

Bài tập thực hành 4.

I) Mục tiêu:

- Tổng quát lại các kiến thức đã học về tràn số nguyên, xử lý dữ liệu đầu vào, lỗi tràn bộ đệm;
- Rèn luyện kỹ năng lập trình.

II) Nội dung thực hành:

Program 1. Viết chương trình thực hiện các công việc sau:

1. Khởi tạo hai mảng A và B chứa các số nguyên gồm n phần tử. Trong đó n được nhập vào từ bàn phím ($10 < n < 20$). Nếu nhập sai yêu cầu nhập lại. Giá trị của các số nguyên sẽ được khởi tạo ngẫu nhiên trong khoảng $[10, 500]$;

2. In ra màn hình 2 mảng A và B vừa khởi tạo;

3. Viết hàm lưu lại các phần tử chẵn trong A và lẻ trong B lưu vào mảng C.

In ra mảng C.

4. In ra màn hình tích các phần tử trong C. Nếu giá trị vượt quá INT_MAX thì đưa ra thông báo “*Integer Overflow*”.

Ví dụ: C[5]: 8 9 12 56 65

Result: $M = 8 * 9 * 12 * 56 * 65 = 3144960$

Program 2. Viết chương trình nhập vào chuỗi S1 và S2. Trong đó S2 chỉ bao gồm các số. Viết các hàm con thực hiện các công việc:

1. Kiểm tra xem S2 có bao gồm toàn số hay không. Nếu không in ra dòng chữ “*Input Validation*” và yêu cầu nhập lại.

2. Kiểm tra xem trong chuỗi S1 có chữ số hay không.

3. Kiểm tra chuỗi S2 có phải là số điện thoại của mobifone không. Giả sử số điện thoại mobifone có dạng 090xxxxxxx.

4. Kiểm tra S2 có là tập con của S1 hay không.

5. Viết hàm thực hiện việc nối chuỗi S2 vào S1, ta thu được chuỗi mới là S3.
In ra màn hình chuỗi S3.

6. Chèn chuỗi ký tự **Buffer_Overflow** vào sau ký tự **c** bất kỳ nhập vào bàn phím. Nếu trong chuỗi S3 không có ký tự **c** thì thêm **Buffer_Overflow** vào cuối chuỗi S3.

Ví dụ:

+S3: Secure coding6789

+ Insert **Buffer_Overflow** after: e

+ S4: **Buffer_Overflow** cure **Buffer_Overflow** coding6789