

Bài 9: Ôn tập cấu trúc dữ liệu & Chuẩn bị Mini Project

W27 - Thứ 6 - Buổi sáng (09:00-12:00)

Mục tiêu:

- Ôn tập toàn bộ kiến thức về cấu trúc dữ liệu đã học
- củng cố kỹ năng lựa chọn cấu trúc dữ liệu phù hợp
- Chuẩn bị cho Mini Project Contact Manager
- Thực hành với các bài tập tổng hợp

1. ÔN TẬP TỔNG QUAN CÁC CẤU TRÚC DỮ LIỆU

1.1 Bảng tóm tắt toàn bộ kiến thức

```
In [1]: print("1. LIST - Danh sách có thứ tự, thay đổi được")
danhsachdiem = [8, 7, 9, 6, 10, 8]
print(f"    Ví dụ: {danhsachdiem}")
print(f"    Truy cập: danhsachdiem[0] = {danhsachdiem[0]}")
print(f"    Slicing: danhsachdiem[1:4] = {danhsachdiem[1:4]}")
print(f"    Phương thức: append(), remove(), sort(), len()")
print(f"    Sử dụng khi: Cần thứ tự, thay đổi được, truy cập theo index")
print()

print("2. DICTIONARY - Ánh xạ key-value, thay đổi được")
thongtinsv = {"ten": "Nguyễn Văn A", "tuoi": 20, "diem": 8.5}
print(f"    Ví dụ: {thongtinsv}")
print(f"    Truy cập: thongtinsv['ten'] = {thongtinsv['ten']}")
print(f"    Phương thức: keys(), values(), items(), get()")
print(f"    Sử dụng khi: Cần ánh xạ key-value, truy cập nhanh theo key")
print()

print("3. TUPLE - Bộ dữ liệu có thứ tự, KHÔNG thay đổi được")
toa_do = (10, 20)
ngaysinh = ("1990", "01", "15")
print(f"    Ví dụ: {toa_do}, {ngaysinh}")
print(f"    Truy cập: toa_do[0] = {toa_do[0]}")
print(f"    Unpacking: nam, thang, ngay = ngay_sinh")
nam, thang, ngay = ngay_sinh
print(f"    Kết quả unpacking: {nam}, {thang}, {ngay}")
print(f"    Sử dụng khi: Dữ liệu cố định như tọa độ, ngày tháng")
print()
```

```

print("4. SET - Tập hợp không thứ tự, KHÔNG trùng lặp")
mon_hoc = {"Toán", "Lý", "Hóa", "Toán"} # Chỉ có 3 phần tử
print(f"    Ví dụ: {mon_hoc}")
print(f"    Phương thức: add(), remove(), union(), intersection()")
print(f"    Sử dụng khi: Loại bỏ trùng lặp, phép toán tập hợp")
print()

print("=== KẾT HỢP CÁC CẤU TRÚC - Học chiều hôm qua ===")
print("- List chứa Dictionary: Danh sách học sinh")
print("- Dictionary chứa List: Tổ chức theo nhóm")
print("- Tuple trong Dictionary: Thông tin cố định")
print("- Set để thống kê: Dữ liệu không trùng lặp")

```

1. LIST - Danh sách có thứ tự, thay đổi được
 Ví dụ: [8, 7, 9, 6, 10, 8]
 Truy cập: danh_sach_diem[0] = 8
 Slicing: danh_sach_diem[1:4] = [7, 9, 6]
 Phương thức: append(), remove(), sort(), len()
 Sử dụng khi: Cần thứ tự, thay đổi được, truy cập theo index
2. DICTIONARY - Ánh xạ key-value, thay đổi được
 Ví dụ: {'ten': 'Nguyễn Văn A', 'tuoi': 20, 'diem': 8.5}
 Truy cập: thong_tin_sv['ten'] = Nguyễn Văn A
 Phương thức: keys(), values(), items(), get()
 Sử dụng khi: Cần ánh xạ key-value, truy cập nhanh theo key
3. TUPLE - Bộ dữ liệu có thứ tự, KHÔNG thay đổi được
 Ví dụ: (10, 20), ('1990', '01', '15')
 Truy cập: toa_do[0] = 10
 Unpacking: nam, thang, ngay = ngay_sinh
 Kết quả unpacking: 1990, 01, 15
 Sử dụng khi: Dữ liệu cố định như tọa độ, ngày tháng
4. SET - Tập hợp không thứ tự, KHÔNG trùng lặp
 Ví dụ: {'Hóa', 'Toán', 'Lý'}
 Phương thức: add(), remove(), union(), intersection()
 Sử dụng khi: Loại bỏ trùng lặp, phép toán tập hợp

```

=== KẾT HỢP CÁC CẤU TRÚC - Học chiều hôm qua ===
- List chứa Dictionary: Danh sách học sinh
- Dictionary chứa List: Tổ chức theo nhóm
- Tuple trong Dictionary: Thông tin cố định
- Set để thống kê: Dữ liệu không trùng lặp

