**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**KHOA CÔNG NGHỆ THÔNG TIN**

****

**Architecture Computer - Assembly**

By

Vũ Hoàng Dương - 1612127

Lê Quốc Dũng – 1612126

Nguyễn Hữu Điền – 1612104

Nguyễn Hoàng Giang – 1612151

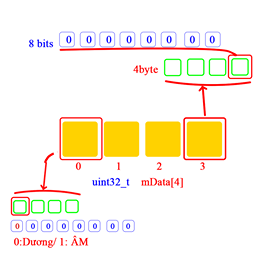
**Teacher’s Name**

Lê Viết Long

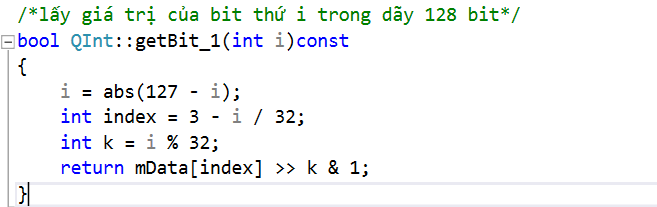
**Class**: KTMTHN-2016/4

# Ý tưởng , THUẬT TOÁN

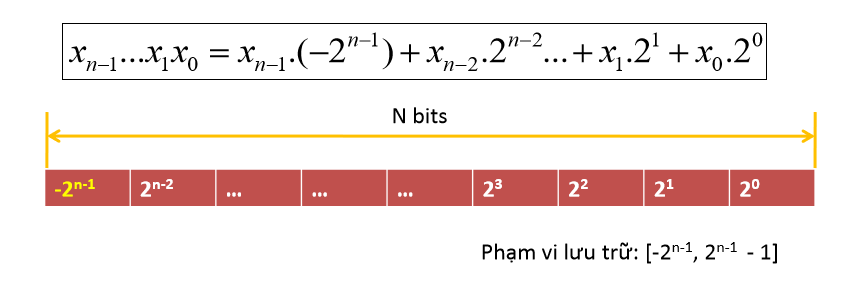
* **Nguyên lí lưu thông tin của số nguyên lớn:**

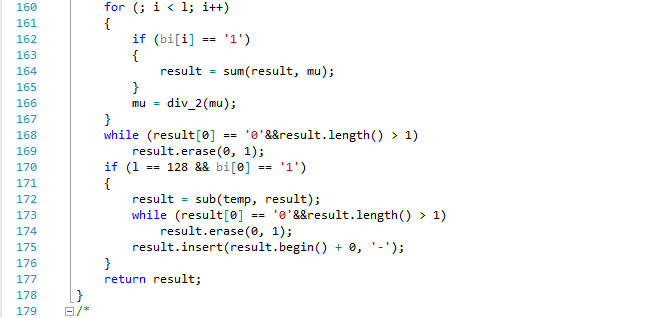
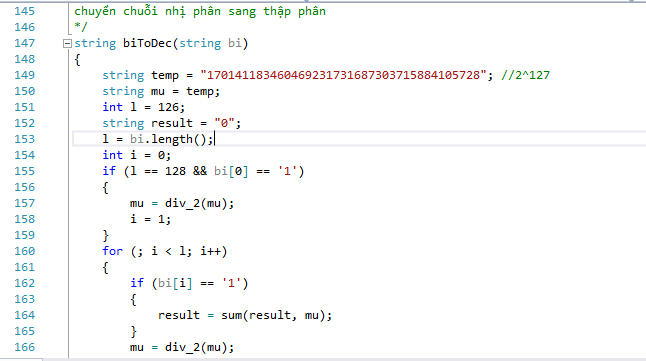
Số nguyên giới hạn biểu diễn trong 128 bit, được lưu trong mảng mData[4] gồm 16 byte.  
Chuyến số nguyên lớn ở các hệ 10,16 về hệ 2. Lưu các bit theo thứ tự từ phải sang trái bằng thuật toán bật bit 1 (1 SHL i) OR x.  
Các bit được lưu trong các phần tử mData từ trái sang phải.  
Với các số nguyên có dấu, chuyển về lưu dưới dạng bù 2. Nguyên tắc: lấy nghịch đảo và cộng thêm 1.  


