

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN  
KHOA CÔNG NGHỆ THÔNG TIN**

---o0o---



**BÀI TẬP LAB MÔN CĐ TCDL**

# **BÀI TẬP 1**

Giảng viên hướng dẫn : **VŨ QUỐC HOÀNG**

Sinh viên thực hiện: **CAO QUỐC VIỆT**

MSSV: **22810218**

Lớp : **CN2022/2**

Khoá : **2022/2**

*TP. Hồ Chí Minh, tháng 02 năm 2025*

# Bài làm

## Bài 1.A: Tính tổ hợp bằng giai thừa

Code:

```
#include <iostream>
using namespace std;

// Tính giai thừa
unsigned long long factorial(int n) {
    unsigned long long result = 1;
    for(int i = 2; i <= n; i++) {
        result *= i;
    }
    return result;
}

// Tính tổ hợp theo công thức giai thừa
unsigned long long C(int n, int k) {
    if (k > n) return 0;
    return factorial(n) / (factorial(k) * factorial(n - k));
}
```

Output:

```
vietcq@Legion7:~/CD_TCDL/baiTap/tuan4_bai1$ g++ -std=c++11 bai_1_a.cpp -o bai_1_a; ./bai_1_a
C(5,2) = 10
C(6,3) = 20
C(10,5) = 252
```

## Bài 1.B: Tính tổ hợp không dùng giai thừa

Code:

```
#include <iostream>
using namespace std;

// Tính tổ hợp không dùng giai thừa
unsigned long long C(int n, int k) {
    if (k > n) return 0;
    if (k > n/2) k = n-k; // Tối ưu hóa
    unsigned long long result = 1;
    for(int i = 0; i < k; i++) {
        result = result * (n - i) / (i + 1);
    }
    return result;
}

int main() {
    // Test cases
    C(5, 3);
    C(7, 2);
    C(10, 5);
}
```

Output:

```
vietcq@Legion7:~/CD_TCDL/baiTap/tuan4_bai1$ g++ -std=c++11 bai_1_b.cpp -o bai_1_b; ./bai_1_b
C(5,3) = 10
C(7,2) = 21
C(10,5) = 252
```

## Bài 1.C: Tính tổ hợp bằng tam giác Pascal

Code:

```
#include <iostream>
using namespace std;

// Tính tổ hợp dùng tam giác Pascal
unsigned long long C(int n, int k) {
    if (k > n) return 0;
    unsigned long long C[100][100] = {0};
    for(int i = 0; i <= n; i++) {
        C[i][0] = 1;
        for(int j = 1; j <= i; j++) {
            C[i][j] = C[i-1][j-1] + C[i-1][j];
        }
    }
    return C[n][k];
}
```

Output:

```
vietcq@Legion7:~/CD_TCDL/baiTap/tuan4_bai1$ g++ -std=c++11 bai_1_c.cpp -o bai_1_c; ./bai_1_c
C(4,2) = 6
C(8,3) = 56
C(10,5) = 252
```

### Bài 1.D: So sánh các phương pháp tính tổ hợp

Phương pháp	Công thức tính	Độ phức tạp	Nhận xét
Giai thừa	$C(n, k) = n! / [k! \times (n - k)!]$	$O(n)$	Dễ implement, chạy nhanh với n nhỏ, nhưng dễ tràn số khi n lớn.
Không dùng giai thừa	$C(n, k) = (n/1) \times [(n-1)/2] \times \dots \times [(n-k+1)/k]$	$O(k)$	Chia số để tránh tràn số, chạy nhanh nhất trong ba phương pháp.
Tam giác Pascal	$C(n, k) = C(n-1, k-1) + C(n-1, k)$	$O(n \cdot k)$	Khó tràn số, nhưng chậm hơn phương pháp không dùng giai thừa khi n lớn.

### Bài 3.5.1: Làm tròn số thực

Code:

```
#include <iostream>
#include <cmath>
using namespace std;

// Ham lam tron so thuc 'x' theo 'pos' chu so thap phan
double round_num(double x, int pos) {
    double factor = pow(10.0, pos);
    return round(x * factor) / factor;
}

int main() {
    // Bo du lieu 1: x = 3.14159, pos = 0
    double x1 = 3.14159;
    cout << "Bo du lieu 1 => x = 3.14159, pos = 0 => "
         << round_num(x1, 0) << endl;
```