```

1.2 Ma trận quyết định lựa chọn cấu trúc

```

In [2]: print("\n=== MA TRẬN QUYẾT ĐỊNH LỰA CHỌN CẤU TRÚC ===")

# Thử nghiệm các tình huống
print("\n=== THỬ NGHIỆM CÁC TÌNH HUỐNG ===")

# Tình huống 1: Lưu điểm số học sinh
print("\nTình huống 1: Lưu điểm số học sinh (có thể thêm điểm mới)")
can_thu_tu = True
can_thay_doi = True

```

```

cho_phep_trung_lap = True
kieu_truy_cap = "index"

print(f"Cần thứ tự: {can_thu_tu}")
print(f"Cần thay đổi: {can_thay_doi}")
print(f"Cho phép trùng lặp: {cho_phep_trung_lap}")
print(f"Kiểu truy cập: {kieu_truy_cap}")

if kieu_truy_cap == "key":
    print("Gợi ý: DICTIONARY - Truy cập nhanh theo key")
elif can_thu_tu and can_thay_doi and cho_phep_trung_lap:
    print("Gợi ý: LIST - Có thứ tự, thay đổi được, cho phép trùng")
elif can_thu_tu and not can_thay_doi:
    print("Gợi ý: TUPLE - Có thứ tự, không thay đổi")
elif not cho_phep_trung_lap:
    print("Gợi ý: SET - Không trùng lặp, tìm kiếm nhanh")
else:
    print("Gợi ý: LIST hoặc TUPLE tùy vào có cần thay đổi không")

# Tình huống 2: Lưu thông tin cá nhân cố định
print("\nTình huống 2: Lưu thông tin cá nhân cố định")
can_thu_tu = True
can_thay_doi = False
cho_phep_trung_lap = True
kieu_truy_cap = "index"

print(f"Cần thứ tự: {can_thu_tu}")
print(f"Cần thay đổi: {can_thay_doi}")
print(f"Cho phép trùng lặp: {cho_phep_trung_lap}")
print(f"Kiểu truy cập: {kieu_truy_cap}")

if can_thu_tu and not can_thay_doi:
    print("Gợi ý: TUPLE - Có thứ tự, không thay đổi")

# Tình huống 3: Danh sách môn học đã đăng ký (không trùng)
print("\nTình huống 3: Danh sách môn học (không cho phép trùng)")
can_thu_tu = False
can_thay_doi = True
cho_phep_trung_lap = False
kieu_truy_cap = "member"

print(f"Cần thứ tự: {can_thu_tu}")
print(f"Cần thay đổi: {can_thay_doi}")
print(f"Cho phép trùng lặp: {cho_phep_trung_lap}")
print(f"Kiểu truy cập: {kieu_truy_cap}")

if not cho_phep_trung_lap:
    print("Gợi ý: SET - Không trùng lặp, tìm kiếm nhanh")

# Tình huống 4: Tra cứu điểm theo tên môn
print("\nTình huống 4: Tra cứu điểm theo tên môn")
can_thu_tu = False
can_thay_doi = True
cho_phep_trung_lap = True
kieu_truy_cap = "key"

```

```
print(f"Cần thứ tự: {can_thu_tu}")
print(f"Cần thay đổi: {can_thay_doi}")
print(f"Cho phép trùng lặp: {cho_phep_trung_lap}")
print(f"Kiểu truy cập: {kieu_truy_cap}")

if kieu_truy_cap == "key":
    print("Gợi ý: DICTIONARY - Truy cập nhanh theo key")
```

=== MA TRẬN QUYẾT ĐỊNH LỰA CHỌN CẤU TRÚC ===

=== THỬ NGHIỆM CÁC TÌNH HUỐNG ===

Tình huống 1: Lưu điểm số học sinh (có thể thêm điểm mới)

Cần thứ tự: True

Cần thay đổi: True

Cho phép trùng lặp: True

Kiểu truy cập: index

Gợi ý: LIST - Có thứ tự, thay đổi được, cho phép trùng

Tình huống 2: Lưu thông tin cá nhân cố định

Cần thứ tự: True

Cần thay đổi: False

Cho phép trùng lặp: True

Kiểu truy cập: index

Gợi ý: TUPLE - Có thứ tự, không thay đổi

Tình huống 3: Danh sách môn học (không cho phép trùng)