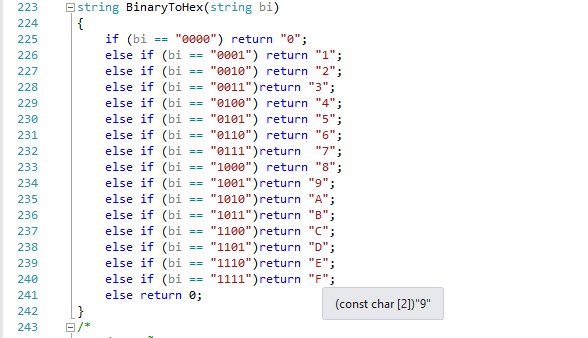
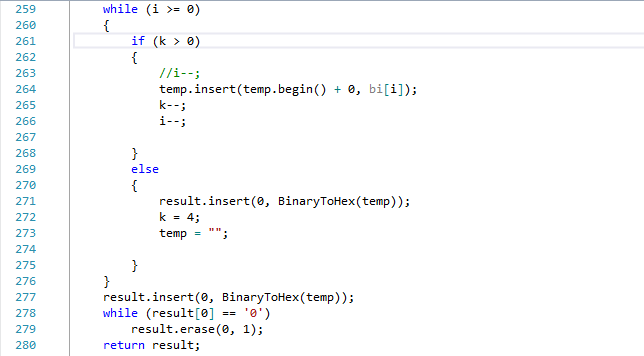
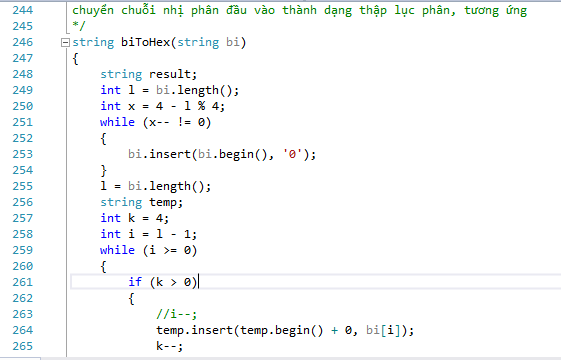
* **Lấy 1 bit từ 16 byte lưu trữ  
  Nguyên tắc:** lợi dụng tính chất một bit với bit 1 đều ra chính nó, ta sẽ sử dụng phép dịch bit đưa bit cần lấy về cuối rồi thực hiện phép toán AND với 1.  
  **Cài đặt:**

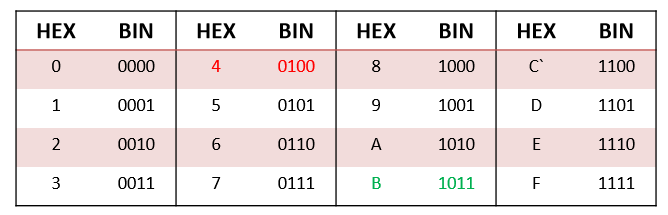


* **Chuyển dãy bit thành số nguyên lớn  
  -Chuyển hệ 2 về hệ 10:**Nguyên tắc: bit MSB bằng 0 thì giá trị bằng giá trị không dấu, bit MSB bằng 1 thì bằng giá trị có dấu trừ đi 2^127



