**BÁO CÁO BÀI TẬP 3 TUẦN 2**

**MẠNG MÁY TÍNH – TRUYỀN THÔNG**

**Đề tài:** Trò chơi thi đố trắc nghiệm.

**Code by:**

* Nguyễn Văn Hoàng
* MSSV: 58131303
* Lớp: 58-TH1
* TTCS - Nhóm 1



Nha Trang, ngày 08 tháng 01 năm 2019

**NỘI DUNG BÁO CÁO:**

1. Trò chơi thi đố trắc nghiệm là gì ?
2. Lợi ích và hạn chế của thi đố trắc nghiệm.
3. Mục tiêu của bài toán đối với coder.
4. Xây dựng thuật toán và các hàm .
5. Trò chơi thi đố trắc nghiệm là gì ?

**Lý thuyết Trắc nghiệm cổ điển** (Classical Test Theory) là một lý thuyết liên quan với nhánh khoa học dđo lường trong giáo dục (educational measurement) và tâm trắc học (psychometrics), phục vụ cho việc thiết kế các công cụ đo lường để xác định giá trị năng lực hoặc trình độ của đối tượng (thí sinh) được đo. Lý thuyết này bắt đầu phát triển khoảng đầu thế kỷ 20 và được hệ thống hóa vào thập niên 1970. Đối sánh với Lý thuyết Trắc nghiệm cổ điển là lý thuyết ứng đáp câu hỏi (Item Response Theory - IRT), được bắt đầu xây dựng từ khoảng đầu nửa sau của thế kỷ 20, dựa trên các mô hình toán học.

1. Lợi ích và hạn chế của thi đố trắc nghiệm.
2. *Lợi ích:*

* Khảo sát được số lượng lớn thí sinh.
* Kết quả nhanh.
* Điểm số đáng tin cậy.
* Công bằng, chính xác, vô tư.
* Ngăn ngừa "học tủ".

1. *Hạn chế:*

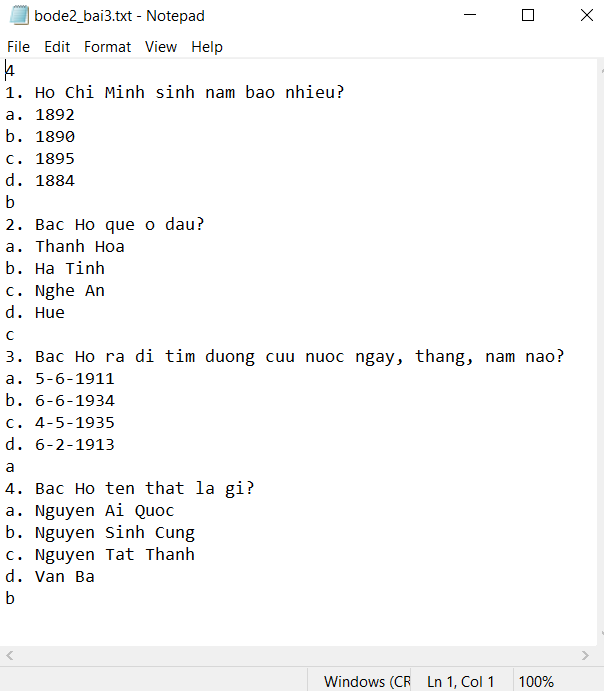
* Thí sinh có khuynh hướng đoán mò đáp án. (Độ may rủi: là xác suất thí sinh đoán mò và làm đúng)
* Không thấy rõ diễn biến tư duy của thí sinh
* Khó soạn đề và tốn công sức
* Theo quan điểm của nhiều người, việc áp dụng thi trắc nghiệm trong kỳ thi tuyển sinh Đại học ở Việt Nam là không thích hợp trong tình hình hiện nay. Về sâu xa, thi trắc nghiệm thực sự có hiệu quả khi cần sàng lọc ở cấp thấp về tri thức.

