**ĐÁP ÁN ÔN TẬP THỰC HÀNH 1**

**Câu 1:**

1. Tổng điểm: 70 điểm.
2. Yêu cầu:

- Cơ sở dữ liệu, framework lập trình được phép tự chọn.

- Cho phép sử dụng một số hàm thư viên có sẵn để xử lý kết nối database, không cần phải lập trình các lớp xử lý kết nối.

- Không cần xây dựng các chức năng liên quan đến tài khoản như đăng nhập, đăng ký, phân quyền…

- Phần giao diện: có thể sử dụng bootstrap có sẵn để xây dựng.

- Yêu cầu các chức năng cần đạt được như sau:

* Trang Xem danh sách các nền được quản lý:

+ Thể hiện các thông tin: Tên nền, Diện tích, Giá trị nền, tình trạng, Người mua (nếu có).

+ Có bộ lọc theo tình trạng.

+ Có tìm kiếm theo tên người mua.

* Thêm thông tin nền và người mua.
* Cập nhật thông tin nền và người mua.
* Xóa thông tin nền và người mua.

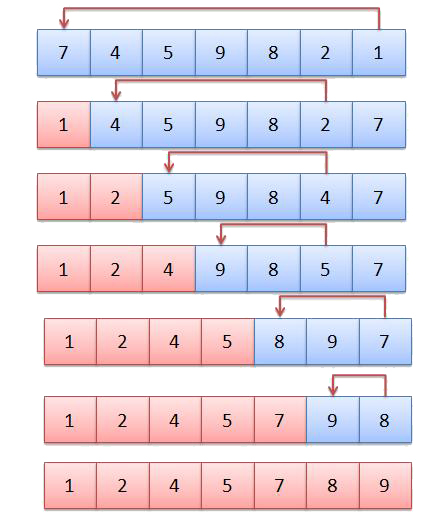
1. Thang điểm

|  |  |
| --- | --- |
| **Yêu cầu** | **Điểm tối đa** |
| Hiển thị được giao diện cho các trang theo yêu cầu | 10 |
| Giao diện các trang được thiết kế hợp lý, dễ sử dụng, có tính thẩm mỹ | 10 |
| Chức năng quản lý danh sách các nền hiển thị đúng | 10 |
| Có bộ lọc danh sách theo tình trạng và tìm kiếm theo tên người mua | 10 |
| Thêm mới thành công thông tin nền và người mua(nếu có) | 10 |
| Cập nhật thành công thong tin nền và người mua(nếu có) | 10 |
| Xóa thông tin nền kèm theo thông tin người mua (nếu có) | 10 |

**Câu 2:**

1. Tổng điểm: 30 điểm.
2. Yêu cầu:

* Dữ liệu đầu vào kiểm thử: Dãy số 7-4-5-9-8-2-1.
* Dữ liệu đầu ra: Dãy số đã sắp xếp 1-2-4-5-7-8-9.
* In được các bước sắp xếp theo thứ tự từng lần thay đổi.



* Lập trình được thuật toán theo selection sort như sau:

1. Thang điểm:

|  |  |
| --- | --- |
| **Thang điểm** | **Điểm tối đa** |
| In được kết quả đã sắp xếp cuối cùng | 10đ |
| In được danh sách tất các lần sắp xếp | 10đ |
| Thuật toán chính xác | 10đ |

void selectionSort(int a[], int n){

int i, j, min, temp;

for (i = 0; i < n-1; i++) {

min = i;

for (j = i+1; j < n; j++){

if (a[j] < a[min]) min = j;

}

swap(a[i], a[min]);

}

}