

# Bài tập lớn kỳ 2024.2

**Đề tài:** Sử dụng CTDL danh sách liên kết đơn để xây dựng logic nghiệp vụ cho ứng dụng quản lý đơn hàng của nhà hàng.

**Học phần:** Cấu trúc dữ liệu và thuật toán

**Mã học phần:** IT3011 - **Mã lớp:** 159516

**Sinh viên thực hiện:** Đặng Tiến Cường - **MSSV:** 20220020

**Giảng viên hướng dẫn:** ThS. Nguyễn Duy Hiệp



# Nội dung trình bày

- Một vài thao tác phân tích quy trình nghiệp vụ đơn giản
  - Đặc tả đơn giản ca sử dụng quản lý đơn hàng
  - Sơ đồ máy trạng thái cho đơn hàng và món ăn
- Mô tả cấu trúc dữ liệu
- Demo chương trình.



## Đặc tả ca sử dụng **Quản lý đơn hàng (order management)**

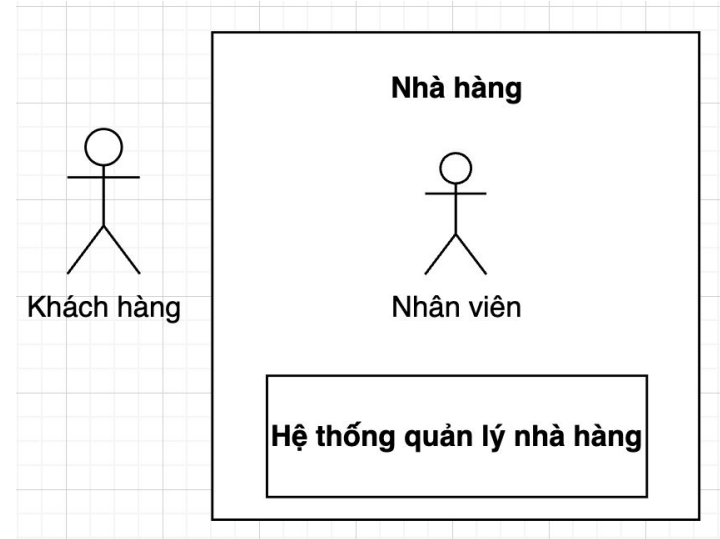
### Các luồng chính:

1. Khách hàng gọi điện đặt bàn hoặc đặt bàn trực tiếp tại quầy lễ tân.
2. Nhân viên nhà hàng tạo đơn hàng (order) theo yêu cầu của khách hàng.
3. Nhân viên lên đơn hàng và gửi cho khách hàng xác nhận.
4. Khách hàng xác nhận đơn hàng và nhân viên gửi đơn hàng cho nhà bếp.
5. Nhà bếp làm các món ăn theo đơn hàng và nhân viên cập nhật thông tin các món ăn đã trả, các đĩa đã trả.
6. Khách hàng thưởng thức các món ăn đã trả, các đĩa đã trả (các món ăn mà nhà bếp đã làm xong).
7. Sau khi ăn xong, nhân viên tạo hoá đơn của đơn hàng tương ứng với mã bàn khi đặt.
8. Khách hàng thanh toán đơn hàng và nhân viên cập nhật trạng thái của đơn hàng là đã thanh toán.

