

## Bài tập ngày 5 (pointer)

- Viết chương trình chuyển đổi 1 số nguyên được biểu diễn ở dạng big endian sang dạng little endian.

**Đề sử dụng cho câu 2, 3:**

Cho 1 mảng các số nguyên:

```
int arr[] = {1, 2, 3, 4, 5};
```

- Viết chương trình sử dụng con trỏ kiểu char để tăng mỗi phần tử của mảng lên 1 đơn vị, sau đó xuất các phần tử của mảng ra màn hình. Output của màn hình sẽ là:

```
2 3 4 5 6
```

- Viết chương trình in ra vùng nhớ biểu diễn mảng trên khi:
  - Bộ xử lý dùng định dạng little endian.
  - Bộ xử lý dùng định dạng big endian.

Kết quả xuất ra màn hình sẽ là:

```
Big Endian: 00 00 00 01 00 00 00 02 00 00 00 03 00 00 00 04 00 00 00 05
```

```
Little Endian: 01 00 00 00 02 00 00 00 03 00 00 00 04 00 00 00 05 00 00 00
```

- Cho đoạn chương trình sau:

```
typedef struct _EntryComponent
{
    int      posX, posY, posZ;
    short    rotX, rotY, rotZ;
    bool      canMove;
} EntryComponent;

int main()
{
    EntryComponent entry = { ?, ?, ?, ?, ?, ?, ? };

    FILE* f = fopen("entry1.bin", "wb");
    fwrite(&entry, sizeof(entry), 1, f);
    fclose(f);

    return 0;
}
```

Biết file entry1.bin (được đính kèm trong đề bài) là file lưu trạng thái của 1 đối tượng EntryComponent. File được lưu trên máy có bộ xử lý dùng định dạng little endian và struct EntryComponent được align 1 byte. Load đối tượng EntryComponent đã được lưu và in thông tin đối tượng đã ra màn hình theo định dạng:

## Problem Solving with C++

---

Entry:

Position: x y z

Rotation: rx ry rz

canMove: True