**Cài đặt:  
**

**-Chuyển hệ 2 về hệ 16:  
Nguyên tắc:** Lấy từng cặp 4 bit chuyển về cơ số hệ 16  
**Cài đặt:  
**

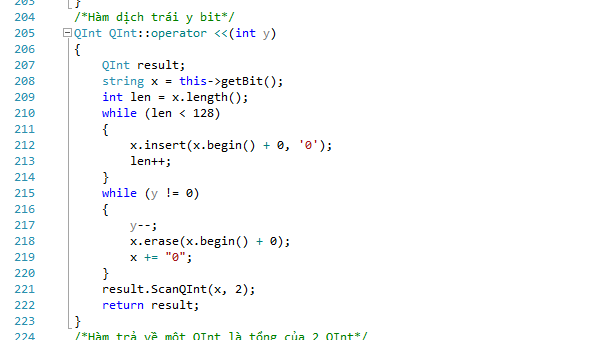


**-Chuyển hệ 10 về hệ 16:**

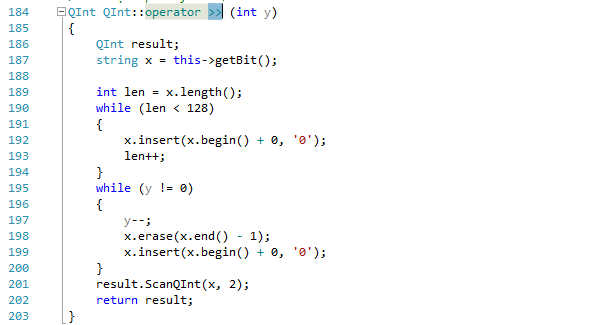
Nguyên tắc: Dùng hệ 2 làm hệ trung gian.  
Chuyển số từ hệ 10 sang hệ 2, rồi từ hệ 2 chuyển sang hệ 16

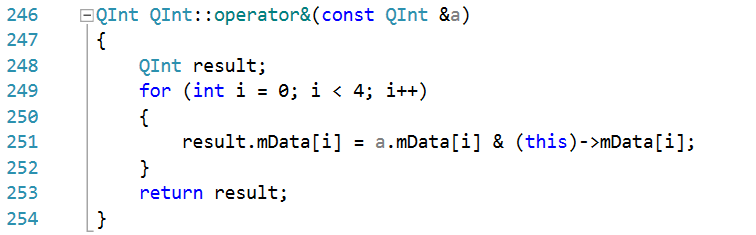
* **Phép dịch bit**

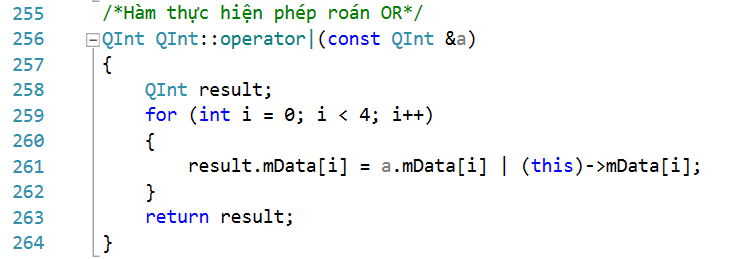
**-Dịch trái: <<**

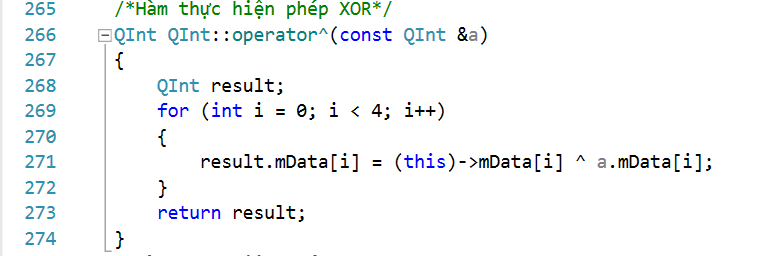
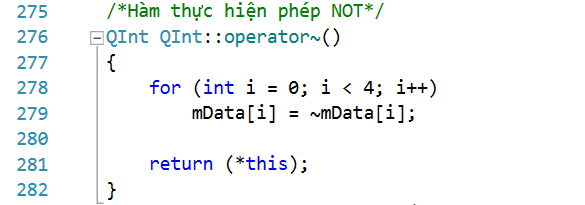
**Nguyên tắc:** Xóa bit trái nhất, sau đó thêm bit 0 vào bit phải nhất  
**Cài đặt:**

