Chương 2: Điều khiển Luồng và Hàm

Phần 1: Lý thuyết:
1. Cấu Trúc Rẽ Nhánh
- if-else:
if (điều_kiện) {
  Code thực hiện nếu điều kiện đúng} else if (điều_kiện_khác) {
  Code nếu điều kiện khác đúng} else {
  Code nếu tất cả điều kiện sai}
Ví dụ:
int score = 85;if (score >= 90) {
 cout << "A";} else if (score >= 80) {
 cout << "B"; Output: B} else {
 cout << "C";}
- switch-case:

switch (biến) {
 case giá_trị_1:
   Code thực hiện
  break;
 case giá_trị_2:
   Code thực hiện
  break;
 default:
   Code mặc định}
Ví dụ:
int day = 3;switch (day) {
 case 1: cout << "Monday"; break;
 case 2: cout << "Tuesday"; break;
 case 3: cout << "Wednesday"; break; Output: Wednesday
 default: cout << "Invalid";}

---

2. Vòng Lặp
- for:
for (khởi_tạo; điều_kiện; cập_nhật) {
  Code lặp}
Ví dụ: In số từ 1 đến 5:
for (int i = 1; i <= 5; i++) {
 cout << i << " "; Output: 1 2 3 4 5}
- while:
while (điều_kiện) {
  Code lặp}
Ví dụ: Tính tổng các số từ 1 đến 5:
int sum = 0, i = 1;while (i <= 5) {
 sum += i;

```
  i++;}
cout << sum;   Output: 15
```
- do-while:
```
do {
   Code lặp} while (điều_kiện);
```
Ví dụ: Nhập số đến khi hợp lệ:
```
int num;do {
  cout << "Nhap so duong: ";
  cin >> num;} while (num <= 0);
```

---

3. Hàm (Function)

Khai báo hàm:

Hàm không trả về giá trị (void)
```
void sayHello() {
  cout << "Hello!";}
```
Hàm trả về giá trị
```
int add(int a, int b) {
  return a + b;}
```
Gọi hàm:
```
sayHello();           Output: Hello!int result = add(3, 5);   result = 8
```
Hàm đệ quy:
```
int factorial(int n) {
  if (n == 0) return 1;
  return n * factorial(n - 1);}
```
Giải thích:

factorial(3) = 3 * factorial(2) → factorial(2) = 2 * factorial(1) → ... → Kết quả: 6

Phần 2: Thực hành

Bài 1: Giải Phương Trình Bậc 2 ($ax^2 + bx + c = 0$)

Code mẫu:
```
#include <iostream>#include <cmath>using namespace std;
int main() {
  float a, b, c, delta;
  cout << "Nhap a, b, c: ";
  cin >> a >> b >> c;

  delta = b * b - 4 * a * c;

  if (delta < 0) {
    cout << "Vo nghiem";
  } else if (delta == 0) {
    float x = -b / (2 * a);
    cout << "Nghiem kep: " << x;
  } else {
    float x1 = (-b + sqrt(delta)) / (2 * a);
    float x2 = (-b - sqrt(delta)) / (2 * a);
```

```
    cout << "x1 = " << x1 << ", x2 = " << x2;
  }
  return 0;}
```

Bài 2: Kiểm Tra Số Nguyên Tố
Logic: Số nguyên tố là số chỉ chia hết cho 1 và chính nó.
Code mẫu:

```
bool isPrime(int n) {
  if (n <= 1) return false;
  for (int i = 2; i <= sqrt(n); i++) {
    if (n % i == 0) return false;
  }
  return true;}
int main() {
  int num;
  cout << "Nhap so: ";
  cin >> num;
  if (isPrime(num)) {
    cout << num << " la so nguyen to";
  } else {
    cout << num << " khong la so nguyen to";
  }
  return 0;}
```

Bài 3: Trò Chơi Đoán Số
Mục tiêu: Máy chọn số ngẫu nhiên (1-100), người chơi đoán.
Code mẫu:

```
#include <iostream>#include <cstdlib>#include <ctime>using namespace std;
int main() {
  srand(time(0));
  int secret = rand() % 100 + 1;
  int guess, attempts = 0;

  do {
    cout << "Doan so (1-100): ";
    cin >> guess;
    attempts++;

    if (guess < secret) cout << "So nay nho hon!" << endl;
    else if (guess > secret) cout << "So nay lon hon!" << endl;
    else cout << "Chuc mung! Ban doan dung sau " << attempts << " lan.";
  } while (guess != secret);

  return 0;}
```

Bài 4: In Hình Tam Giác Bằng *
Ví dụ: Tam giác vuông có chiều cao 5:
*
**
***
****
*****
Code mẫu:

```
int main() {
  int height;
  cout << "Nhap chieu cao: ";
  cin >> height;

  for (int i = 1; i <= height; i++) {
    for (int j = 1; j <= i; j++) {
      cout << "*";
    }
    cout << endl;
  }
  return 0;}
```

Lỗi Thường Gặp & Cách Khắc Phục
Quên break trong switch-case: Dẫn đến chạy tiếp các case phía sau.
Vòng lặp vô hạn: Kiểm tra điều kiện dừng và cập nhật biến đếm.
Không khởi tạo biến: Gây kết quả sai (ví dụ: int sum; → sum += i sẽ sai