

W26 - Thứ 5 Buổi Chiều: Vẽ Patterns với Loops và Counting Exercises

Mục tiêu học tập:

- Sử dụng nested loops để vẽ các patterns phức tạp
- Thực hành counting exercises với for loops
- Tạo ra các simple iterations để giải quyết bài toán
- Phát triển tư duy logic và khả năng debug

1. Ôn tập nhanh For Loops và Range

Trước khi bắt đầu vẽ patterns, chúng ta ôn lại kiến thức buổi sáng:

```
In [ ]: # Ôn tập cơ bản
print("=== ÔN TẬP FOR LOOPS ===")

# Đếm từ 1 đến 5
print("Đếm từ 1 đến 5:")
for i in range(1, 6):
    print(i, end=" ")

print("\n\nĐếm ngược từ 5 về 1:")
for i in range(5, 0, -1):
    print(i, end=" ")

print("\n\nCác số chẵn từ 0 đến 10:")
for i in range(0, 11, 2):
    print(i, end=" ")
```

2. Nested Loops - Vòng lặp lồng nhau

Nested loops là kỹ thuật đặt một vòng lặp bên trong vòng lặp khác. Đây là công cụ mạnh mẽ để tạo patterns 2D.

```
In [ ]: # Ví dụ đơn giản về nested loops
print("Ma trận 3x3 với tọa độ:")

for hang in range(3):
    for cot in range(3):
        print(f"({hang},{cot})", end=" ")
    print() # Xuống dòng sau mỗi hàng
```

3. Patterns cơ bản - Hình chữ nhật và Vuông

Bắt đầu với những patterns đơn giản nhất:

```
In [ ]: # Pattern 1: Hình chữ nhật đặc
print("=== HÌNH CHỮ NHẬT ĐẶC ===")
rong = int(input("Nhập chiều rộng: "))
cao = int(input("Nhập chiều cao: "))

for i in range(cao):
    for j in range(rong):
        print("* ", end="")
    print()
```

```
In [ ]: # Pattern 2: Hình chữ nhật rỗng
print("\n=== HÌNH CHỮ NHẬT RỖNG ===")
rong = int(input("Nhập chiều rộng: "))
cao = int(input("Nhập chiều cao: "))

for i in range(cao):
    for j in range(rong):
        # Vẽ viền: hàng đầu, hàng cuối, cột đầu, cột cuối
        if i == 0 or i == cao-1 or j == 0 or j == rong-1:
            print("* ", end="")
        else:
            print("  ", end="") # 2 khoảng trắng
    print()
```

4. Patterns tam giác - Từ đơn giản đến phức tạp

```
In [ ]: # Pattern 3: Tam giác vuông trái
print("=== TAM GIÁC VUÔNG TRÁI ===")
n = int(input("Nhập số dòng: "))

for i in range(1, n + 1):
    for j in range(i):
        print("* ", end="")
    print()
```

```
In [ ]: # Pattern 4: Tam giác vuông phải
print("\n=== TAM GIÁC VUÔNG PHẢI ===")
n = int(input("Nhập số dòng: "))

for i in range(1, n + 1):
    # In khoảng trắng
    for j in range(n - i):
        print("  ", end="")
    # In dấu sao
    for k in range(i):
        print("* ", end="")
    print()
```

```
In [ ]: # Pattern 5: Tam giác cân
print("\n=== TAM GIÁC CÂN ===")
```

```
n = int(input("Nhập số dòng: "))

for i in range(1, n + 1):
    # In khoảng trắng để căn giữa
    for j in range(n - i):
        print(" ", end="")
    # In dấu sao
    for k in range(2 * i - 1):
        print("*", end="")
    print()
```

```
In [ ]: # Pattern 6: Tam giác ngược
print("\n=== TAM GIÁC NGƯỢC ===")
n = int(input("Nhập số dòng: "))

for i in range(n, 0, -1):
    # In khoảng trắng
    for j in range(n - i):
        print(" ", end="")
    # In dấu sao
    for k in range(2 * i - 1):
        print("*", end="")
    print()
```