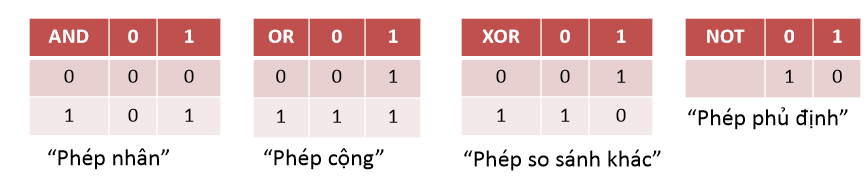
-**Dịch phải: >>**

**Nguyên tắc:** Xoá bit phải nhất, sau đó thêm bit 0 vào bit trái nhất   
**Cài đặt:**

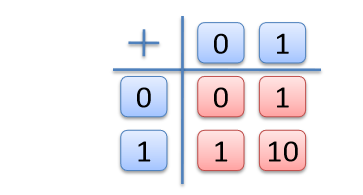
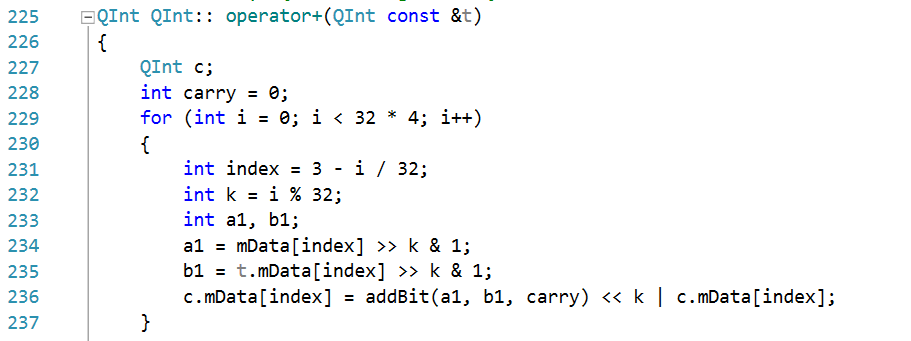
* **Phép AND (&)  
  Nguyên tắc**: Lấy từng phần tử mData **AND** với nhau theo đúng thứ tự.  
  **Cài đặt:**
* **Phép OR (|)**

**Nguyên tắc:** Lấy từng phần tử mData **OR** với nhau theo đúng thứ tự.  
**Cài đặt:**

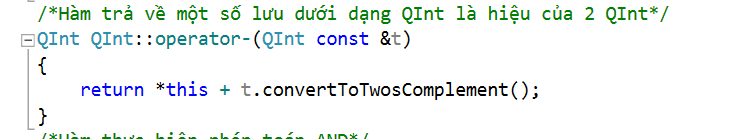
* **Phép XOR (^)  
  Nguyên tắc:** Lấy từng phần tử mData **XOR** với nhau theo đúng thứ tự.  
  **Cài đặt:**
* **Phép NOT(~)  
  Nguyên tắc:** Lấy từng phần tử mData **NOT** với nhau theo đúng thứ tự.  
  **Cài đặt:**   
  



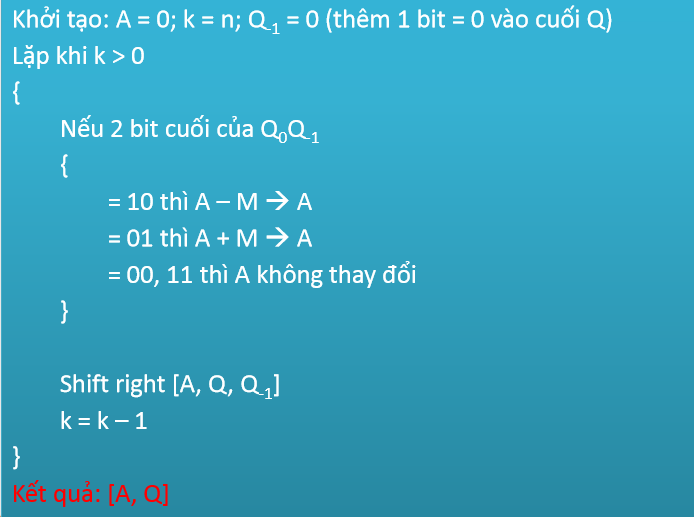
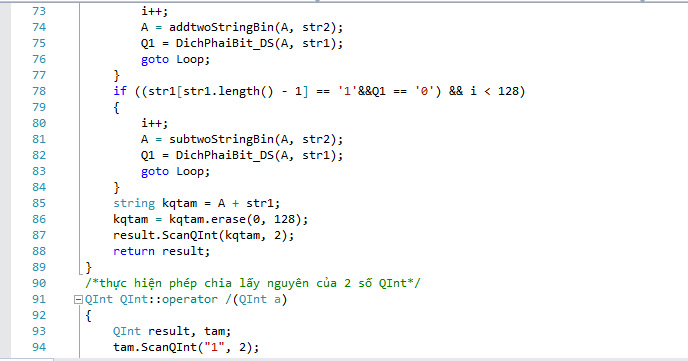
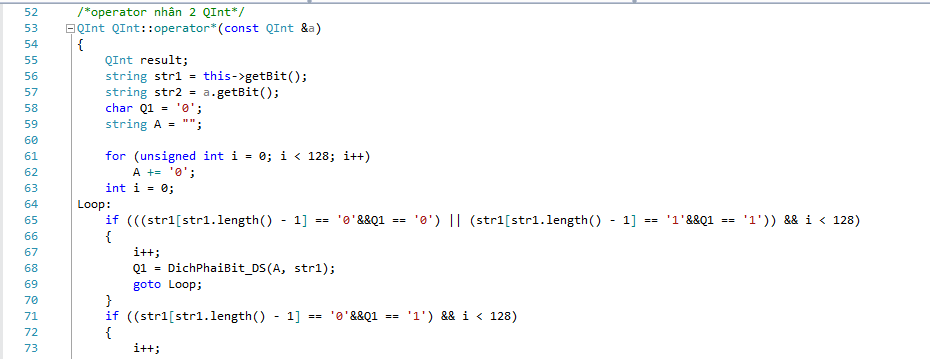
* **Phép cộng (+)**