Cần thứ tự: False

Cần thay đổi: True

Cho phép trùng lặp: False

Kiểu truy cập: member

Gợi ý: SET - Không trùng lặp, tìm kiếm nhanh

Tình huống 4: Tra cứu điểm theo tên môn

Cần thứ tự: False

Cần thay đổi: True

Cho phép trùng lặp: True

Kiểu truy cập: key

Gợi ý: DICTIONARY - Truy cập nhanh theo key



TODO 1: Thực hành lựa chọn cấu trúc

```
In [ ]: # TODO 1: Cho các tình huống sau, hãy chọn cấu trúc dữ liệu phù hợp

# Tình huống A: Lưu trữ lịch sử mua hàng của khách (thời gian, sản phẩm, giá)
# TODO: Chọn cấu trúc và tạo ví dụ
lich_su_mua_hang = [
    {"thoi_gian": "2025-07-04", "san_pham": "ao samsung", "gia": 250},
    {"thoi_gian": "2025-07-03", "san_pham": "quan samsung", "gia": 200},
    {"thoi_gian": "2025-07-01", "san_pham": "dien thoai samsung", "gia": 1340}
]
print("Tình huống A - Lịch sử mua hàng:")
print(f"Cấu trúc chọn: List của dictionary")
print(f"Lý do: ...")
print(f"Ví dụ: {lich_su_mua_hang}")
```

```

print(type(lich_su_mua_hang))

# Tình huống B: Lưu tọa độ GPS của các địa điểm (không thay đổi)
# TODO: Chọn cấu trúc và tạo ví dụ
toa_do_truong = (198.2763, 200.2567)
toa_do_benh_vien = (150.563, 10.0067)
print("\nTình huống B - Tọa độ GPS:")
print(f"Cấu trúc chọn: tuple")
print(f"Lý do: ...")

# Tình huống C: Quản lý danh sách bạn bè (tên, tuổi, sở thích)
# TODO: Chọn cấu trúc và tạo ví dụ
# danh_sach_ban_be = ?
# print("\nTình huống C - Danh sách bạn bè:")
# print(f"Cấu trúc chọn: ...")
# print(f"Lý do: ...")

# Tình huống D: Thống kê các thành phố đã đi du lịch (không trùng)
# TODO: Chọn cấu trúc và tạo ví dụ
cac_thanh_pho_da_di = {"Ha Noi", "Ho Chi Minh"}
cac_thanh_pho_o_Viet_Nam = {"Ha Noi", "Ho Chi Minh", "Da Nang", "Hung Yen"}
thanh_pho_moi = "Da Lat"
cac_thanh_pho_da_di.add("Ha Noi")

print("\nTình huống D - Thành phố đã đi:")
print(f"Cấu trúc chọn: ...")
print(f"Lý do: ...")

print("Hoàn thành TODO 1 để thực hành lựa chọn cấu trúc!")

```

Tình huống A - Lịch sử mua hàng:

Cấu trúc chọn: List của dictionary

Lý do: ...

Ví dụ: [{ 'thoi_gian': '2025-07-04', 'san_pham': 'ao samsung', 'gia': 250}, { 'thoi_gian': '2025-07-03', 'san_pham': 'quan samsung', 'gia': 200}, { 'thoi_gian': '2025-07-01', 'san_pham': 'dien thoai samsung', 'gia': 1340}]

<class 'list'>

Tình huống B - Tọa độ GPS:

Cấu trúc chọn: tuple

Lý do: ...

Hoàn thành TODO 1 để thực hành lựa chọn cấu trúc!

```

In [10]: tong = 0
         for mua_hang in lich_su_mua_hang:
             tong += mua_hang["gia"]
         tong

```

Out[10]: 1790

```

In [15]: import math
         x1, y1 = toa_do_truong
         x2, y2 = toa_do_benh_vien
         khoang_cach = math.sqrt((x1 - x2) ** 2 + (y1 - y2) ** 2)
         round(khoang_cach, 2)

```

Out[15]: 196.14

```
In [11]: tong_mua_hang = sum(mua_hang["gia"] for mua_hang in lich_su_mua_hang)
tong_mua_hang
```