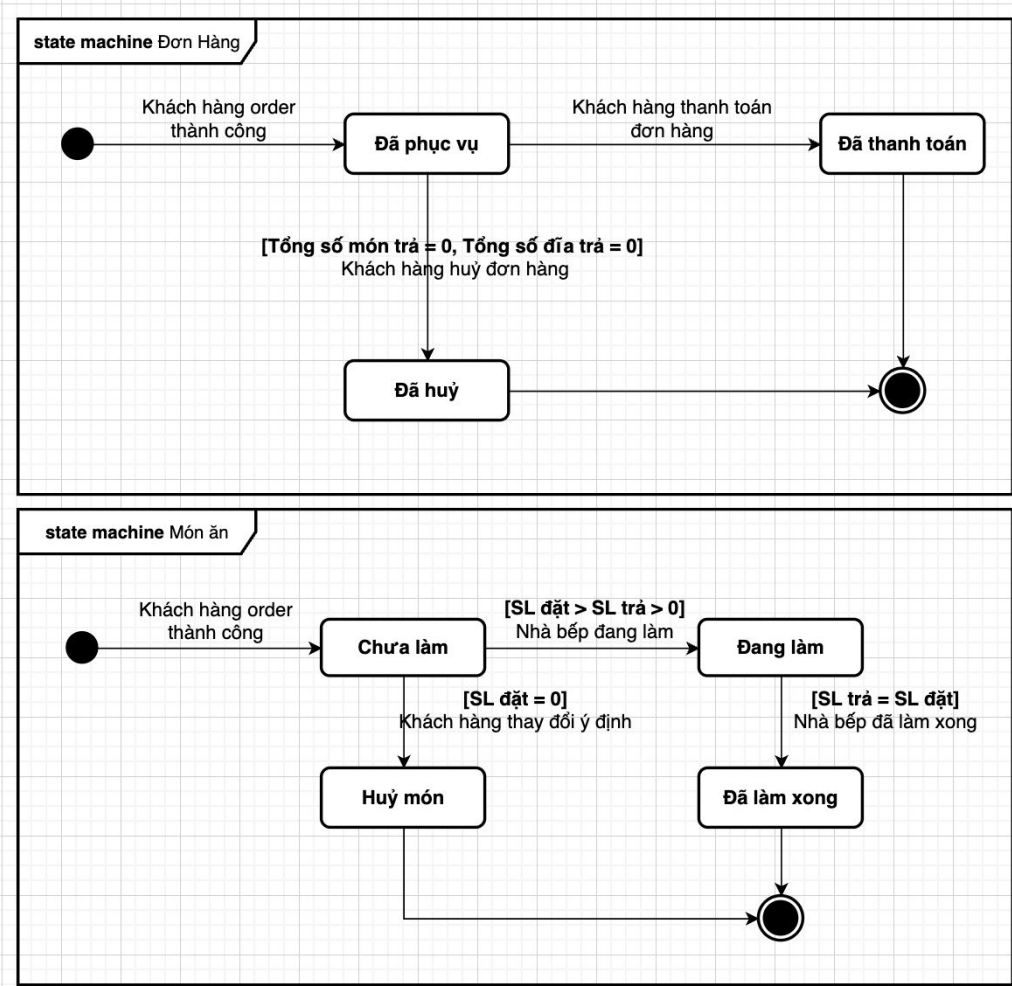
### Các luồng thay thế/ tương đương:

4. Khách hàng huỷ đơn hàng
  - 4.1. Nhân viên tìm kiếm đơn hàng theo mã bàn
  - 4.2. Nhân viên huỷ đơn hàng nếu như đơn hàng chưa trả món nào.
4. Khách hàng gọi thêm món
  - 4.1. Nhân viên tìm kiếm đơn hàng theo mã bàn.
  - 4.2. Nhân viên cập nhật thông tin đơn hàng.
4. Khách hàng huỷ món
  - 4.1. Nhân viên tìm kiếm đơn hàng theo mã bàn.
  - 4.2. Nhân viên cập nhật thông tin đơn hàng.