1. Mục tiêu của bài toán đối với coder.

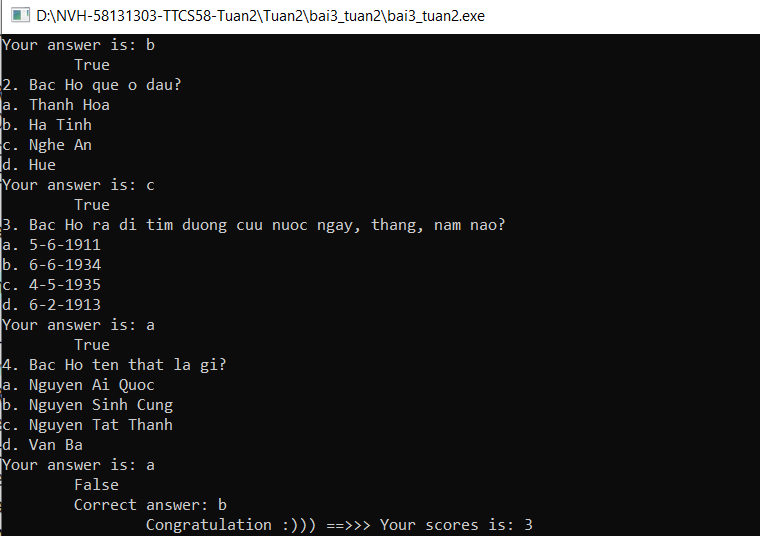
* Giúp người lập trình có kỹ năng thao tác:
* Trên file.
* Trên kiểu dữ liệu cấu trúc.
* Xử lý các vòng lặp.
* Câu lệnh rẽ nhánh.

1. Xây dựng thuật toán và các hàm .

* Chương trình sẽ được xây dựng bởi 5 hàm chính:
* Đầu tiên là hàm **ReadQuestionFromFile**: ( đọc file )
* Hàm này sẽ đọc dữ liệu từ file ( file nào thì tùy vào người chơi chọn bộ đề nào ).
* Đầu tiên hàm sẽ đọc số lượng câu hỏi có trong bộ đề thi.
* Tiếp đến là 4 đáp án mà người chơi sẽ phải lựa chọn 1 trong 4 đáp án đó.
* Cuối cùng là đáp án chính xác. Đáp án chính xác này sẽ không được hiển thị ra màn hình cmd mà chỉ để dùng so sánh với đáp án mà người chơi sẽ chọn.
* VD 1 file:



* Tiếp đến là hàm **OutputQuestion**: ( hiển thị câu hỏi )
* Ở hàm này sẽ có biến scores để tính điểm cho người chơi, mỗi lần người chơi trả lời đúng câu hỏi thì nó sẽ in ra màn hình “True” và biến scores này sẽ tăng thêm 1.
* Còn nói câu trả lời của người chơi sai thì nó sẽ in ra màn hình “False” và “Correct Answer” và đáp án đúng của câu hỏi đó.
* Và cuối cùng sẽ xuất ra số điểm của người chơi ( tức là biến scores ).



* Tiếp theo là hàm **check**: ( kiểm tra câu trả lời )
* Hàm này có kiểu giá trị là **bool** tức là trả về giá trị đúng sai. Nó sẽ so sánh câu trả lời của người chơi nhập từ bàn phím vào có giống với đáp án đúng của câu hỏi đó hay không.
* Nếu đúng thì nó sẽ trả về **True** và sai sẽ trả về **False**.
* Kế cuối cùng là hàm **chon**: ( để người chơi lựa chọn bộ đề mà mình muốn làm )
* Ở hàm này người chọn sẽ lựa chọn bộ đề mà mình muốn làm. Hàm **menu** nằm trong hàm **chon** sẽ hiển thị số thứ tự gắn liền với từng chủ đề của bộ đề đó.
* VD: bộ đề về những thông tin cần biết về Bác Hồ, bộ đề có chủ đề về các loài vật, …
* Và cuối cùng là hàm quan trọng không thể thiếu đó là **int main.**

Hàm này chỉ gọi lại các hàm chính trong chương trình mà thôi.

Vì ở hàm **chon** đã làm hết các công việc lựa chọn cho người chơi rồi ở hàm **int main** chỉ gọi lại hàm **chon** mà thôi.