# C++ Programming Practice Set

## Control Flow Statements & Modular Programming with Functions

---

## EASY QUESTIONS (1-3)

### Question 1: Traffic Light Simulator

Write a C++ program that:

- Takes current traffic light color (R/G/Y) and duration as input
- Uses if-else statements to determine the next light color and timing:
  - Red (30 sec) → Green (45 sec)
  - Green (45 sec) → Yellow (5 sec)
  - Yellow (5 sec) → Red (30 sec)
- Uses a for loop to simulate countdown timer
- Displays appropriate messages for drivers

**Sample Input:**

```
Enter current light (R/G/Y): G
Enter remaining time: 10
```

**Expected Output:**

```
Current: GREEN light
Countdown: 10 9 8 7 6 5 4 3 2 1 0
YELLOW light activated for 5 seconds
Next: RED light will activate after 5 seconds
```

---

### Question 2: Password Strength Checker

Write a C++ program that:

- Takes a password string as input
- Uses while loop to analyze each character
- Counts uppercase, lowercase, digits, and special characters
- Uses nested if statements to determine password strength:
  - Weak: Length < 6 or missing 2+ categories
  - Medium: Length 6-8 with 3+ categories
  - Strong: Length 9+ with all 4 categories
- Displays detailed analysis

**Sample Input:**

```
Enter password: MyPass123!
```

**Expected Output:**

```
Password Analysis:
Length: 10 characters
Uppercase: 2
Lowercase: 4
Digits: 3
Special chars: 1
Strength: STRONG
Recommendation: Password meets all security criteria!
```

---

### Question 3: Vending Machine Simulator

Write a C++ program that:

- Displays available items with prices using arrays and loops
- Uses switch-case to handle item selection
- Uses do-while loop for money insertion until sufficient amount
- Calculates and dispenses change using nested loops
- Continues until user selects exit option

**Sample Menu:**

```
===== VENDING MACHINE =====
1. Chips - $2.50
2. Soda - $1.75
3. Candy - $1.25
4. Water - $1.00
5. Exit

Select item: 2
Selected: Soda ($1.75)

Insert money (enter 0 when done):
Enter amount: 1.00
Total inserted: $1.00 (Need $0.75 more)
Enter amount: 1.00
Total inserted: $2.00

Dispensing: Soda
Change: $0.25
Thank you!
```

---

## MEDIUM QUESTIONS (4-5)

### Question 4: ASCII Art Generator Functions

Write a C++ program with the following functions:

- `void drawBox(int width, int height)` - draws rectangular box with borders
- `void drawDiamond(int size)` - draws diamond pattern with given size

- `void drawSpiral(int n)` - draws number spiral pattern
- `char getPatternChoice()` - gets and validates user pattern choice
- `int getDimension(string prompt)` - gets positive integer with validation
- Use nested loops within each drawing function

**Expected Output Examples:**

```
Box (5x3):
*****
*   *
*****

Diamond (size 3):
  *
 ***
*****
 ***
  *

Spiral (4x4):
1  2  3  4
12 13 14 5
11 16 15 6
10 9  8  7
```

---

## Question 5: File Processing Simulator

Write a C++ program with modular functions that simulates file operations:

- `bool createFile(string filename)` - simulates file creation with validation
- `void processFile(string filename)` - processes file line by line using loops
- `int countWords(string line)` - counts words in a line using character analysis
- `void generateReport()` - creates summary using statistical functions
- `void backupFiles()` - simulates backup process with progress indication
- Use arrays to store multiple filenames and their properties

**Sample Program Flow:**

```
===== FILE PROCESSING SYSTEM =====
1. Create File
2. Process File
3. Generate Report
4. Backup Files
5. Exit

Enter choice: 2
Available files:
1. document.txt (245 lines)
2. data.csv (150 lines)
3. report.doc (89 lines)

Select file: 1
Processing document.txt...
Line 1: 15 words processed ████████████████████ 20%
Line 2: 8 words processed  ████████████████████ 40%
...
Processing complete!
Total words: 3,450
Average words per line: 14.1
```

---

## HARD QUESTION (6)

### Question 6: Hotel Reservation Management System

Write a comprehensive C++ program for hotel room management using functions and control structures:

**Core Functions Required:**

1. `void initializeRooms()` - sets up room availability matrix
2. `void displayRoomStatus()` - shows room grid with availability using nested loops
3. `bool checkAvailability(int roomType, int nights)` - validates room availability
4. `double calculatePrice(int roomType, int nights, bool isWeekend)` - computes total cost
5. `void makeReservation()` - handles booking process with validation loops
6. `void checkInOut()` - manages check-in/out with date calculations
7. `void cancelReservation()` - processes cancellations with confirmation loops
8. `void generateOccupancyReport()` - creates detailed statistics
9. `void maintenanceMode()` - toggles room maintenance status
10. `int findAvailableRoom(int type)` - searches for available rooms

**Room Configuration:**

```cpp
// Room types: 1=Single($100), 2=Double($150), 3=Suite($300)
// Hotel has 10 floors, 20 rooms per floor (rooms 101-200)
int roomStatus[10][20]; // 0=Available, 1=Occupied, 2=Maintenance
```

**Complex Control Flow Requirements:**

- Use nested for loops for room grid operations
- Implement date validation using while loops and conditionals
- Use switch-case with fallthrough for seasonal pricing
- Implement search algorithms using nested loops
- Use break and continue for optimization
- Implement input validation with do-while loops

**Sample Program Execution:**

```
===== HOTEL RESERVATION SYSTEM =====
Current Date: March 15, 2024

1. View Room Status
2. Make Reservation
3. Check In/Out
4. Cancel Reservation
5. Occupancy Report
6. Maintenance Mode
7. Exit

Enter choice: 2

=== MAKE RESERVATION ===
Room Types:
1. Single Room ($100/night)
2. Double Room ($150/night)
3. Suite ($300/night)

Select room type: 2
Enter check-in date (DD/MM/YYYY): 20/03/2024
Enter number of nights: 3

Searching for available Double rooms...
Available rooms found: 205, 207, 315, 318

Calculating price...
Base cost: $150 x 3 nights = $450
Weekend surcharge (1 night): $25
Tax (10%): $47.50
Total: $522.50

Confirm reservation? (Y/N): Y

Select preferred room:
1. Room 205 (Floor 2, City view)
2. Room 207 (Floor 2, Garden view)
3. Room 315 (Floor 3, City view)
4. Room 318 (Floor 3, Pool view)

Choice: 4
Reservation confirmed!
Booking ID: HTL2024001
Room: 318
Guest: [Enter guest details...]

Enter choice: 5

=== OCCUPANCY REPORT ===
Date: March 15, 2024

Floor-wise Occupancy:
Floor 1: █████████████████ 85% (17/20 rooms)
Floor 2: ████████████████  80% (16/20 rooms)
Floor 3: ███████████████     75% (15/20 rooms)
...

Room Type Statistics:
Single Rooms: 65% occupied (65/100)
Double Rooms: 78% occupied (78/100)
Suites: 45% occupied (9/20)

Revenue Analysis:
Today's bookings: $15,650
This month: $342,500
Pending check-outs: 23 rooms
```

**Advanced Features to Implement:**

- Date arithmetic for stay duration calculations
- Dynamic pricing based on occupancy and season
- Room upgrade suggestions using conditional logic
- Waiting list management for fully booked dates
- Housekeeping schedule generation using loops
- Guest loyalty program calculations
- Group booking discounts with nested conditions

**Function Design Examples:**

cpp

```cpp
bool checkAvailability(int roomType, int nights) {
    int startFloor, endFloor;

    // Determine floor range based on room type
    switch(roomType) {
        case 1: startFloor = 0; endFloor = 5; break;   // Singles: floors 1-6
        case 2: startFloor = 6; endFloor = 8; break;   // Doubles: floors 7-9
        case 3: startFloor = 9; endFloor = 9; break;   // Suites: floor 10
    }

    // Search using nested loops
    for(int floor = startFloor; floor <= endFloor; floor++) {
        for(int room = 0; room < 20; room++) {
            if(roomStatus[floor][room] == 0) {
                // Check consecutive availability for multiple nights
                // Implementation with additional validation loops
                return true;
            }
        }
    }
    return false;
}
```