## Xác định đường biên hệ thống



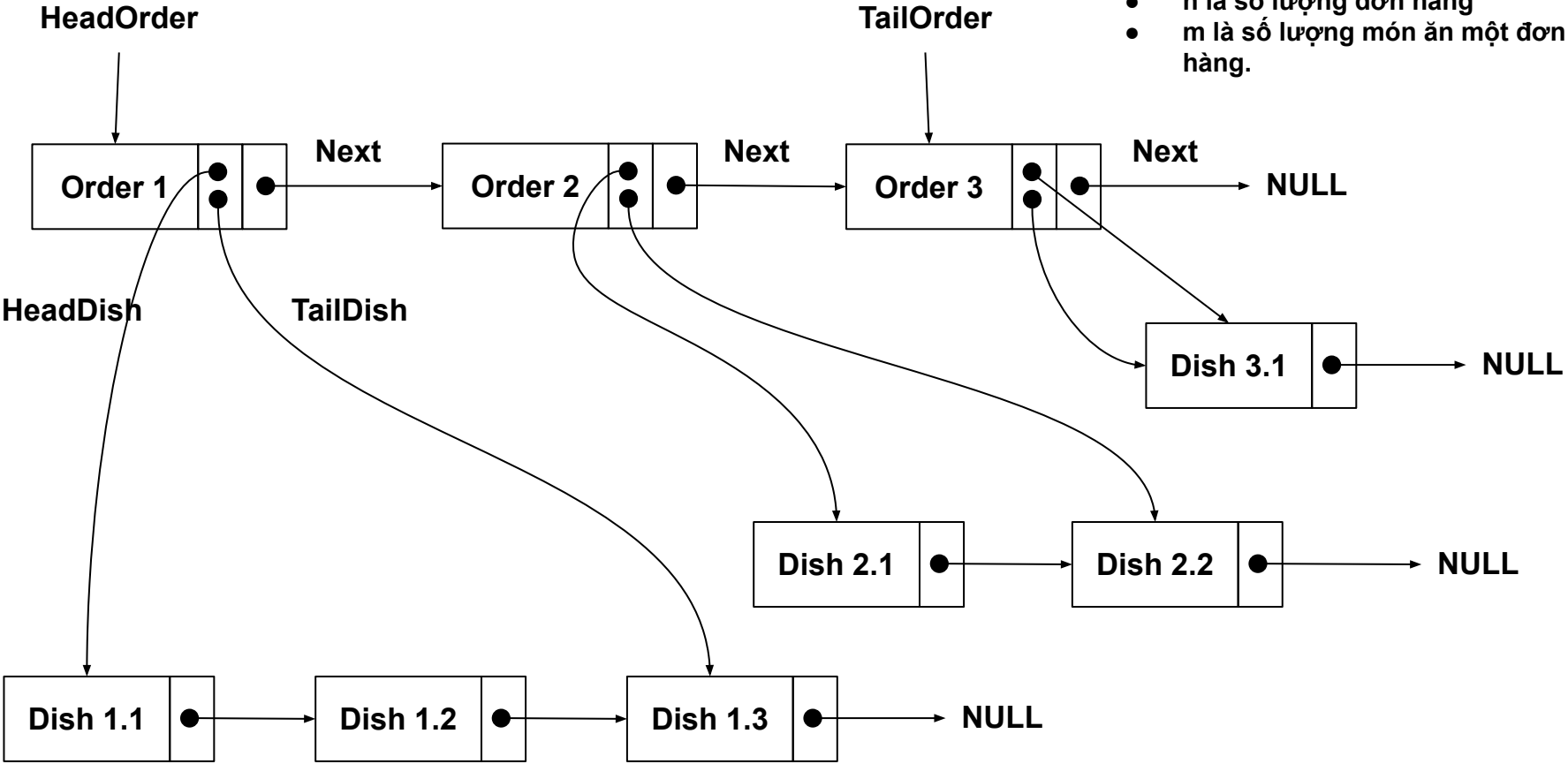
# Trạng thái của đối tượng đơn hàng (order) và món ăn (dish)



# Mô tả cấu trúc dữ liệu danh sách đơn hàng và món ăn

Thêm đơn hàng:  $O(1)$   
Thêm món ăn:  $O(n)$   
Tìm kiếm, cập nhật đơn hàng, món ăn:  $O(n) + O(m)$

- $n$  là số lượng đơn hàng
- $m$  là số lượng món ăn một đơn hàng.



