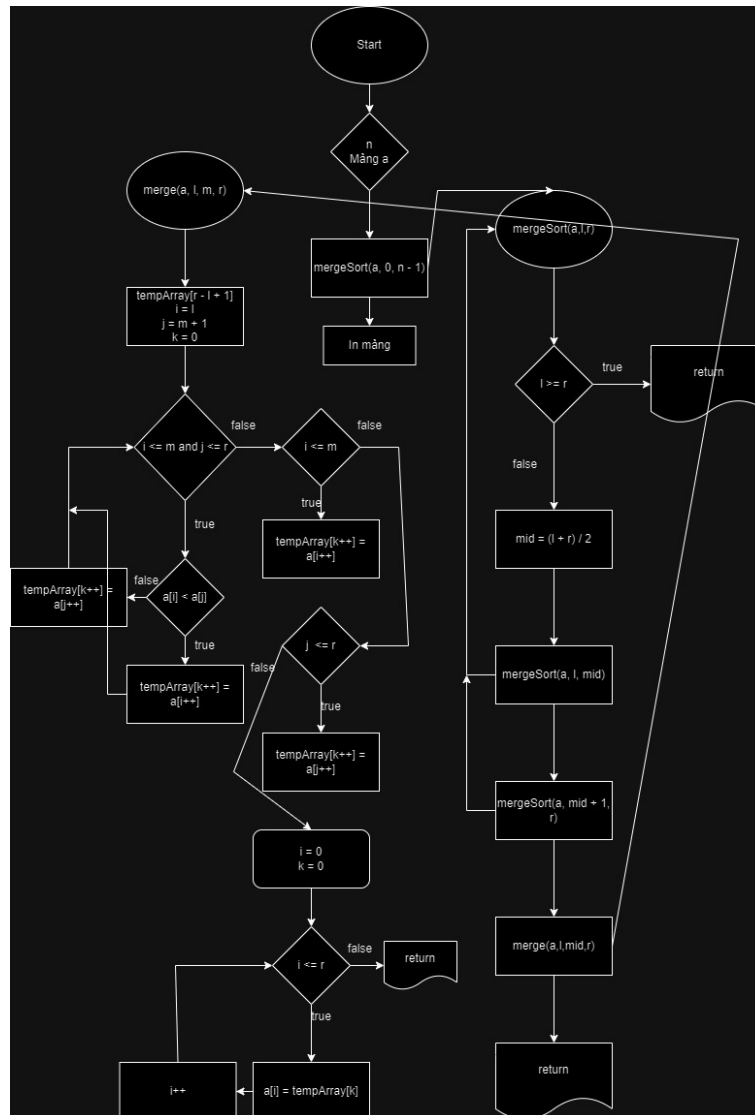
1 SẮP XẾP CHỮ SỐ

1. Xác định input, output:

- Input: Mảng a gồm n phần tử
- Output: Mảng a gồm các phần tử là các chữ số của phần tử ban đầu, được sắp xếp theo thứ tự tăng dần

2. Phân tích bài toán: Ta sẽ áp dụng thuật toán Merge Sort để sắp xếp các chữ số

3. Vẽ lưu đồ: Lưu đồ được thể hiện ở hình 1.



Hình 1: Lưu đồ thuật toán bài 1

4. Chạy từng bước

5. Viết code: Code được viết như sau:

```

#include <iostream>
#include <vector>
#include <map>
using namespace std;

void mergeSort(vector<int>& a, int l, int r);
void merge(vector<int>& a, int l, int m, int r);
void testCase();
int main () {
    int t;
    cin >> t;
    while (t--) {
        testCase();
        cout << endl;
    }

    return 0;
}

void testCase() {
    int x;
    cin >> x;
    vector<int> a;
    map<int, int> mp;
    for (int i = 0; i < x; i++) {
        string s;
        cin >> s;
        for (int i = 0; i < s.size(); i++) {
            mp[s[i] - '0']++;
        }
    }
    for (auto x : mp) {
        a.push_back(x.first);
    }
    int n = a.size();
    mergeSort(a, 0, n - 1);
    for (int i = 0; i < n; i++) {
        cout << a[i] << " ";
    }
}

void mergeSort(vector<int>& a, int l, int r) {
    if (l >= r) {
        return;
    }
    int mid = (l + r) / 2;
    mergeSort(a, l, mid);
    mergeSort(a, mid + 1, r);
    merge(a, l, mid, r);
}

void merge(vector<int>& a, int l, int m, int r) {

```

```

vector<int> tempArray(r - l + 1);
int i = l;
int j = m + 1;
int k = 0;
while (i <= m && j <= r) {
    if (a[i] < a[j]) {
        tempArray[k++] = a[i++];
    } else {
        tempArray[k++] = a[j++];
    }
}
while (i <= m) {
    tempArray[k++] = a[i++];
}
while (j <= r) {
    tempArray[k++] = a[j++];
}
for (i = l, k = 0; i <= r; i++, k++) {
    a[i] = tempArray[k];
}
}

```

6. Đánh giá độ phức tạp thuật toán: $O(t * \log(n))$