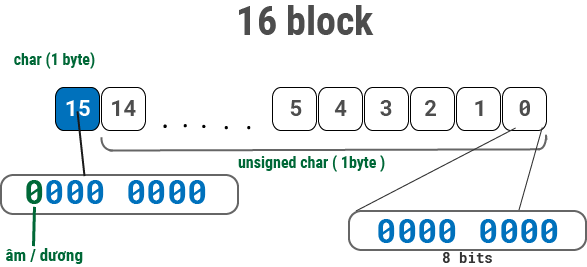
II. THUẬT TOÁN VÀ KIỂM TRA TRƯỜNG HỢP

1. Mô tả cách thức thuật toán :.
2. Nguyên lý lưu thông tin số nguyên lớn của đồ án :

- Chia 128 bits dữ liệu thành 1 mảng 16 block độ lớn 1 byte (unsigned char)

- Riêng block cao nhất được định nghĩa là [char] để có thể dễ nhận dạng số âm.

- Bit cao nhất (tức là bit cao nhất của block thứ 15) dùng để đánh dấu số âm.

- Lưu số nguyên theo định nghĩa số bias bù 2.  


1. Chuyển chuỗi thành số nguyên lớn (16 bytes) :  
   (\*) Chuỗi ký tự bit  
   - Ví dụ: “10101011”, “00111001”, ….  
   - Mỗi kí tự có 2 giá trị là ‘0’ và ‘1’

* Ý tưởng :
  + Kí tự thứ n trong chuỗi sẽ là giá trị của bit thứ n trong số nguyên
  + Do ta chia số nguyên thành 16 block nên ta cần phải tính toán vị trí bit trong số ứng
  + Sử dụng các phép logic để gán bit đó vào vị trí cụ thể của block.

Bit = ChuoiBin[len - Position - 1];

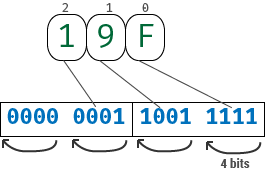
Vị\_Tri\_Block = Position / 8;

Vi\_Tri\_Trong\_Block = Position % 8;

//Su dung phep toan ^ de gan bit vao 1 so

Block[Vi\_Tri\_Block] = Block[Vi\_Tri\_Block] ^ (Bit Vi\_Tri\_Trong\_Block);

1. Chuỗi kí tự hex :

* Ví dụ : ”00FFAA”, “13B”, …
* Mỗi kí tự là một số tương ứng trong hệ số thập lục phân (1,2,3,4,5,6,7,8,9,A,B,C,D,E,F)  
  
* Ý tưởng giống với chuyển chuỗi kí tự bit nhưng :
  + Giá trị ở hệ số n trong chuỗi sẽ tương ứng với 4 bits thứ n trong số nguyên 16 bytes.
  + Sử dụng các phép logic để gán 4 bit vào vị trí cụm 4 bit thứ n.

Hex = ChuoiHex[Position];

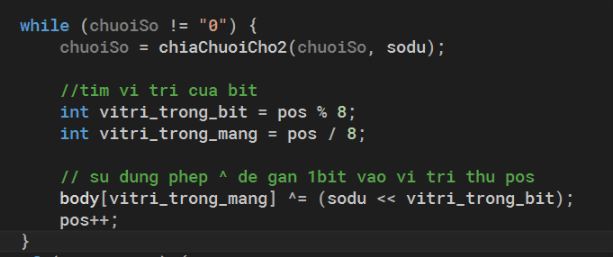
Vi\_Tri\_Block = Position / 2;

Vi\_Tri\_Trong\_Block = Position % 2;

Block[Vi\_Tri\_Block] = Block[Vi\_Tri\_Block] ^ (Hex << 4\*(Vi\_Tri\_Trong\_Block)));

1. Chuỗi thập phân :

* Ví dụ :”12345”,”93”, …
* Mỗi kí tự là số 0 – 9.
* Do sơ cấp 10 không là bội ước của 255, nên việc chia thành từng dãy bits không dễ dàng. Cách duy nhất chúng ta có thể làm được là “chia 2” chuỗi số dần để lấy số dư – là bit thứ 2n của chuỗi số đó.
* Ý tưởng :
  + Thiết lập thuật toán “chia 2” chuỗi số và xuất ra số dư
  + Chia dần chuỗi số, tìm số dư
  + Gán bit (là số dư tìm được) vào vị trí tăng dần khi chia.