Output:

```
vietcq@Legion7:~/CD_TCDDL/baiTap/tuan4_bai1$ g++ -std=c++11 bai_3_5_1.cpp -o bai_3_5_1; ./bai_3_5_1
Bo du lieu 1 => x = 3.14159, pos = 0 => 3
Bo du lieu 2 => x = 2.71828, pos = -1 => 0
Bo du lieu 3 => x = 1.99999, pos = 2 => 2
Bo du lieu 4 => x = -3.14159, pos = 0 => -3
Bo du lieu 5 => x = -2.71828, pos = -1 => -0
Bo du lieu 6 => x = 0.12345, pos = -2 => 0
```

#### Bài 4.1.6: Tìm trung vị bằng Quickselect

Code:

```
#include <iostream>
#include <vector>
#include <cstdlib>
#include <ctime>
using namespace std;

// Hamphan vung Quickselect
int partition(vector<double>& arr, int low, int high) {
    double pivot = arr[high];
    int i = low - 1;
    for (int j = low; j < high; ++j) {
        if (arr[j] <= pivot) {
            ++i;
            swap(arr[i], arr[j]);
        }
    }
    ,
```

Output:

```
vietcq@Legion7:~/CD_TCDDL/baiTap/tuan4_bai1$ g++ -std=c++11 bai_4_1_6.cpp -o bai_4_1_6; ./bai_4_1_6
Trung vi: 3.9
Trung vi: 7
Trung vi: 2
```

### Bài 4.2.8: Tìm “điểm yên ngựa”

Code:

```
#include <iostream>
#include <vector>
using namespace std;

// Ham tim diem yen ngua
void tim_yen_ngua(const vector<vector<int>>& a, int m, int n) {
    for(int i = 0; i < m; i++) {
        for(int j = 0; j < n; j++) {
            bool la_min_hang = true;
            bool la_max_cot = true;
            // Kiem tra min hang & max cot
            for(int k = 0; k < n; k++) {
                if(a[i][k] < a[i][j]) {
                    la_min_hang = false;
                    break;
                }
            }
            for(int l = 0; l < m; l++) {
                if(a[l][j] > a[i][j]) {
                    la_max_cot = false;
                    break;
                }
            }
            if(la_min_hang & la_max_cot) {
                cout << "Diem yen ngua tai (" << i << ", " << j << "): " << a[i][j] << endl;
            }
        }
    }
}
```

Output:

```
vietcq@Legion7:~/CD_TCDL/baiTap/tuan4_bai1$ g++ -std=c++11 bai_4_2_8.cpp -o bai_4_2_8; ./bai_4_2_8
Ma tran 4x4:
1 2 3 4
0 5 6 7
5 6 7 8
9 10 11 12
Diem yen ngua tai (3,0): 9
Ma tran 5x5:
1 2 3 4 5
0 9 9 9 9
0 9 9 9 9
0 9 9 9 9
0 9 9 9 9
Diem yen ngua tai (0,0): 1
Ma tran 2x7:
1 2 3 4 5 6 7
0 10 11 12 13 14 15
Diem yen ngua tai (0,0): 1
```

## Bài 5.A: Chuyển chuỗi số thành số nguyên

Code:

```
#include <iostream>
using namespace std;

// Chuyen chuoai so sang so nguyen
unsigned long long to_number(const char* s) {
    unsigned long long num = 0;
    for(int i = 0; s[i] != '\0'; i++) {
        num = num * 10 + (s[i] - '0');
    }
    return num;
}

int main() {
    // Bo du lieu 1
    {
        char s[100];
        strcpy(s, "123456");
```

Output:

```
vietcq@Legion7:~/CD_TCCL/baiTap/tuan4_bai1$ g++ -std=c++11 bai_5_a.cpp -o bai_5_a; ./bai_5_a
So: 123456
So: 99999
So: 2025
```

## Bài 5.B: Chuyển số nguyên thành chuỗi

Code:

```
#include <iostream>
using namespace std;

// Ham chuyen so nguyen 'num' thanh chuoai va luu vao 's'
void to_str(unsigned long long num, char* s) {
    if(num == 0) {
        s[0] = '0';
        s[1] = '\0';
        return;
    }
    int len = 0;
    unsigned long long temp = num;
    while(temp > 0) {
        temp /= 10;        // Chia dần để đếm số chữ số
        len++;
    }
```

Output:

```
vietcq@Legion7:~/CD_TCDL/baiTap/tuan4_bai1$ g++ -std=c++11 bai_5_b.cpp -o bai_5_b; ./bai_5_b
Bo du lieu 1 => num = 123456 => Chuoi: 123456
Bo du lieu 2 => num = 99999 => Chuoi: 99999
Bo du lieu 3 => num = 2025 => Chuoi: 2025
```