**Nguyên tắc:** Cộng theo nguyên tắc cộng của số bù 2.  
  
**Cài đặt:**

* **Phép trừ (-)  
  Nguyên tắc:** Cộng số bù 2 của của số bị trừ.

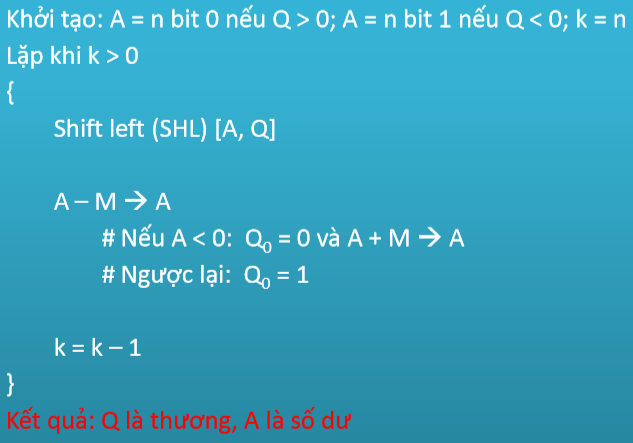
  
**Cài đặt:**

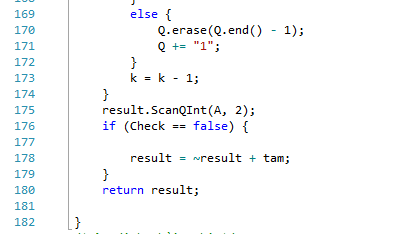
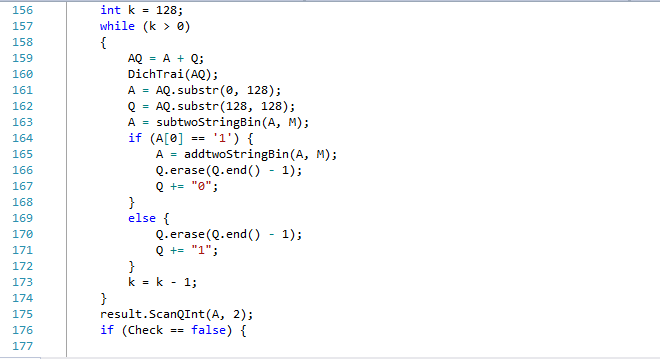
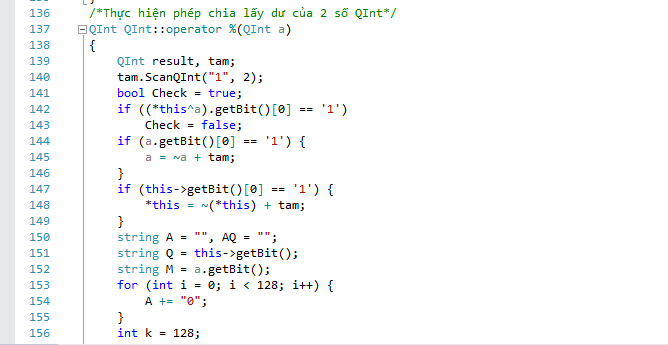
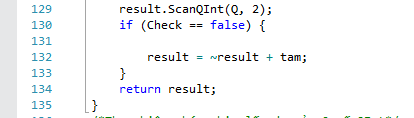
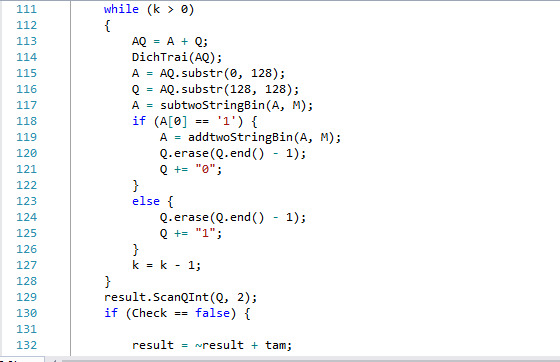
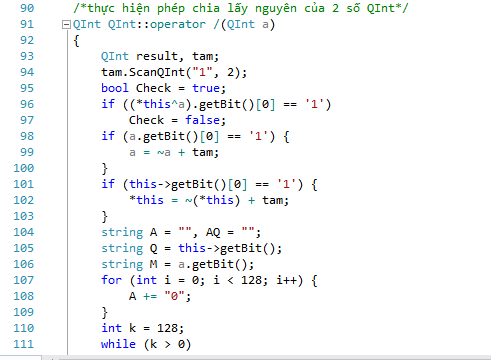
* **Phép nhân (\*)**