# Cài đặt cấu trúc dữ liệu

```
// Trạng thái đơn hàng
typedef enum {
    DANG_PHUC_VU,
    DA_THANH_TOAN,
    DON_HUY
} TrangThaiDonHang;
```

```
// Trạng thái món ăn
typedef enum {
    CHUA_LAM,
    DANG_LAM,
    DA_LAM_XONG,
    DA_HUY
} TrangThaiMonAn;
```

```
enum TrangThaiDonHang {
    DANG_PHUC_VU,
    DA_THANH_TOAN,
    DON_HUY
};

enum TrangThaiMonAn {
    CHUA_LAM,
    DANG_LAM,
    DA_LAM_XONG,
    DA_HUY
};
```

```
32 // Cấu trúc món ăn
33 typedef struct Dish {
34     char maMon[20];
35     char tenMon[MAX_NAME];
36     int soLuongDat;
37     int soLuongTra;
38     int giaTien;
39     char thoiGianTaoMon[20];
40     char thoiGianCapNhat[20];
41     char ghiChu[MAX_NOTE];
42     TrangThaiMonAn trangThai;
43     struct Dish* next;
44 } Dish;
```

```
45
46 // Cấu trúc danh sách món ăn
47 typedef struct DishList {
48     Dish* headDish;
49     Dish* tailDish;
50 } DishList;
```

```
51
52
```

```
enum TrangThaiDonHang {
    DANG_PHUC_VU,
    DA_THANH_TOAN,
    DON_HUY
};

enum TrangThaiMonAn {
    CHUA_LAM,
    DANG_LAM,
    DA_LAM_XONG,
    DA_HUY
};
```

```
57 // Cấu trúc đơn hàng
58 typedef struct Order {
59     int maBan;
60     char maNhanVien[20];
61     int tongSoMon; // Tổng s
62     int tongSoDiaDat; // Tổng s
63     int tongSoMonTra; // Tổng s
64     int tongSoDiaTra; // Tổng s
65     long long tongTien;
66     char thoiGianTaoDon[20];
67     char thoiGianCapNhat[20];
68     DishList* danhSachMon;
69     TrangThaiDonHang trangThai;
70     struct Order* next;
71 } Order;
```

```
72
73 // Cấu trúc danh sách đơn hàng
74 typedef struct {
75     Order* headOrder;
76     Order* tailOrder;
77 } OrderList;
```

```
--
```

## Các hàm quản lý đơn hàng

```
// Hàm tạo đối tượng đơn hàng
Order* makeNewOrder(int maBan, char* maNV, char* thoiGianTaoDon);

// Hàm tìm kiếm và hiển thị
Order* search_order(OrderList *orderList, int maBan);
void print_order(OrderList* orderList, int maBan);

// Hàm tạo mới đơn hàng
Order* create_order(OrderList* orderList, int maBan, char* maNV, char* thoiGian);

// Hàm quản lý đơn hàng – trả về 1 nếu thành công, 0 nếu thất bại
int cancel_order(OrderList *orderList, int maBan);

// Hàm xuất hóa đơn
void create_bill(OrderList *orderList, int maBan);

// Hàm quản lý bộ nhớ
OrderList* init_order_list();
void free_order_list(OrderList* orderList);
```

## Các hàm quản lý món ăn

// Hàm tạo đối tượng món ăn

```
Dish* makeNewDish(char* maMon, char* tenMon, int giaTien,  
| | | | int soLuongDat, char *thoiGianTaoMon, char* ghiChu);
```

// Hàm tìm kiếm món ăn

```
Dish* search_dish(DishList *dishList, char* maMon, char *tenMon);
```

// Hàm quản lý món ăn – trả về 1 nếu thành công, 0 nếu thất bại

```
int add_dish(OrderList *orderList, char* maNV, int maBan,  
| | | | char* maMon, char* tenMon, int soLuong, int giaTien, char* ghiChu);  
int update_dish(OrderList *orderList, int maBan, char* maMon, char* tenMon, int soLuongTra);  
int cancel_dish(OrderList* orderList, int maBan, char* maMon, char* tenMon, char* ghiChu);
```

// Hàm giải phòng món ăn khỏi danh sách

```
void free_dish(DishList *dishList, Dish* searchDish);  
void free_dish_list(DishList *dishList);
```



