

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**  
**KHOA CÔNG NGHỆ THÔNG TIN**  
**LỚP 18\_4**



**BÁO CÁO ĐỒ ÁN 1**  
**BIỂU DIỄN VÀ TÍNH TOÁN SỐ NGUYÊN LỚN**

**Giáo viên hướng dẫn:**  
Lê Quốc Hòa

**Sinh viên thực hiện:**

1. Lê Chí Sơn. Mã số sinh viên: 18120535
2. Phạm Văn Thật. Mã số sinh viên: 18120568
3. Phạm Minh Vương. Mã số sinh viên: 18120655

**TP.HCM, tháng 6/2020**

**1. Bảng phân công công việc:**

<b>Họ và tên</b>	<b>Mã số sinh viên</b>	<b>Công việc</b>
Lê Chí Sơn	18120535	<ul style="list-style-type: none"><li>- Xử lý các phép toán với các operator +, -, &gt;&gt;, &lt;&lt;.</li><li>- Phép chuyển chuỗi kí tự hệ 16 thành 128 bit lưu biến kiểu QInt.</li><li>- Chuyển 128 bit được lưu trong QInt thành chuỗi kí tự biểu diễn giá trị của biến QInt ở hệ 16.</li></ul>
Phạm Văn Thật	18120568	<ul style="list-style-type: none"><li>- Xử lý các phép toán với các operator &amp;,  , ^, ~, rol, ror.</li><li>- Phép chuyển chuỗi kí tự hệ 10 thành 128 bit lưu trong biến kiểu QInt.</li><li>- Chuyển biến QInt thành chuỗi kí tự biểu diễn cho giá trị của nó ở hệ 10.</li></ul>
Phạm Minh Vương	18120655	<ul style="list-style-type: none"><li>- Lên kế hoạch cho nhóm, phân chia quá trình cài đặt và kiểm thử đồ án, phân chia công việc cho cả nhóm.</li><li>- Tổng hợp mã code của nhóm và build chương trình.</li><li>- Xử lý các phép toán với các operator *, /.</li><li>- Phép chuyển chuỗi ký tự hệ 2 thành 128 bit lưu trong biến kiểu QInt.</li><li>- Chuyển biến QInt thành chuỗi kí tự biểu diễn cho giá trị của nó ở hệ 2.</li><li>- Xây dựng hàm đọc file, xử lý tính toán trên dòng và ghi kết quả ra file.</li></ul>

**2. Môi trường lập trình:**

- Chương trình soạn thảo: Visual Studio.
- Chương trình dịch: Visual Studio.
- Các công cụ tìm kiếm, sửa lỗi...: Visual Studio.

**3. Ý tưởng thiết kế và thực hiện đồ án:**

- Để biểu diễn và tính toán số nguyên lớn thì nhóm đã lên ý tưởng thiết kế một kiểu dữ liệu mới là “QInt” với kích thước là 128bit.

- Được biểu diễn như sau:

```
class QInt
{
private:
    unsigned int arrayBits[4];
}
```

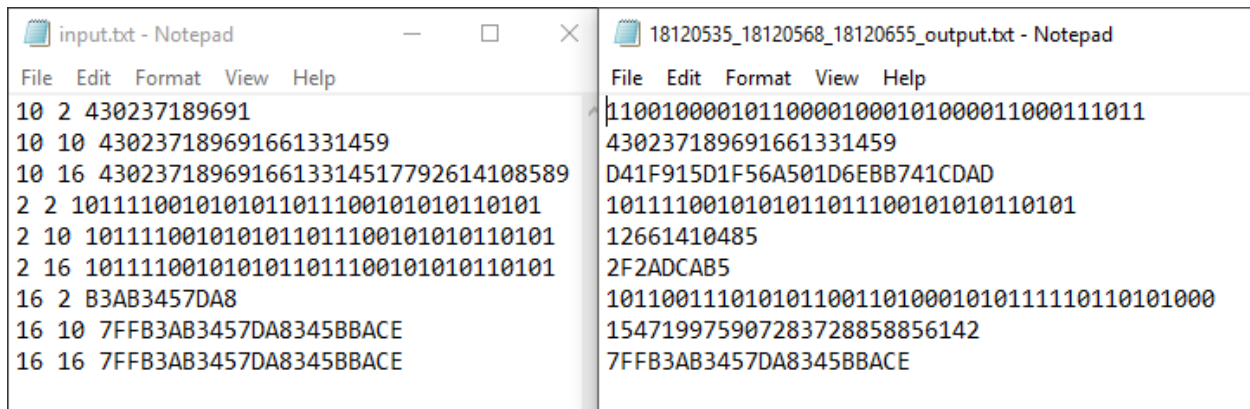
- Với một mảng gồm 4 phần tử kiểu “*unsigned int*” mỗi phần tử lần lượt sẽ cung cấp vùng nhớ gồm 32bit cho số nguyên cần biểu diễn.
- Các cơ số (2, 10, 16) được truyền vào đều được chuyển về dạng 128 bit và lưu trữ trong thuộc tính arrayBits của QInt.
- Qua đó các phép toán số học trên các cơ số thực chất là được thực hiện ở hệ 2 thông qua các chuỗi bits tạm, kết quả tính toán được truyền lại vào biến QInt kết quả dưới dạng hệ 2 và trả ra một QInt kết quả.
- Với yêu cầu trả về kết quả ở dạng hệ 2, hệ 10 và hệ 16 chương trình sẽ có từng phương thức riêng biệt tương ứng để chuyển 128 bits trong QInt thành chuỗi số có giá trị tương ứng với từng hệ mà người dùng yêu cầu.
- Do đó phép chuyển đổi qua lại giữa các hệ đều thông qua thành phần trung gian là QInt.