**Nguyên tắc:** Nhân theo thuật toán BOOTH.   
  
  
 **Cài đặt:**

* **Phép chia (/)**

**Nguyên tắc:**



**Cài đặt:  
**

# CẤU TRÚC XỬ LÍ CỦA CHƯƠNG TRÌNH

**Chương trình gồm 4 file cơ bản:**

* **Main**Được viết theo cấu trúc commandline:

1612151\_1612126\_1612127\_1612104.exe <input.txt> <output.txt>

* **File**

**Đọc dữ liệu từ file <Input.txt> theo cấu trúc định dạng:**  
-Gồm n dòng (n không biết trước).

-Hệ xử lí (2,10,16), các toán tử (+,-,\*,/,&,|,^,~), các số hạng được cách nhau bằng dấu cách, dòng trên và dòng dưới cách nhau bằng dấu \n.

-Đối với các toán tử +,-,\*,\,&,|,^ sẽ được lưu theo dòng với cấu trúc:  
 <hệ xử lí> <số thứ 1> <toán tử> <số thứ 2>

-Đối với toán tử ~ sẽ được lưu với cấu trúc:

<hệ xử lí> <toán tử> <số hạng>

-Đối với phép toán chuyển đổi cơ số sẽ được lưu với cấu trúc:

<hệ xử lí ban đầu> <hệ xử lí cần chuyển đổi> <số hạng>

**\*\*Lưu ý: Chương trình chỉ xử lí được một phép tính hoặc chuyển đổi trên mỗi dòng.**

**Ghi dữ liệu vào file <Output.txt> theo định dạng:**  
<Kết quả phép tính 1>  
<Kết quả phép tính 2>  
…………………………………

<Kết quả phép tính n>

* **Qint**

**Gồm các hàm xử lí với các hàm chính sau:**

1. ScanQInt.
2. PrintQInt.
3. Các hàm chuyển đổi: binToHex, binToDec.
4. Các toán tử +,-,\*,/,&,^,|,~,<<,>>.
5. Xử lí trên từng chuỗi đọc vào từ file

* **Menu**

Xây dựng menu chương trình, xuất hiện menu khi không nhập file input.txt và output.txt.

## **Phạm vi biểu diễn của kiểu dữ liệu thiết kế**

**Kiểu dữ liệu QInt được thiết kế có thể chứa 128 bit, bao gồm cả bit dấu.**

### Max QInt

Số dương lớn nhất có thể biểu diễn gồm 127 bit 1 và 1 bit 0.  
Giá trị biểu diễn:

+…++=170141183460469231731687303715884105727

### min qint

Số âm bé nhất có thể biểu diễn gồm 1 bit 1 và 127 bit 0 (bit 1 ở vị trí 0).

