

Bài thực hành 14

Mục tiêu

- Đọc/ghi trên luồng
- Công cụ đọc/ghi trên luồng
- Quan hệ thừa kế giữa các lớp biểu diễn luồng nhập/xuất
- Truy cập ngẫu nhiên vào file

Bài tập

Câu 1.

Copy hoặc viết lại các chương trình ví dụ trong chương 12 giáo trình rồi chạy thử. Đặt tên các tệp tương ứng với cách đánh số trong giáo trình.

Đánh số ví dụ	Tên tệp	Mô tả
Display 12.1	display121.cpp	Đọc 3 số nguyên từ infile.txt, tính rồi ghi tổng của chúng ra outfile.txt.
Display 12.2	display122.cpp	Giả sử trong file data.txt đã có sẵn một đoạn văn bản, ghi nối vào file này. (Nội dung có sẵn trong data.txt và đoạn cần ghi nối xem trong giáo trình.)
Display 12.3	display123.cpp	Bổ sung vào display121.cpp đoạn mã kiểm tra xem các thao tác mở tệp có thành công hay không.
Display 12.4	display124.cpp	Sao chép nội dung tệp story.txt sang tệp numstory.txt, bổ sung số dòng vào đầu mỗi dòng.
Display 12.7	display127.cpp	Đọc tất cả các số từ tệp rawdata.txt sau đó in ra màn hình và ghi ra tệp neat.txt mỗi số trên 1 dòng, in tới 5 chữ số sau dấu phẩy và căn lề phải.
Display 12.8	display128.cpp	Đọc tệp cad.txt chứa đoạn văn bản mô tả về ngôn ngữ C, thay toàn bộ "C" thành "C++" rồi ghi kết quả vào cppad.txt. Giả sử trong cad.txt tất cả các chữ C viết hoa đều nói về ngôn ngữ C.

Câu 2. [viewcode1.cpp]

Viết chương trình in toàn bộ nội dung tệp chương trình bạn đang soạn thảo ra màn hình.

Câu 3. [viewcode2.cpp]

Viết chương trình ghi toàn bộ nội dung tệp chương trình bạn đang soạn thảo ra tệp văn bản viewcode2.txt nhưng bỏ đi tất cả những chú thích tạo bởi /* ... */

Câu 4. [number.cpp]

Tệp văn bản number.txt chứa các số nguyên hoặc thực trên nhiều dòng (mỗi dòng có thể chứa nhiều số tách nhau bởi ít nhất 1 dấu trắng). Hãy phân tích tệp number.txt và ghi ra tệp văn bản info.txt các thông tin:

- Số lượng số
- Số lượng số nguyên
- Số lượng số thực
- Số lượng số âm

Câu 5. [student_file.cpp]

Hãy sử dụng kiểu dữ liệu Student định nghĩa trong Hình 8.5 (Chương 8 giáo trình tiếng Việt) để viết chương trình có menu tính năng sau:

Chương trình quản lý sinh viên

-
1. Xem danh sách sinh viên
 2. Nhập thêm sinh viên
 3. Tìm sinh viên theo tên
 4. Thoát

- Danh sách sinh viên được lưu trong tệp nhị phân student.dat.
- Tính năng 1 thực hiện đọc tệp student.dat và in toàn bộ danh sách ra màn hình (dạng bảng).
- Tính năng 2 cho phép người dùng nhập tên và điểm sinh viên từ bàn phím, ghi thông tin sinh viên này vào cuối tệp student.dat
- Tính năng 3 cho phép người dùng nhập vào 1 tên và tìm tên đó trong student.dat. In kết quả ra màn hình, đồng thời ghi ra file văn bản search.txt.

student.h

```
#ifndef STUDENT_H
#define STUDENT_H
#include <iostream>
#include <iomanip>
#include <cstring>
using namespace std;

const int MAX_NAME_LENGTH = 20;

struct Student{
    char name[MAX_NAME_LENGTH + 1];
    float score;
    Student();
    Student(const char * aname, float ascore);
    void println();
};

Student::Student(){
    strcpy(name, "no name");
    score = 0;
}

Student::Student(const char * aname, float ascore){
    if(strlen(aaname) > MAX_NAME_LENGTH){
        strncpy(name, aaname, MAX_NAME_LENGTH);
        name[MAX_NAME_LENGTH] = '\0';
    }
    else strcpy(name, aaname);
    score = ascore;
}

void Student::println(){
    cout << setw(MAX_NAME_LENGTH) << name << setw(7) << score << endl;
}
#endif
```