Out[11]: 1790

2. BÀI TẬP ÔN TẬP TỔNG HỢP

2.1 Bài tập 1: Phân tích dữ liệu bán hàng

```
In [4]: print("\n=== BÀI TẬP 1: PHÂN TÍCH DỮ LIỆU BÁN HÀNG ===")

# Dữ liệu bán hàng (kết hợp nhiều cấu trúc)
du_lieu_ban_hang = {
    "thong_tin_shop": {
        "ten": "Shop Online ABC",
        "dia_chi": ("123 Đường XYZ", "Quận 1", "TP.HCM"),
        "ngay_mo": (2020, 5, 15)
    },

    "san_pham": [
        {"id": 1, "ten": "Áo thun", "gia": 150000, "danh_muc": "Thời trang"},
        {"id": 2, "ten": "Quần jean", "gia": 300000, "danh_muc": "Thời trang"},
        {"id": 3, "ten": "Giày sneaker", "gia": 500000, "danh_muc": "Giày dép"},
        {"id": 4, "ten": "Túi xách", "gia": 200000, "danh_muc": "Phụ kiện"},
        {"id": 5, "ten": "Áo khoác", "gia": 400000, "danh_muc": "Thời trang"}
    ],

    "don_hang": [
        {"id": 1, "khach_hang": "Nguyễn A", "san_pham_id": 1, "so_luong": 2, "ngay": "2020-05-15"},
        {"id": 2, "khach_hang": "Trần B", "san_pham_id": 3, "so_luong": 1, "ngay": "2020-05-16"},
        {"id": 3, "khach_hang": "Lê C", "san_pham_id": 2, "so_luong": 1, "ngay": "2020-05-17"},
        {"id": 4, "khach_hang": "Nguyễn A", "san_pham_id": 4, "so_luong": 3, "ngay": "2020-05-18"},
        {"id": 5, "khach_hang": "Phạm D", "san_pham_id": 1, "so_luong": 1, "ngay": "2020-05-19"}
    ]
}

print(f"Shop: {du_lieu_ban_hang['thong_tin_shop']['ten']}")
print(f"Số sản phẩm: {len(du_lieu_ban_hang['san_pham'])}")
print(f"Số đơn hàng: {len(du_lieu_ban_hang['don_hang'])}")

# Unpacking địa chỉ shop
duong, quan, thanh_pho = du_lieu_ban_hang["thong_tin_shop"]["dia_chi"]
print(f"Địa chỉ: {duong}, {quan}, {thanh_pho}")

# Unpacking ngày mở shop
nam, thang, ngay = du_lieu_ban_hang["thong_tin_shop"]["ngay_mo"]
print(f"Ngày mở shop: {ngay}/{thang}/{nam}")

# Tạo dictionary để tra cứu sản phẩm theo ID
san_pham_dict = {}
for sp in du_lieu_ban_hang["san_pham"]:
```

```

    san_pham_dict[sp["id"]] = sp

print(f"\nSẢN PHẨM ID 3: {san_pham_dict[3]['ten']} - {san_pham_dict[3]['gia']:,} VNĐ")

# Tính tổng doanh thu cho từng đơn hàng
print("\n=== DOANH THU TỪNG ĐƠN HÀNG ===")
tong_doanh_thu = 0

for don_hang in du_lieu_ban_hang["don_hang"]:
    sp_id = don_hang["san_pham_id"]
    so_luong = don_hang["so_luong"]
    gia = san_pham_dict[sp_id]["gia"]
    doanh_thu_don = gia * so_luong

    print(f"Đơn {don_hang['id']}: {san_pham_dict[sp_id]['ten']} x{so_luong} = {doanh_thu_don}")
    tong_doanh_thu += doanh_thu_don

print(f"\nTổng doanh thu: {tong_doanh_thu:,} VNĐ")

# Tìm tất cả khách hàng duy nhất (sử dụng Set)
tat_ca_khach_hang = set()
for don_hang in du_lieu_ban_hang["don_hang"]:
    tat_ca_khach_hang.add(don_hang["khach_hang"])

print(f"\n=== KHÁCH HÀNG ===")
print(f"Tổng số khách hàng duy nhất: {len(tat_ca_khach_hang)}")
print(f"Danh sách: {sorted(list(tat_ca_khach_hang))}")

# Thống kê số lần mua của từng khách
thong_ke_khach = {}
for don_hang in du_lieu_ban_hang["don_hang"]:
    khach = don_hang["khach_hang"]
    if khach not in thong_ke_khach:
        thong_ke_khach[khach] = 0
    thong_ke_khach[khach] += 1

print("\n=== THỐNG KÊ KHÁCH HÀNG ===")
for khach, so_lan in thong_ke_khach.items():
    print(f"{khach}: {so_lan} lần mua")

# Tìm các danh mục sản phẩm duy nhất
cac_danh_muc = set()
for sp in du_lieu_ban_hang["san_pham"]:
    cac_danh_muc.add(sp["danh_muc"])

print(f"\n=== DANH MỤC SẢN PHẨM ===")
print(f"Các danh mục: {sorted(list(cac_danh_muc))}")

```

=== BÀI TẬP 1: PHÂN TÍCH DỮ LIỆU BÁN HÀNG ===

Shop: Shop Online ABC

Số sản phẩm: 5

Số đơn hàng: 5

Địa chỉ: 123 Đường XYZ, Quận 1, TP.HCM

Ngày mở shop: 15/5/2020

Sản phẩm ID 3: Giày sneaker - 500,000 VNĐ

=== DOANH THU TỪNG ĐƠN HÀNG ===

Đơn 1: Áo thun x2 = 300,000 VNĐ

Đơn 2: Giày sneaker x1 = 500,000 VNĐ

Đơn 3: Quần jean x1 = 300,000 VNĐ

Đơn 4: Túi xách x3 = 600,000 VNĐ

Đơn 5: Áo thun x1 = 150,000 VNĐ

Tổng doanh thu: 1,850,000 VNĐ

=== KHÁCH HÀNG ===

Tổng số khách hàng duy nhất: 4

Danh sách: ['Lê C', 'Nguyễn A', 'Phạm D', 'Trần B']