Giá trị biểu diễn: = -170141183460469231731687303715884105728

## **ĐÁNH GIÁ MỨC ĐỘ HOÀN THÀNH**

* **Đối với tính năng lưu trữ số nguyên lớn:**

-Lưu thành công vào mảng mData (kể cả số âm).

-Giới hạn biểu diễn 128 bit (kể cả bit dấu).

-Lấy thành công các bit trong mảng để xử lí.

**Đánh giá tính năng hoàn thành:** 100%

* **Đối với tính năng chuyển đổi hệ cơ số**  
  -Chuyển đổi thành công hệ 2 sang hệ 10.

-Chuyển đổi thành công hệ 2 sang hệ 16.

-Chuyển đổi thành công hệ 10 sang hệ 16.

**Đánh giá tính năng hoàn thành:** 100%

* **Đối với tính năng thực hiện các toán tử**

-Thực hiện thành công toán tử +

-Thực hiện thành công toán tử -

-Thực hiện thành công toán tử \*

-Thực hiện thành công toán tử /

-Thực hiện thành công toán tử &

-Thực hiện thành công toán tử |

-Thực hiện thành công toán tử ^

**-Thực hiện thành công toán tử ~**

-Thực hiện thành công toán tử >>

-Thực hiện thành công toán tử <<

(Thực hiện chính xác trên số nguyên âm và dương)

**Đánh giá tính năng hoàn thành:** 100%

* **Đối với Menu chương trình**

Tự đánh giá: khá hoàn thiện.

**Đánh giá tính năng hoàn thành:** 90%

* **Đối với xử lí file**

-Đọc file: đọc thành công, xử lí trên từng chuỗi đọc vào tốt.

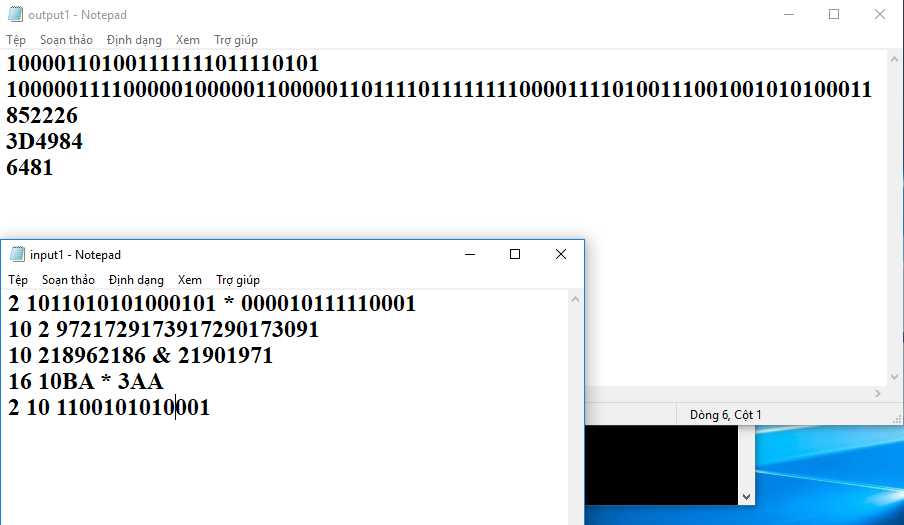
-Ghi file: ghi thành công, ghi thành từng dòng kết quả.