Trong hình: **body** là *dãy block*, **pos** là vị trí đếm từ 0 lên.

1. Hàm chuyễn đổi thập phân xang nhị phân :  
   - Đem chuỗi ***chia*** 2 lấy dư

* Số dư đưa vào kết quả
* Qúa trình lập lại cho đến khi số nguyên bằng 0.

1. Hàm chuyễn đổi nhị phân xang thập phân :

* Chuỗi kết quả ban đầu là 0
* Số x là vị trí bit cao nhất (khác bit dấu)
* Đem chuỗi nhân 2 công giá trị bit thứ x
* Giảm x xuống 1 đơn vị
* Lập lại cho đến khi x bằng 0.

1. Hàm chuyễn đổi nhị phân xang thập lục phân :

* Chọn 4 bit từ 8 bit của 1 ô rồi so sánh với bảng đã biết

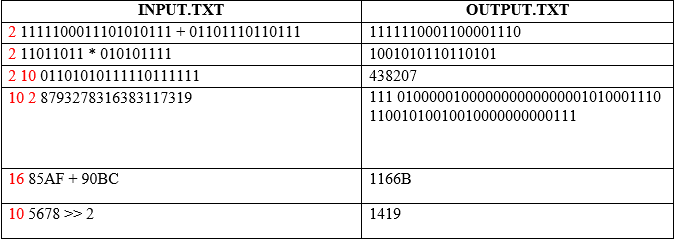
1. Toán tử operator “ + ”

* Đem block thứ i của số A cộng cho block thứ i của số B
* Phần tràn đem cộng vào block i + 1 của kết quả
* Cộng hết các block tương ứng với nhau
* Do ta sử dụng nguyên tắc số bias thì phần tràn lớn nhất sẽ bị bỏ đi.

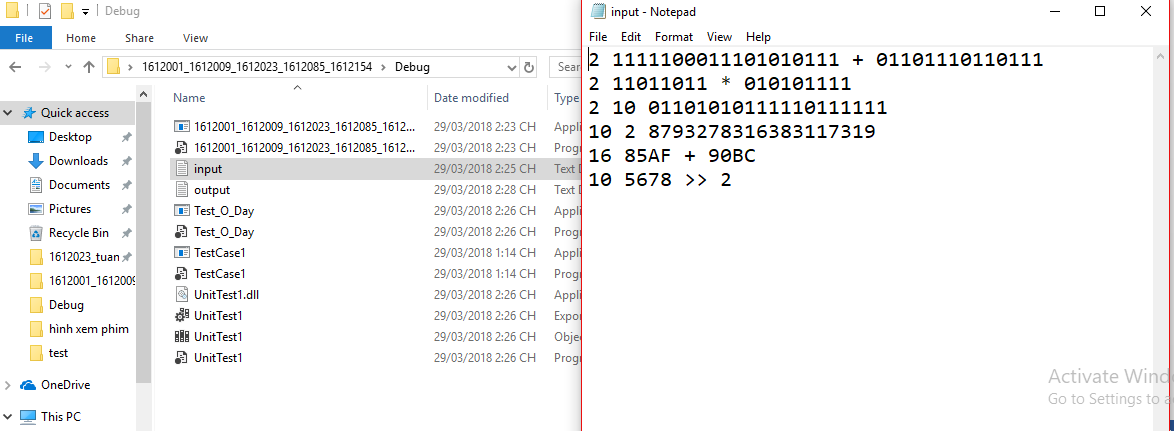
1. Toán tử operator” - ”

* Sử dụng nguyên tắt A – B = A + ( - B )
* Sử dụng lại toán tử cộng.

1. Toán tử operator” \* ”
2. Toán tử operator” / ”
3. Kiểm tra trường hợp đề bai :
   1. Test theo ví dụ đề bài :



* File test Intput :



* File test Output :

