

Số học số nguyên lớn

Trong chuyên đề này chúng ta xem xét việc biểu diễn số nguyên lớn ($\approx 10^{100000}$) bằng C++.

1. Biểu diễn số nguyên lớn

Chúng ta chọn COSO=100000000 (10^8) làm cơ số để biểu diễn. Như vậy mỗi chữ số sẽ tương ứng với 8 chữ số trong cơ số 10:

```
#define maxD 10000 // Số lượng chữ số tối đa trong cơ số  $10^8$ 
#define COSO 100000000 // Định nghĩa hằng

struct big{int num;int d[maxD];}; // num - bậc cao nhất, d - mảng chữ số
```

Như vậy các chữ số bậc 0, bậc 1, ... sẽ là d[0], d[1],....

2. Nhập / xuất và so sánh số nguyên lớn

a) Đọc số nguyên lớn từ input

Nhập số nguyên lớn như chuỗi ký tự, sau đó đảo ngược các chữ số của chuỗi để hàng đơn vị đứng đầu,.... Tiếp theo nhóm 8 chữ số của cơ số 10 để tạo ra 1 chữ số của cơ số mới:

```
char s[10*maxD];
void read_big(big &x) {
    scanf("%s",s);
    int n=strlen(s);
    reverse(s,s+n);
    x.num=-1;
    int u=0;
    while (u<n) {
        x.d[++x.num]=0;
        int lt=1;
        for(int i=0;i<8;i++) {
            if (u+i>=n) break;
            x.d[x.num]+=(s[u+i]-48)*lt;
            lt*=10;
        }
        u+=8;
    }
}
```

b) Ghi số nguyên ra output:

Chú ý, ngoại trừ chữ số bậc cao nhất, các chữ số còn lại phải thêm ký tự '0' ở phía trước để cho đủ 8 ký tự (format in "%08d"):

```
void write_big(big x) {
    printf("%d",x.d[x.num]);
    for(int i=x.num-1;i>=0;i--) printf("%08d",x.d[i]);
}
```

c) Gán một số nguyên kiểu int, long long thành số nguyên lớn

```
void Gan(int k,big &x) {
    x.num=-1;
    do {
        x.d[++x.num]=k % COSO;
        k/=COSO;
    } while (k);
}
```

d) So sánh hai số nguyên lớn

Ta viết hàm cmp(x,y) trả về -1,0,1 tùy thuộc vào $x<y$, $x=y$, $x>y$:

```
int cmp(big x,big y) {
    if (x.num>y.num) return 1;
    if (x.num<y.num) return -1;
    for(int i=x.num;i>=0;i--) {
```