5. Patterns với Số - Number Patterns

```
In [ ]: # Pattern 7: Tam giác số tăng dần
print("=== TAM GIÁC SỐ TĂNG DẦN ===")
n = int(input("Nhập số dòng: "))

for i in range(1, n + 1):
    for j in range(1, i + 1):
        print(j, end=" ")
    print()
```

```
In [ ]: # Pattern 8: Tam giác số lặp
print("\n=== TAM GIÁC SỐ LẶP ===")
n = int(input("Nhập số dòng: "))

for i in range(1, n + 1):
    for j in range(i):
        print(i, end=" ")
    print()
```

```
In [ ]: # Pattern 9: Bảng cửu chương dạng ma trận
print("\n=== BẢNG CỬU CHƯƠNG MA TRẬN ===")

print(" ", end="")
for i in range(1, 11):
    print(f"{i:4}", end="")
print()

print(" " + "-" * 40)
```

```
for i in range(1, 11):
    print(f"{i:2} |", end="")
    for j in range(1, 11):
        print(f"{i*j:4}", end="")
    print()
```

6. Counting Exercises - Bài tập đếm

Thực hành các bài toán đếm sử dụng for loops:

```
In [ ]: # Counting Exercise 1: Đếm chữ số
print("=== ĐẾM CHỮ SỐ ===")
so = input("Nhập một số: ")

# Đếm từng loại chữ số
dem_chan = 0
dem_le = 0

for chu_so in so:
    if chu_so.isdigit():
        if int(chu_so) % 2 == 0:
            dem_chan += 1
        else:
            dem_le += 1

print(f"Số chữ số chẵn: {dem_chan}")
print(f"Số chữ số lẻ: {dem_le}")
print(f"Tổng chữ số: {dem_chan + dem_le}")
```

[illegible]

```
print(f"Số: {dem_so}")
print(f"Khoảng trắng: {dem_khoang_trang}")
```

```
In [ ]: # Counting Exercise 3: Đếm số trong khoảng
print("\n=== ĐẾM SỐ TRONG KHOẢNG ===")
bat_dau = int(input("Số bắt đầu: "))
ket_thuc = int(input("Số kết thúc: "))

dem_nguyen_to = 0
dem_chinh_phuong = 0
dem_hoan_hao = 0

def la_nguyen_to(n):
    if n < 2:
        return False
    for i in range(2, int(n**0.5) + 1):
        if n % i == 0:
            return False
    return True

def la_chinh_phuong(n):
    can_bac_hai = int(n**0.5)
    return can_bac_hai * can_bac_hai == n

def la_hoan_hao(n):
    if n < 2:
        return False
    tong_uoc = 1
    for i in range(2, int(n**0.5) + 1):
        if n % i == 0:
            tong_uoc += i
            if i != n // i:
                tong_uoc += n // i
    return tong_uoc == n

print(f"\nPhân tích từ {bat_dau} đến {ket_thuc}:")
for so in range(bat_dau, ket_thuc + 1):
    if la_nguyen_to(so):
        dem_nguyen_to += 1
    if la_chinh_phuong(so):
        dem_chinh_phuong += 1
    if la_hoan_hao(so):
        dem_hoan_hao += 1

print(f"Số nguyên tố: {dem_nguyen_to}")
print(f"Số chính phương: {dem_chinh_phuong}")
print(f"Số hoàn hảo: {dem_hoan_hao}")
```

7. Simple Iterations - Lặp đơn giản

Các ví dụ về lặp đơn giản để giải quyết bài toán thực tế:

```
In [ ]: # Simple Iteration 1: Tìm số lớn nhất
print("=== TÌM SỐ LỚN NHẤT ===")
```

```

n = int(input("Nhập số lượng số: "))
so_lon_nhat = float('-inf') # Khởi tạo với giá trị rất nhỏ

for i in range(n):
    so = float(input(f"Nhập số thứ {i+1}: "))
    if so > so_lon_nhat:
        so_lon_nhat = so

print(f"Số lớn nhất là: {so_lon_nhat}")

```

```

In [ ]: # Simple Iteration 2: Tính lãi suất kép
print("\n=== TÍNH LÃI SUẤT KÉP ===")
von_ban_dau = float(input("Số tiền ban đầu: "))
lai_suat = float(input("Lãi suất hàng năm (%): ")) / 100
so_nam = int(input("Số năm: "))

so_tien = von_ban_dau
print(f"\nNăm 0: {so_tien:,.0f} VND")

for nam in range(1, so_nam + 1):
    so_tien = so_tien * (1 + lai_suat)
    print(f"Năm {nam}: {so_tien:,.0f} VND")

lai_thu_duoc = so_tien - von_ban_dau
print(f"\nLãi thu được: {lai_thu_duoc:,.0f} VND")