**Đánh giá mức độ hoàn thành:** 100%

### **ĐÁNH GIÁ MỨC ĐỘ HOÀN THÀNH TRÊN TOÀN BỘ CHƯƠNG TRÌNH: 99%**

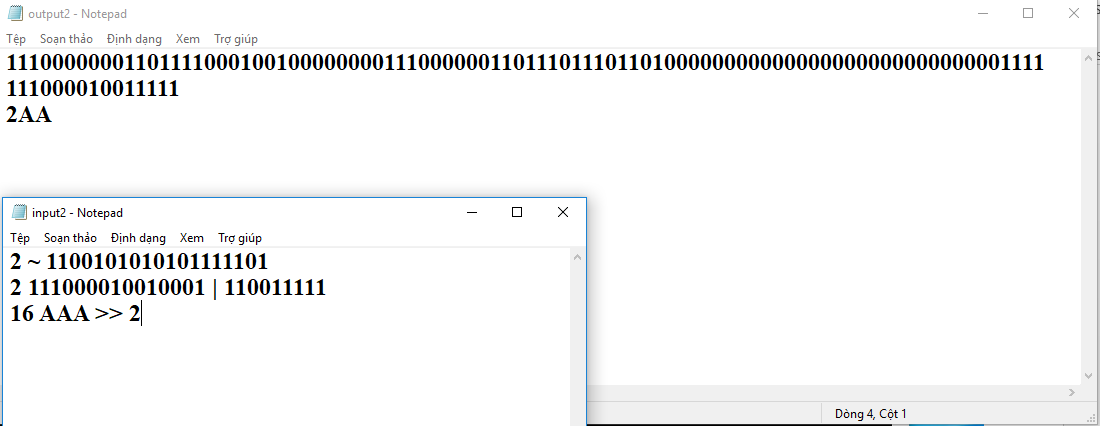
## **TEST CASE**

* **CASE 1:**

****

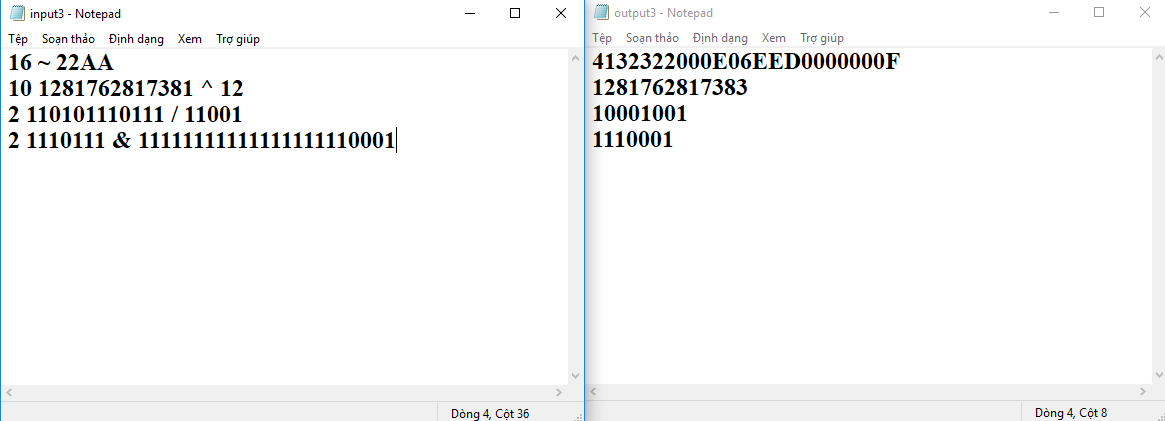
**File input1.txt File output1.txt**

* **CASE 2:**

****

**File input2.txt File output2.txt**

* **CASE 3:**

****

**File input3.txt File output3.txt**

## **KHÓ KHĂN GẶP PHẢI**

**-**Nguồn tài liệu về đề tài quá ít.

-Chưa hoàn thiện kĩ năng làm việc nhóm.

-Thời tiết ảnh hưởng đến quá trình làm việc.

**TÀI LIỆU THAM KHẢO**

1. [**https://www.geeksforgeeks.org/sum-two-large-numbers/**](https://www.geeksforgeeks.org/sum-two-large-numbers/) **(tổng 2 số nguyên lớn)**
2. [**http://calc.penjee.com/**](http://calc.penjee.com/) **(test kết quả)**
3. [**https://stackoverflow.com/questions/28493153/bigint-subtraction-in-c**](https://stackoverflow.com/questions/28493153/bigint-subtraction-in-c) **(trừ 2 số nguyên lớn)**
4. **http://www.cplusplus.com/forum/beginner/56063/  
   (chuyển từ hệ 2 sang hệ 10)**
5. [**https://github.com/rfaulkner/Big-Number**](https://github.com/rfaulkner/Big-Number)
6. [**https://inspirit.net.in/books/academic/Computer%20Organisation%20and%20Architecture%208e%20by%20William%20Stallings.pdf**](https://inspirit.net.in/books/academic/Computer%20Organisation%20and%20Architecture%208e%20by%20William%20Stallings.pdf) **(Ebook của W.Stalling)**