=== THỐNG KÊ KHÁCH HÀNG ===

Nguyễn A: 2 lần mua

Trần B: 1 lần mua

Lê C: 1 lần mua

Phạm D: 1 lần mua

=== DANH MỤC SẢN PHẨM ===

Các danh mục: ['Giày dép', 'Phụ kiện', 'Thời trang']

2.2 Bài tập 2: Hệ thống thư viện đơn giản

```
In [5]: print("\n=== BÀI TẬP 2: HỆ THỐNG THƯ VIỆN ===")

# Cấu trúc dữ liệu thư viện
thu_vien = {
    "thong_tin": {
        "ten": "Thư viện Trung tâm",
        "dia_chi": ("456 Đường ABC", "Quận 3", "TP.HCM"),
        "gio_mo_cua": (8, 0), # 8:00
        "gio_dong_cua": (20, 0) # 20:00
    },
    "sach": [
        {"id": 1, "ten": "Lập trình Python", "tac_gia": "Nguyễn A", "the_loai": "Công"},
        {"id": 2, "ten": "Khoa học dữ liệu", "tac_gia": "Trần B", "the_loai": "Công"},
        {"id": 3, "ten": "Toán học cơ bản", "tac_gia": "Lê C", "the_loai": "Toán học"},
        {"id": 4, "ten": "Lịch sử Việt Nam", "tac_gia": "Phạm D", "the_loai": "Lịch"},
        {"id": 5, "ten": "Trí tuệ nhân tạo", "tac_gia": "Nguyễn A", "the_loai": "Công"},
    ],
    "doc_gia": [
        {"id": 1, "ten": "Học sinh A", "tuoi": 18, "loai_the": "Sinh viên"},
        {"id": 2, "ten": "Giáo viên B", "tuoi": 35, "loai_the": "Giảng viên"},
    ]
}
```



```

        {"id": 3, "ten": "Học sinh C", "tuoi": 19, "loai_the": "Sinh viên"}
    ],

    "muon_sach": [
        {"doc_gia_id": 1, "sach_id": 1, "ngay_muon": (2024, 1, 15), "han_tra": (202
        {"doc_gia_id": 2, "sach_id": 3, "ngay_muon": (2024, 1, 10), "han_tra": (202
        {"doc_gia_id": 1, "sach_id": 4, "ngay_muon": (2024, 1, 20), "han_tra": (202
    ]
}

print(f"Tên thư viện: {thu_vien['thong_tin']['ten']}")
print(f"Số sách: {len(thu_vien['sach'])}")
print(f"Số độc giả: {len(thu_vien['doc_gia'])}")
print(f"Số lượt mượn: {len(thu_vien['muon_sach'])}")

# Unpacking giờ mở cửa
gio_mo, phut_mo = thu_vien["thong_tin"]["gio_mo_cua"]
gio_dong, phut_dong = thu_vien["thong_tin"]["gio_dong_cua"]
print(f"Giờ hoạt động: {gio_mo:02d}:{phut_mo:02d} - {gio_dong:02d}:{phut_dong:02d}")

# Tạo Set chứa tất cả thể loại sách
tat_ca_the_loai = set()
for sach in thu_vien["sach"]:
    tat_ca_the_loai.add(sach["the_loai"])

print(f"\n=== THỂ LOẠI SÁCH ===")
print(f"Các thể loại: {sorted(list(tat_ca_the_loai))}")
print(f"Số thể loại: {len(tat_ca_the_loai)}")

# Tạo Dictionary để tra cứu sách theo ID
sach_dict = {}
for sach in thu_vien["sach"]:
    sach_dict[sach["id"]] = sach

# Tạo Dictionary để tra cứu độc giả theo ID
doc_gia_dict = {}
for dg in thu_vien["doc_gia"]:
    doc_gia_dict[dg["id"]] = dg

# Liệt kê sách có sẵn
print("\n=== SÁCH CÓ SẴN ===")
sach_co_san = []
for sach in thu_vien["sach"]:
    if sach["co_san"] == True:
        sach_co_san.append(sach)

print(f"Số sách có sẵn: {len(sach_co_san)}/{len(thu_vien['sach'])}")
for sach in sach_co_san:
    print(f"    - {sach['ten']} ({sach['tac_gia']})")

# Thống kê ai đang mượn sách gì
print("\n=== TÌNH TRẠNG MƯỢN SÁCH ===")
for muon in thu_vien["muon_sach"]:
    ten_doc_gia = doc_gia_dict[muon["doc_gia_id"]]["ten"]
    ten_sach = sach_dict[muon["sach_id"]]["ten"]