```

```

In [ ]: # Simple Iteration 3: Máy bán hàng tự động
print("\n=== MÁY BÁN HÀNG TỰ ĐỘNG ===")
print("Menu:")
print("1. Nước ngọt - 15,000 VND")
print("2. Bánh mì - 25,000 VND")
print("3. Cà phê - 20,000 VND")
print("4. Kẹo - 5,000 VND")

gia_san_pham = [0, 15000, 25000, 20000, 5000] # Index 0 không dùng
ten_san_pham = ["", "Nước ngọt", "Bánh mì", "Cà phê", "Kẹo"]

tong_tien = 0
gio_hang = []

while True:
    lua_chon = input("\nChọn sản phẩm (1-4) hoặc 'q' để thanh toán: ")

    if lua_chon.lower() == 'q':
        break

    try:
        ma_sp = int(lua_chon)
        if 1 <= ma_sp <= 4:
            so_luong = int(input(f"Số lượng {ten_san_pham[ma_sp]}: "))
            thanh_tien = gia_san_pham[ma_sp] * so_luong
            tong_tien += thanh_tien
            gio_hang.append(f"{ten_san_pham[ma_sp]} x{so_luong}: {thanh_tien:,.0} VND")
            print(f"Đã thêm {so_luong} {ten_san_pham[ma_sp]} - {thanh_tien:,.0} VND")
        else:

```

```

        print("Lựa chọn không hợp lệ!")
    except ValueError:
        print("Vui lòng nhập số hợp lệ!")

# Hiển thị hóa đơn
if gio_hang:
    print("\n" + "="*30)
    print("          HÓA ĐƠN")
    print("="*30)
    for item in gio_hang:
        print(item)
    print("-"*30)
    print(f"TỔNG CỘNG: {tong_tien:,} VND")
    print("="*30)
else:
    print("Cảm ơn bạn đã ghé thăm!")

```

8. Patterns nâng cao - Thử thách sáng tạo

```

In [ ]: # Advanced Pattern 1: Hình thoi
print("=== HÌNH THOI ===")
n = int(input("Nhập số dòng cho nửa trên (số lẻ): "))

# Nửa trên của hình thoi
for i in range(1, n + 1, 2):
    khoang_trang = (n - i) // 2
    print(" " * khoang_trang + "*" * i)

# Nửa dưới của hình thoi
for i in range(n - 2, 0, -2):
    khoang_trang = (n - i) // 2
    print(" " * khoang_trang + "*" * i)

```

```

In [ ]: # Advanced Pattern 2: Tam giác Pascal
print("\n=== TAM GIÁC PASCAL ===")
n = int(input("Nhập số dòng: "))

for i in range(n):
    # In khoảng trắng để căn giữa
    print(" " * (n - i - 1), end="")

    # Tính và in các số trong tam giác Pascal
    so = 1
    for j in range(i + 1):
        print(f"{so:3}", end=" ")
        so = so * (i - j) // (j + 1)
    print()

```

```

In [ ]: # Advanced Pattern 3: Đồng hồ cát
print("\n=== ĐỒNG HỒ CÁT ===")
n = int(input("Nhập kích thước (số lẻ): "))

# Nửa trên
for i in range(n, 0, -2):

```

```

    khoang_trang = (n - i) // 2
    print(" " * khoang_trang + "*" * i)

# Nửa dưới (bỏ dòng giữa)
for i in range(3, n + 1, 2):
    khoang_trang = (n - i) // 2
    print(" " * khoang_trang + "*" * i)

```

9. Mini Games với Loops

```

In [ ]: # Mini Game 1: Đoán số với giới hạn lần thử
import random

print("=== TRÒ CHƠI ĐOÁN SỐ ===")
so_bi_mat = random.randint(1, 100)
so_lan_thu = 7

print(f"Tôi đã nghĩ ra một số từ 1 đến 100. Bạn có {so_lan_thu} lần đoán!")

for lan_thu in range(1, so_lan_thu + 1):
    print(f"\nLần thử {lan_thu}/{so_lan_thu}:")
    try:
        doan = int(input("Số bạn đoán: "))

        if doan == so_bi_mat:
            print(f"🎉 Chúc mừng! Bạn đã đoán đúng số {so_bi_mat}!")
            print(f"Bạn đã thắng trong {lan_thu} lần thử!")
            break
        elif doan < so_bi_mat:
            print("📉 Số của tôi lớn hơn!")

```