

# 舞蹈机器人软件组第一次考核任务——C++相关

## 前言

首先，欢迎大家参与我们舞蹈机器人软件组的考核任务。我们的任务以让大家学习为主要目的，所以不会特别为难大家。第一次的考核主要想让大家先掌握 c++ 这一基础的语言，并加以一定的运用。考核分为**基础部分**和**进阶部分**，基础部分要求全部完成，进阶部分大家**选做**，但是请尽量完成。本次考核持续时间为两周，请在**第二周的周日晚 10 点前**将考核任务按照各自组长的要求进行提交。

期待大家的精彩表现！

P.S. 如果你有途径，你可以让 GPT 帮你写代码，但是前提是你得弄懂你代码的含义哦~

## 基础部分

### 一、C++ 的普通语法

掌握一个语言，语法肯定是最重要的。那么开篇第一个任务我们决定让大家熟悉一下 c++ 的基本语法，那么请实现输出一个 Hello World 吧！

额，开玩笑，这个太简单了，不符合我们工大学子的水平！（狗头保命）

那要不这样，实现一个归并排序吧。请大家自学完成归并排序，写在一个名为 merge\_sort.cpp 的文件里并提交。

### 二、C++ 的高级语法

好的，恭喜你已经掌握了 C++ 的普通语法，现在请你去了解一下简单的数据结构相关知识。请你了解链表、栈、队列、哈希表、堆的相关知识点，并自己尝试实现这些数据结构的功能。

（在你学了之后你就会发现一个叫做 STL 的神奇的东西。这个是 C++ 帮你封装好的库，但是我希望你们先不要用它，而是自己手写实现以理解其思想）

然后，有一个 C++ 和 C 语言的很大的分水岭是——面向对象的思想！

请你自学类的相关知识点，然后将上面我列举的几个简单的数据结构都写成一个类。要求实现：[]、<< 和 >> 符号的重载，插入、删除、更新、查找操作。请你将这些类写成头文件，并编写一个主函数文件对你的这些类加以测试。测试用的主函数大概如下，以 list 为类：

```
// main.cpp
#include <iostream>
#include "data_structures.h"
```

```

int main() {
    your_data_type list;

    std::cout << "Enter elements for the linked list (separated by spaces): ";
    std::cin >> list;
    std::cout << "LinkedList: " << list << std::endl;

    int index;
    std::cout << "Enter an index to access: ";
    std::cin >> index;
    int element = list[index];
    std::cout << "Element at index " << index << ": " << element << std::endl;

    int value;
    std::cout << "Enter a value to search in the linked list: ";
    std::cin >> value;
    bool found = list.find(value);
    if (found) {
        std::cout << "Value " << value << " found in the linked list." << std::endl;
    } else {
        std::cout << "Value " << value << " not found in the linked list." << std::endl;
    }

    int valueToRemove;
    std::cout << "Enter a value to remove from the linked list: ";
    std::cin >> valueToRemove;
    list.remove(valueToRemove);
    std::cout << "LinkedList after removing " << valueToRemove << ": " << list << std::endl;

    int oldValue, newValue;
    std::cout << "Enter an old value to update and a new value: ";
    std::cin >> oldValue >> newValue;
    list.update(oldValue, newValue);
    std::cout << "LinkedList after updating " << oldValue << " to " << newValue << ": " << list << std::endl;
    return 0;
}

```

记得将你的这些文件打包一下哦~

## 提高部分

### 一、C++的专家语法

欢迎来到提高部分（话说这是什么起名方式，真怪）

回归正题。在这一部分，我们想要看到大家的学习成果。首先是归并排序。请大家完成下面这题：

给定一个长度为  $n$  的整数数列，请你计算数列中的逆序对的数量。

逆序对的定义如下：对于数列的第  $i$  个和第  $j$  个元素，如果满足  $i < j$  且  $a[i] > a[j]$ ，则其为一个逆序对；否则不是。

输入格式：第一行输入整数  $n$ ，表示数列的长度；第二行包含  $n$  个整数，表示整个数列；

输出格式：输出一个整数，表示逆序对的个数

数据范围： $1 \leq n \leq 100000$ ，数列中元素取值范围 $[1, 10^9]$ 。

大家将代码写入一个 `cpp` 文件就行了，名字随意。

### 二、C++的写实专家语法——矩阵类

这天，Steven\_Song 正在尝试通关 L4D2 中教区的大桥关，但是突然发现自己的线代还没动过，现在的他慌得一笔。聪明的你，请你帮帮可怜的 Steven\_Song，写一个矩阵类来帮他渡过难关吧！

你的矩阵类需要实现最基本的矩阵运算（输入，输出，加减以及乘法、求逆即可）。同时，请实现矩阵的 $+$ ， $-$ ， $*$ ， $[\ ]$ ， $<<$ ， $>>$ 这些符号的重载。同时写一个方阵类，为矩阵类的子类，并添加求其行列式的方法。

还是一样，请大家将自己的矩阵类写成一个头文件，并编写一个主函数对其进行测试，要求主函数写成测试平台的样式。矩阵中的数据都为 `double` 即可。

```
int main() {  
    Matrix A(2, 2);  
    Matrix B(2, 2);  
  
    std::cout << "Enter values for matrix A (2x2):" << std::endl;  
    std::cin >> A;  
  
    std::cout << "Enter values for matrix B (2x2):" << std::endl;  
    std::cin >> B;  
  
    Matrix C = A + B;  
    Matrix D = A - B;
```

```

Matrix E = A * B;
Matrix F = A.reverse() * A;

std::cout << "A + B = \n" << C;
std::cout << "A - B = \n" << D;
std::cout << "A * B = \n" << E;
std::cout << "A.reverse() * A = \n" << F;

SquareMatrix S(2);

std::cout << "Enter values for a square matrix S (2x2):" << std::endl;
std::cin >> S;

double det = S.determinant();
std::cout << "Determinant of S = " << det << std::endl;

return 0;
}

```

记得将自己的代码打包哦

## 后记

暂时就这些东西了吧。还有很重要的一点，为了让大家养成良好的代码习惯，请大家务必**为自己的代码添加注释!!!**

还有，写代码的时候注意多加空格，增加代码的可读性；代码缩进要规范，并且尽量少用指针(用的时候一定要做判断上保险!)，以及请注意你们的代码中**变量和函数的命名规范!**

要说的暂时就这些吧，提前做完的同学可以提前提交哦~