

KIỂM TRA CUỐI KỲ

Thời gian làm bài: 75 phút

Ngày thi: 22/12/2022

Quy định

1 Quy định làm bài

- KHÔNG sử dụng tài liệu.
- Sinh viên làm bài thi trên máy tính bàn ở phòng máy. KHÔNG dùng máy tính cá nhân.
- Bài thi phải được sinh viên thực hiện tại phòng máy.

2 Quy định nộp bài

- Sinh viên thực hiện mỗi bài làm trong một tập tin `bai0x.cpp`. Ví dụ làm bài 1, tên tập tin là `bai01.cpp`.
- Toàn bộ bài làm sẽ được đặt vào thư mục MSSV. Sau đó thực hiện nén thư mục MSSV thành tập tin `MSSV.zip`.
- Sinh viên nộp tập tin `MSSV.zip` lên Moodle trước deadline. KHÔNG NHẬN BÀI NỘP TRỄ.

3 Quy định chấm bài

- KHÔNG chấm ý tưởng, chỉ chấm ở kết quả chạy của chương trình.
- Những trường hợp sau đây sẽ bị trừ nhiều điểm:
 - Nộp sai quy định.
 - Chương trình có lỗi cú pháp.
- Những sinh viên có BÀI LÀM GIỐNG NHAU sẽ bị 0 điểm.

Nội dung

Bài 1 (4.5 điểm)

Một laptop có các thông tin liên quan bao gồm: thương hiệu (tối đa 10 ký tự), dòng máy (20 ký tự), kích thước màn hình và giá tiền. Ví dụ: laptop có thương hiệu Thinkpad, dòng máy T15, kích thước màn hình 15.6 inches và giá tiền là 25000000 đồng.

Hãy thực hiện các yêu cầu sau:

1. (0.5 điểm) Hãy thiết kế struct **Laptop** để lưu trữ các thông tin trên.
2. (1 điểm) Viết hàm cho phép nhập vào danh sách các laptop từ bàn phím.
3. (0.5 điểm) Viết hàm in ra danh sách các laptop, mỗi laptop nằm trên một dòng.
4. (1.5 điểm) Viết hàm điều chỉnh giá tiền của tất cả các laptop theo quy tắc làm tròn lên đến hàng nghìn (ngàn). Ví dụ giá tiền là 25214135 đồng \rightarrow 25215000 đồng.
5. (1 điểm) Viết chương trình cho phép người dùng nhập vào danh sách các laptop, điều chỉnh thông tin giá tiền theo quy tắc làm tròn đến hàng nghìn (ngàn), in ra danh sách sau khi điều chỉnh. Chương trình sử dụng các hàm đã cài đặt trên.

Bài 2 (3.5 điểm)

Dịp tổng kết năm học và cũng gần Giáng sinh, cô giáo chủ nhiệm của lớp 22CLCX muốn tặng một phần quà cho một sinh viên may mắn. Cô giáo cho tất cả sinh viên ghi mã số sinh viên (MSSV) vào các mảnh giấy sau đó nộp lại. Vì cô giáo không thông báo trước cách để chọn ra sinh viên may mắn nên có một vài bạn đã gian lận và ghi MSSV của mình **hiều hơn một lần** để tăng khả năng được phần quà trong trường hợp cô giáo bốc thăm ngẫu nhiên.

Tuy nhiên cô giáo đã phát hiện nên sau khi thu toàn bộ mảnh giấy, cô giáo bắt đầu công bố những sinh viên gian lận (nộp nhiều hơn một lần) và loại những sinh viên ra khỏi danh sách. Tiếp theo, cô giáo công bố sinh viên may mắn nhận quà là sinh viên **có tổng các chữ số trong MSSV chia hết cho tổng số lượng sinh viên còn lại trong danh sách**. Nếu có nhiều sinh viên thỏa yêu cầu trên, chọn sinh viên có MSSV nhỏ nhất thỏa yêu cầu. Nếu không có sinh viên nào thỏa yêu cầu trên, chọn sinh viên có tổng chữ số lớn nhất (bất kỳ).

Viết chương trình cho phép nhập vào danh sách MSSV lúc cô giáo thu toàn bộ mảnh giấy. Sau đó, chương trình thông báo các sinh viên gian lận ra màn hình. Cuối cùng, thông báo sinh viên may mắn được tặng quà.

Ví dụ:

- Cô giáo nhập vào danh sách MSSV thu được lần lượt là: [101, 196, 293, 101, 653, 123, 516, 293].
- Sinh viên gian lận bị loại khỏi danh sách là: 101 và 293.
- Sinh viên may mắn được tặng quà là: 196.
- Giải thích: Sau khi loại sinh viên gian lận, danh sách còn lại 4 sinh viên là: [196, 653, 123, 516]. Có 2 sinh viên đều thỏa mãn với tổng các chữ số chia hết cho 4 là 196 (tổng 16) và 516 (tổng 12). Vì luật ưu tiên sinh viên có MSSV nhỏ hơn \rightarrow chọn 196.

Bài 3 (2 điểm)

Cho một câu tiếng Anh với mỗi từ cách nhau bởi một khoảng trắng, có dấu câu là dấu phẩy hoặc dấu chấm, các từ trong câu không viết hoa.

Hãy thực hiện các yêu cầu sau:

1. *(0.5 điểm)* Viết hàm đếm số lượng từ có trong câu.
Prototype: `int countTotalWords(char sent[]);`
2. *(1 điểm)* Viết hàm tìm từ xuất hiện nhiều nhất trong câu và số lần xuất hiện của từ đó.
Prototype: `int findMostWord(char sent[], char mostWord[]);`
3. *(0.5 điểm)* Viết chương trình cho phép nhập vào một câu, chương trình in ra màn hình số lượng từ có trong câu và từ xuất hiện nhiều nhất (kèm số lần xuất hiện). Chương trình sử dụng các hàm đã viết trên.

Ví dụ: Người dùng nhập vào câu "to be, or not to be." Chương trình sẽ in ra màn hình số lượng từ có trong câu là: 6, và từ xuất hiện nhiều nhất là: to (xuất hiện 2 lần) hoặc be (xuất hiện 2 lần).

Hết