#### 4. Chạy kiểm tra và chú thích chức năng:

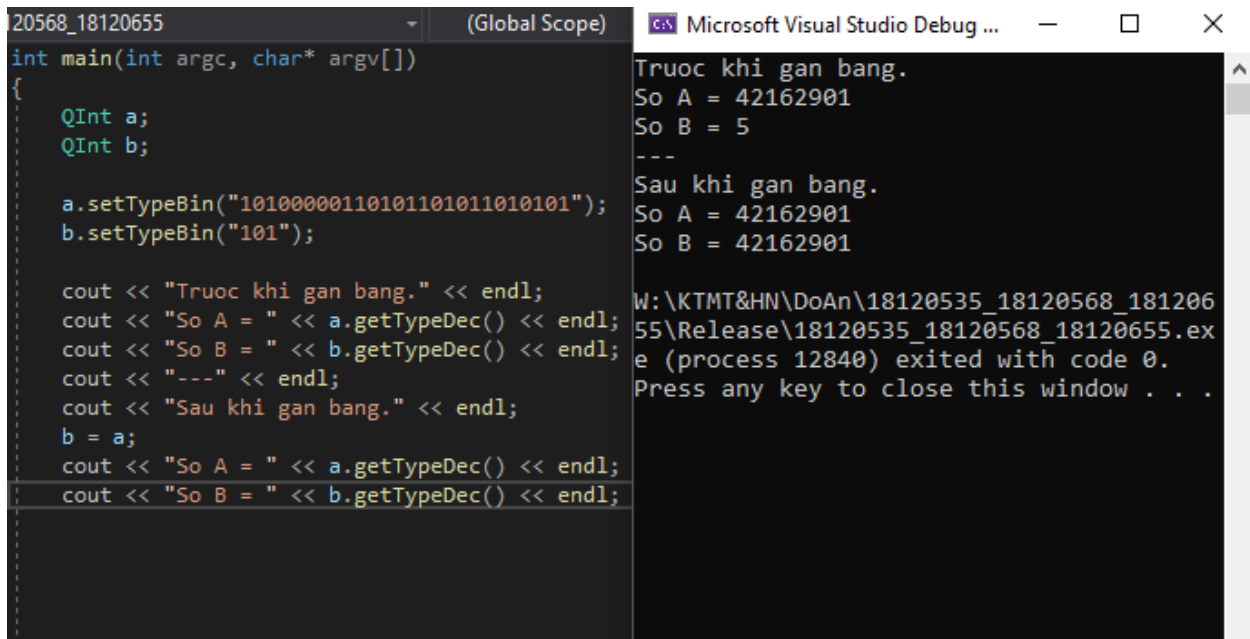
input.txt - Notepad	18120535_18120568_18120655_output.txt - Notepad
File Edit Format View Help	File Edit Format View Help
10 4647 + 4302371896916613314517792614108589	4302371896916613314517792614113236
2 1010 + 101	1111
16 7FFB3A + BD35F	8BCE99
10 4357 - 2626237189691661331451779261410835	-2626237189691661331451779261406478
2 1001 - 101	100
16 8AFDB53FBAD5 - 7B2747BA8	8AF602CB3F2D
10 -4647 * 4302371896916613314517792614108589	-19993122204971502072564182277762613083
2 1010 * 101	110010
16 7F * ABCD35F	553ACDC21
10 4302371896916613314517792614108589 / 8578	501558859514643659887828469819
2 1010 / 101	10
16 7FFB3A / BD35F	A

Hình 1: Chức năng cộng "+", trừ "-", nhân "\*", chia "/" trên các hệ số.





Hình 4: Minh họa các chức năng chuyển đổi qua lại giữa các hệ số.



Hình 5: Minh họa phương thức gán bằng "=".

## 5. Chức năng đã làm được:

- Chuyển đổi số QInt từ hệ thập phân sang hệ nhị phân (dạng bù 2) và ngược lại
- Chuyển đổi số QInt từ hệ nhị phân (dạng bù 2) sang hệ thập lục phân và ngược lại
- Chuyển đổi số QInt từ hệ thập lục phân sang thập phân và ngược lại
- Các operator =, operator +, operator -, operator \*, operator / trên các hệ cơ số
- Các toán tử AND "&", OR "|", XOR "^", NOT "~"
- Các toán tử: dịch trái "<<" và dịch phải ">>" số học
- Các phép xoay trái "rol", xoay phải "ror" mỗi lần xoay chỉ xử lý cho đúng 1 bit, không cần xử lý cho trường hợp tổng quát xoay k bit

## 6. Chức năng chưa làm được:

- Không có chức năng nào chưa làm được.
- 7. **Mức độ hoàn thành đồ án:** *100%*.
- 8. **Nguồn tài liệu tham khảo:**
  - Slide bài giảng môn Kiến trúc máy tính và hợp ngữ.