**Assignment 1 FUNDAMENTALS**

Viết một chương trình cho phép user nhập vào tới 20 số nguyên (integers). In ra kết quả gồm số trung bình của các số nhập vào và số lượng số có giá trị nhỏ hơn giá trị trung bình này.

Viết 1 hàm chuyển các giá trị chẵn về bên phải, lẻ về bên trái của mảng đó

Yêu cầu: dùng mảng các số nguyên để xử lý kết quả.

**Assignment 2 Pointers Basic**

Viết 1 chương trình bao gồm tính tổng/tích của 2 ma trận

Yêu cầu:

-          Có hàm nhập ma trận từ bàn phím (size & content), dùng hàm cấp phát động

-          Có hàm in ma trận lên màn hình

-          Có hàm tính tổng 2 ma trận

-          Có hàm tính tích 2 ma trận

**Assignment 3 Functions & Arrays**

**Bài 1**: Viết 1 hàm tính diện tích hình chữ nhật cho biết chu vi & đường chéo.

|  |
| --- |
| **Input screen:**  10 4    **Output screen:**  S=xxx  or The value inputs are not valid, please re-enter new inputs. |

**Bài 2**: Viết chương trình bao gồm tính tổng/ tích của 2 ma trận.

Yêu cầu:

·         Có hàm nhập ma trận từ bàn phím

·         Có hàm in ma trận lên màn hình

·         Có hàm kiểm tra xem có nhân được/ cộng được 2 ma trận đã nhập.

·         Có hàm tính tổng 2 ma trận

·         Có hàm tính tích 2 ma trận

* Có function kiểm tra kích thước mảng nhập đầu vào nhé.

Nếu đúng, thì continue, nếu sai thì break ra nhé.

Ko dùng con trỏ

 Anh bổ sung thêm yêu cầu đề bài:

- Có function kiểm tra kích thước mảng nhập đầu vào nhé. Nếu đúng, thì continue, nếu sai thì break ra nhé.

-

Yêu cầu về output screen:

|  |
| --- |
| **Input screen:**  Matrix A Row: 2  Matrix A Col: 2  Matrix B Row: 2  Matrix B Col: 2    Matrix A:  1 2 3 4    Matrix B:  5 6 7 8    **Output screen:**  Matrix A, Matrix B can/ cant add together  Matrix A, Matrix B can/ cant multi together  Matrix B, Matrix A can/ cant multi together    Matrix A:  1 2  3 4    Matrix B:  5 6  7 8    Matrix A + Matrix B: (*if cannot, print N/A at here*)  6 8  10 12    Matrix A \* Matrix B: (*if cannot, print N/A at here*)  6 8  10 12    Matrix B \* Matrix A: (*if cannot, print N/A at here*)  6 8  10 12 |

**Assignment 4 – data structure and algorithms**

*Bài 1:Viết 1 chương trình quản lý thông tin lớp fresher bao gồm:*

* *Mã số nhân viên*
* *Tên*
* *Account*
* *Điểm trung bình*

*Yêu cầu:*

* *Sử dụng link list đơn để quản lý danh sách*
* *Có các hàm để thêm/bớt 1 nhân viên*
* *Có hàm để sắp xếp nhân viên theo* 
  + *Điểm trung bình*
  + *Tên*

·         **Tuyệt đối không sử dụng các hàm/phương thức (ví dụ scanf, getc, gets, ….) lấy input từ người chạy chương trình khi làm bài**.

**Assignment 5- File handling**

Write a program to parse s-record file. Print out the info:

* Converted address
* Converted data

**Requirement: (Violate any req, will be marked as 0)**

Not allow to use functions to allocate dynamic memory such as malloc, calloc.

Separate the program to layers: parse, app.

Not allow sharing any global variable between layers.

The prototype of parse function is required as below:

**typedef enum {**

    e\_parseStatus\_done = 0x00U,

e\_parseStatus\_start = 0x01U,

    e\_parseStatus\_inprogress = 0x02U,

    e\_parseStatus\_error = 0xFFU,

} parse\_status\_t;

typedef struct

{

    uint32\_t address;

    uint8\_t data[MAX\_LINE\_DATA\_SIZE];

    uint8\_t dataLength;

} parse\_data\_struct\_t;

parse\_status\_t parseData(uint8\_t pInput[], parse\_data\_struct\_t \*pOutput);

**Param:**

* pInput[] – srec line
* pOutput – output of parsed data

**Return:**

Start – When parsing S0 line

Inprogress – when parsing S1 or 2,3,4,5 or 6 line

Done – when parsing S7 or 8, 9 line

Error – When error

Assignment 6 Pointer Advanced

Bài 1: Viết 1 chương quản lý thông tin của các nhân viên mới vào làm việc cho Fsoft, bao gồm:

- ID (4 bytes)

- Class (1 byte)

- GPA (1byte)

Biết rằng các nhân viên mới này vừa hoàn thành khóa học từ 1 trong 2 lớp fresher A, fresher B.

Yêu cầu:

- Sử dụng link list đơn để quản lý danh sách các nhân viên

- Có các hàm để thêm/bớt 1 nhân viên (2 nhân viên không được trùng ID), sắp xếp lại linked list theo điểm trung bình (GPA) từ cao đến thấp ngay khi thêm/bớt nhân viên.

- Trường hợp 2 nhân viên có điểm trung bình bằng nhau, thì nhân viên nào thuộc lớp A sẽ được tính là cao điểm hơn.

- Trường hợp 2 nhân viên có điểm trung bình bằng nhau và cùng chung lớp A hoặc B thì nhân viên nào được thêm vào trước sẽ được tính là có điểm cao hơn.//them luon

- Biết số lượng nhân viên tối đa là 20 bạn. Không được sử dụng các hàm cấp phát bộ nhớ động mà sẽ sử dụng 1 mảng 20 phần tử ( với mỗi phần tử được khởi tạo: ID = 0xFFFFFFFF, Class = 0xFF, Điểm trung bình GPA = 0xFF) để lưu thông tin cho mỗi nhân viên mới. Ngoài các trường ID, Class, GPA thì có thể thêm các trường khác để quản lý nếu cần.

- Việc thêm/bớt nhân viên có thể diễn ra đồng thời (ví du: thêm nhân viên A, thêm nhân viên B, bớt nhân viên A, thêm nhân viên C, thêm nhân viên D, thêm nhân viên E, bớt nhân viên D, thêm nhân viên F …etc -> việc thêm/bớt do user quyết định).

Assignment 7 Using macros and bit operation & Optimization in C

- Viết hàm swapBit và swapByte cho 1 số 32bit

- Viết hàm display các số trước và sau khi gọi hàm swapBit, swapByte