```

```

# Unpacking ngày mượn và hạn trả
nam_muon, thang_muon, ngay_muon = muon["ngay_muon"]
nam_tra, thang_tra, ngay_tra = muon["han_tra"]

print(f" - {ten_doc_gia} mượn '{ten_sach}'")
print(f"    Ngày mượn: {ngay_muon}/{thang_muon}/{nam_muon}")
print(f"    Hạn trả: {ngay_tra}/{thang_tra}/{nam_tra}")
print()

# Đếm số sách từng tác giả viết
thong_ke_tac_gia = {}
for sach in thu_vien["sach"]:
    tac_gia = sach["tac_gia"]
    if tac_gia not in thong_ke_tac_gia:
        thong_ke_tac_gia[tac_gia] = 0
    thong_ke_tac_gia[tac_gia] += 1

print("=== THỐNG KÊ TÁC GIẢ ===")
for tac_gia, so_sach in thong_ke_tac_gia.items():
    print(f"{tac_gia}: {so_sach} cuốn")

```

=== BÀI TẬP 2: HỆ THỐNG THƯ VIỆN ===

Tên thư viện: Thư viện Trung tâm

Số sách: 5

Số độc giả: 3

Số lượt mượn: 3

Giờ hoạt động: 08:00 - 20:00

=== THỂ LOẠI SÁCH ===

Các thể loại: ['Công nghệ', 'Lịch sử', 'Toán học']

Số thể loại: 3

=== SÁCH CÓ SẴN ===

Số sách có sẵn: 3/5

- Lập trình Python (Nguyễn A)
- Toán học cơ bản (Lê C)
- Lịch sử Việt Nam (Phạm D)

=== TÌNH TRẠNG MƯỢN SÁCH ===

- Học sinh A mượn 'Lập trình Python'
Ngày mượn: 15/1/2024
Hạn trả: 15/2/2024
- Giáo viên B mượn 'Toán học cơ bản'
Ngày mượn: 10/1/2024
Hạn trả: 10/2/2024
- Học sinh A mượn 'Lịch sử Việt Nam'
Ngày mượn: 20/1/2024
Hạn trả: 20/2/2024

=== THỐNG KÊ TÁC GIẢ ===

Nguyễn A: 2 cuốn

Trần B: 1 cuốn

Lê C: 1 cuốn

Phạm D: 1 cuốn

3. CHUẨN BỊ CHO MINI PROJECT: CONTACT MANAGER

3.1 Phân tích yêu cầu Contact Manager

Mini Project: Hệ thống quản lý danh bạ (Contact Manager)

Mô tả dự án:

- Xây dựng ứng dụng quản lý danh bạ điện thoại
- Lưu trữ thông tin liên lạc: tên, số điện thoại, email, địa chỉ
- Các chức năng: thêm, sửa, xóa, tìm kiếm, hiển thị
- Menu điều hướng đơn giản

Cấu trúc dữ liệu cần dùng:

- **List:** Lưu danh sách tất cả contacts
- **Dictionary:** Thông tin từng contact
- **Tuple:** Địa chỉ (đường, quận, thành phố)
- **Set:** Thống kê các thành phố, nhóm contact

Kiến thức áp dụng:

- Variables, operators, if/else
- Loops (for, while)
- List methods (append, remove, etc.)
- Dictionary operations
- Tuple unpacking
- Set operations
- Functions cơ bản

```
In [6]: print("\n=== CHUẨN BỊ CHO MINI PROJECT: CONTACT MANAGER ===")

print("\nMô tả dự án:")
print("- Xây dựng ứng dụng quản lý danh bạ điện thoại")
print("- Lưu trữ thông tin liên lạc: tên, số điện thoại, email, địa chỉ")
print("- Các chức năng: thêm, sửa, xóa, tìm kiếm, hiển thị")
print("- Menu điều hướng đơn giản")

print("\nCấu trúc dữ liệu cần dùng:")
print("- LIST: Lưu danh sách tất cả contacts")
print("- DICTIONARY: Thông tin từng contact")
print("- TUPLE: Địa chỉ (đường, quận, thành phố)")
print("- SET: Thống kê các thành phố, nhóm contact")

# Thiết kế cấu trúc dữ liệu cho Contact Manager
print("\n=== THIẾT KẾ CẤU TRÚC DỮ LIỆU CONTACT MANAGER ===")
```

```

# Cấu trúc một contact (Dictionary)
contact_mau = {
    "id": 1,
    "ten": "Nguyễn Văn An",
    "so_dien_thoai": "0123456789",
    "email": "an.nguyen@email.com",
    "dia_chi": ("123 Đường ABC", "Quận 1", "TP.HCM"), # Tuple
    "nhom": "Bạn bè",
    "ghi_chu": "Bạn học cùng lớp"
}

print("1. Cấu trúc một contact:")
for key, value in contact_mau.items():
    print(f"    {key}: {value}")

# Danh sách tất cả contacts (List chứa Dictionary)
danh_ba = [contact_mau] # Sẽ chứa nhiều contact

print(f"\n2. Danh bạ (List): {len(danh_ba)} contact(s)")

# Thống kê (Set để Lưu dữ liệu không trùng)
cac_nhom = set() # Các nhóm contact
cac_thanh_pho = set() # Các thành phố

for contact in danh_ba:
    cac_nhom.add(contact["nhom"])
    # Unpacking địa chỉ
    duong, quan, thanh_pho = contact["dia_chi"]
    cac_thanh_pho.add(thanh_pho)

print(f"\n3. Thống kê:")
print(f"    Các nhóm: {cac_nhom}")
print(f"    Các thành phố: {cac_thanh_pho}")

# Cấu trúc hoàn chỉnh
contact_manager = {
    "contacts": [], # List chứa tất cả contacts
    "thong_ke": {
        "tong_contact": 0,
        "cac_nhom": set(),
        "cac_thanh_pho": set()
    },
    "cai_dat": {
        "sap_xep_theo": "ten", # "ten", "nhom", "thanh_pho"
        "hien_thi_chi_tiet": True
    }
}

print(f"\n4. Cấu trúc hoàn chỉnh Contact Manager đã sẵn sàng!")

```

=== CHUẨN BỊ CHO MINI PROJECT: CONTACT MANAGER ===

Mô tả dự án:

- Xây dựng ứng dụng quản lý danh bạ điện thoại
- Lưu trữ thông tin liên lạc: tên, số điện thoại, email, địa chỉ
- Các chức năng: thêm, sửa, xóa, tìm kiếm, hiển thị
- Menu điều hướng đơn giản

Cấu trúc dữ liệu cần dùng:

- LIST: Lưu danh sách tất cả contacts
- DICTIONARY: Thông tin từng contact
- TUPLE: Địa chỉ (đường, quận, thành phố)
- SET: Thống kê các thành phố, nhóm contact

=== THIẾT KẾ CẤU TRÚC DỮ LIỆU CONTACT MANAGER ===

1. Cấu trúc một contact:

```
id: 1
ten: Nguyễn Văn An
so_dien_thoai: 0123456789
email: an.nguyen@email.com
dia_chi: ('123 Đường ABC', 'Quận 1', 'TP.HCM')
nhom: Bạn bè
ghi_chu: Bạn học cùng lớp
```

2. Danh bạ (List): 1 contact(s)

3. Thống kê:

```
Các nhóm: {'Bạn bè'}
Các thành phố: {'TP.HCM'}
```

4. Cấu trúc hoàn chỉnh Contact Manager đã sẵn sàng!

4. KẾ HOẠCH THỰC HIỆN MINI PROJECT

4.1 Roadmap cho buổi chiều

Kế hoạch buổi chiều W27-T6 (14:00-17:00): "Mini project: Contact Manager"

Bước 1 (30 phút): Thiết lập project

- Tạo cấu trúc dữ liệu chính
- Viết các hàm cơ bản: thêm, hiển thị
- Test với dữ liệu mẫu

Bước 2 (45 phút): Chức năng tìm kiếm

- Tìm theo tên, nhóm, thành phố
- Hiển thị kết quả tìm kiếm
- Test các trường hợp tìm kiếm

Bước 3 (45 phút): Chức năng quản lý

- Sửa thông tin contact
- Xóa contact
- Thống kê và báo cáo

Bước 4 (30 phút): Menu và hoàn thiện

- Tạo menu điều hướng
- Xử lý lỗi cơ bản
- Test toàn bộ ứng dụng

Bước 5 (30 phút): Trình bày và đánh giá

- Demo ứng dụng
- Code review
- Thảo luận cải tiến

5. TÓM TẮT VÀ CHUẨN BỊ

5.1 Kiến thức đã ôn tập

Trong buổi học này, chúng ta đã:

1. Ôn tập toàn bộ cấu trúc dữ liệu:

- List: Danh sách có thứ tự, thay đổi được
- Dictionary: Ánh xạ key-value, truy cập nhanh
- Tuple: Bộ dữ liệu không thay đổi, unpacking
- Set: Tập hợp không trùng lặp, phép toán tập hợp

2. Ma trận quyết định lựa chọn cấu trúc:

- Phân tích yêu cầu: thứ tự, thay đổi, trùng lặp, kiểu truy cập
- Gợi ý cấu trúc phù hợp cho từng tình huống
- Thực hành với các bài toán thực tế

3. Bài tập tổng hợp:

- Phân tích dữ liệu bán hàng
- Hệ thống thư viện
- Kết hợp nhiều cấu trúc trong một ứng dụng

4. Chuẩn bị Mini Project:

- Thiết kế Contact Manager
- Viết các hàm cơ bản
- Lập kế hoạch thực hiện

Kiến thức nền tảng đã sử dụng:

- Variables, Data types, Operators
- If/else statements, Loops (for, while)
- List operations và methods
- Dictionary operations
- Tuple unpacking
- Set operations
- Functions cơ bản



TODO 6: Kiểm tra sẵn sàng cho Mini Project

```
In [7]: # TODO 6: Kiểm tra độ sẵn sàng cho Mini Project

print("=== KIỂM TRA ĐỘ SẴN SÀNG CHO MINI PROJECT ===")
print()

# Checklist kiến thức
kien_thuc_checklist = {
    "Tạo và thao tác List": False,
    "Tạo và thao tác Dictionary": False,
    "Sử dụng Tuple và unpacking": False,
    "Sử dụng Set và các phép toán": False,
    "Viết functions cơ bản": False,
    "Sử dụng loops và if/else": False,
    "Kết hợp nhiều cấu trúc dữ liệu": False,
    "Thiết kế cấu trúc dữ liệu phức tạp": False
}

# TODO: Đánh dấu True cho những kiến thức bạn đã nắm vững
# kien_thuc_checklist["Tạo và thao tác List"] = True
# kien_thuc_checklist["Tạo và thao tác Dictionary"] = True
# ... (đánh dấu các kiến thức khác)

# Kiểm tra và hiển thị kết quả
so_kien_thuc_ok = sum(kien_thuc_checklist.values())
tong_so_kien_thuc = len(kien_thuc_checklist)

print("Kiến thức đã nắm vững:")
for kien_thuc, da_ok in kien_thuc_checklist.items():
    status = "✅" if da_ok else "❌"
    print(f" {status} {kien_thuc}")

ti_le_san_sang = (so_kien_thuc_ok / tong_so_kien_thuc) * 100
print(f"\n📊 Độ sẵn sàng: {so_kien_thuc_ok}/{tong_so_kien_thuc} ({ti_le_san_sang:.1f}%)")

if ti_le_san_sang >= 80:
    print("🎉 Tuyệt vời! Bạn đã sẵn sàng cho Mini Project!")
elif ti_le_san_sang >= 60:
    print("👍 Khá tốt! Hãy ôn lại một số kiến thức còn thiếu.")
else:
    print("📖 Cần ôn tập thêm trước khi bắt đầu Mini Project.")
```

```
# Gợi ý ôn tập
can_on_tap = [kien_thuc for kien_thuc, da_ok in kien_thuc_checklist.items() if not
if can_on_tap:
    print("\n📖 Cần ôn tập:")
    for kien_thuc in can_on_tap:
        print(f"    - {kien_thuc}")

print("\n💡 Hoàn thành TODO 6 để kiểm tra sự chuẩn bị!")
```

=== KIỂM TRA ĐỘ SẴN SÀNG CHO MINI PROJECT ===

Kiến thức đã nắm vững:

- ❌ Tạo và thao tác List
- ❌ Tạo và thao tác Dictionary
- ❌ Sử dụng Tuple và unpacking
- ❌ Sử dụng Set và các phép toán
- ❌ Viết functions cơ bản
- ❌ Sử dụng loops và if/else
- ❌ Kết hợp nhiều cấu trúc dữ liệu
- ❌ Thiết kế cấu trúc dữ liệu phức tạp

📊 Độ sẵn sàng: 0/8 (0.0%)

📖 Cần ôn tập thêm trước khi bắt đầu Mini Project.

📖 Cần ôn tập:

- Tạo và thao tác List
- Tạo và thao tác Dictionary
- Sử dụng Tuple và unpacking
- Sử dụng Set và các phép toán
- Viết functions cơ bản
- Sử dụng loops và if/else
- Kết hợp nhiều cấu trúc dữ liệu
- Thiết kế cấu trúc dữ liệu phức tạp

💡 Hoàn thành TODO 6 để kiểm tra sự chuẩn bị!

5.2 Chuẩn bị cho buổi chiều

Buổi chiều W27-T6 (14:00-17:00): Mini project: Contact Manager

Những gì cần chuẩn bị:

1. Kiến thức kỹ thuật:

- Ôn lại cách tạo và thao tác với List, Dictionary, Tuple, Set
- Ôn lại cách viết functions và sử dụng parameters
- Ôn lại cách sử dụng loops và if/else statements

2. Tư duy thiết kế:

- Cách phân tích yêu cầu thành cấu trúc dữ liệu
- Cách chia nhỏ bài toán thành các functions
- Cách tổ chức code có hệ thống

3. Kỹ năng thực hành:

- Debug và sửa lỗi cơ bản
- Test functions với dữ liệu mẫu
- Trình bày và giải thích code

Mục tiêu buổi chiều:

- Hoàn thành Contact Manager với đầy đủ chức năng
- Áp dụng tất cả kiến thức đã học trong tuần 27
- Chuẩn bị cho Mini Project lớn hơn ở cuối tuần

Tiêu chí đánh giá:

- Tính đúng đắn của code
- Cách sử dụng cấu trúc dữ liệu
- Tổ chức code và đặt tên biến
- Khả năng giải thích và trình bày



Chúc các bạn thành công với Mini Project Contact Manager!