TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA CÔNG NGHỆ THÔNG TIN



**BỘ MÔN: KIẾN TRÚC MÁY TÍNH VÀ HỢP NGỮ**

**BÁO CÁO ĐỒ ÁN 1**

ĐỀ TÀI: Biểu diễn và tính toán số học trên máy tính

GVHD: cô Chung Thùy Linh

Hồ Chí Minh, ngày 1 tháng 6 năm 2020

**MỤC LỤC**

[**I.** **BÁO CÁO VỀ NHÓM :** 3](#_Toc40038335)

[**1.** **THÀNH VIÊN** 3](#_Toc40038336)

[**2.** **PHÂN CÔNG** 3](#_Toc40038337)

[**3.** **TỶ LỆ HOÀN THÀNH** 3](#_Toc40038338)

[**II.** **BÁO CÁO** **VỀ ĐỒ ÁN** 4](#_Toc40038339)

[**1.** **ĐÁNH GIÁ MỨC ĐỘ HOÀN THÀNH** 4](#_Toc40038340)

[**2.** **PHẠM VI BIỂU DIỄN CỦA CÁC KIỂU DỮ LIỆU** 4](#_Toc40038341)

[**3.** **CÁC TEST CASE** 5](#_Toc40038342)

[**4.** **CÁC NGUỒN THAM KHẢO** 8](#_Toc40038345)

1. **BÁO CÁO VỀ NHÓM:**
2. **THÀNH VIÊN**

|  |  |
| --- | --- |
| Họ và tên | MSSV |
| Nguyễn Đình Trí | 18120611 |
| Hồ Hoàng Thương | 18120586 |
| Lưu Trường Vũ | 18120653 |

1. **PHÂN CÔNG**

|  |  |
| --- | --- |
| Họ và tên | Phân công |
| Nguyễn Đình Trí | Xử lý AND, OR, XOR, NOT; chuyển đổi hệ số, viết báo cáo |
| Hồ Hoàng Thương | Chuyển đổi sang QInt, xử lý cộng trừ nhân chia |
| Lưu Trường Vũ | Xử lý xoay, dịch bit, đọc ghi file |

1. **TỶ LỆ HOÀN THÀNH**

|  |  |
| --- | --- |
| Họ và tên | Tỉ lệ hoàn thành |
| Nguyễn Đình Trí | 100% |
| Hồ Hoàng Thương | 100% |
| Lưu Trường Vũ | 100% |

1. **BÁO CÁO** **VỀ ĐỒ ÁN**

Link github đồ án: [*https://github.com/tringuyen1092000/Project1\_ComputerArchitecture*](https://github.com/tringuyen1092000/Project1_ComputerArchitecture)

1. **ĐÁNH GIÁ MỨC ĐỘ HOÀN THÀNH**

|  |  |  |
| --- | --- | --- |
|  | **Yêu cầu** | **Mức độ hoàn thành** |
| **QInt** | Nhập file, xuất file | 100% |
| Chuyển đổi các hệ cơ số | 100% |
| Toán tử + - \* / | 100% |
| Chuyển đổi giữa QInt và String | 100% |
| Toán tử & ^ | ~ | 100% |
| Toán tử dịch >> << | 100% |
| Rol, ror | 100% |

1. **PHẠM VI BIỂU DIỄN CỦA CÁC KIỂU DỮ LIỆU**

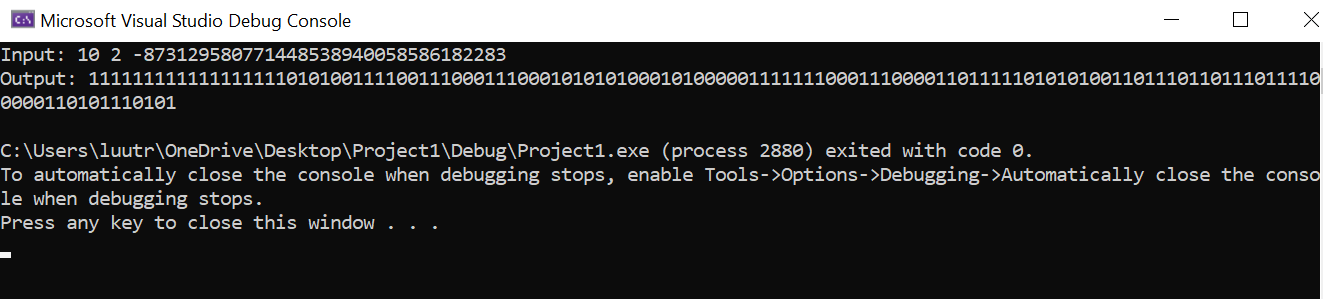
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Kích thước |  | Phạm vi biểu diễn (\*) | |
| Min | Max |
| QInt | 16 byte |  | -2^127 | 2^127-1 |

(\*)

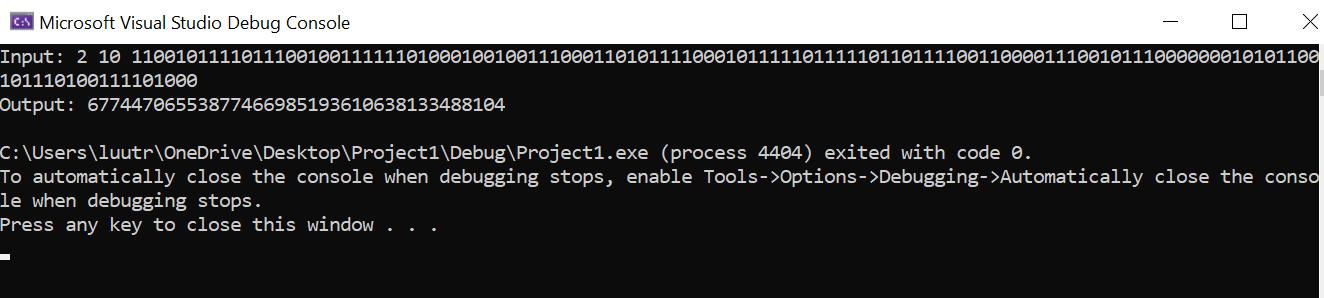
-2^127 = -170141183460469231731687303715884105728

2^127-1 = 170141183460469231731687303715884105727

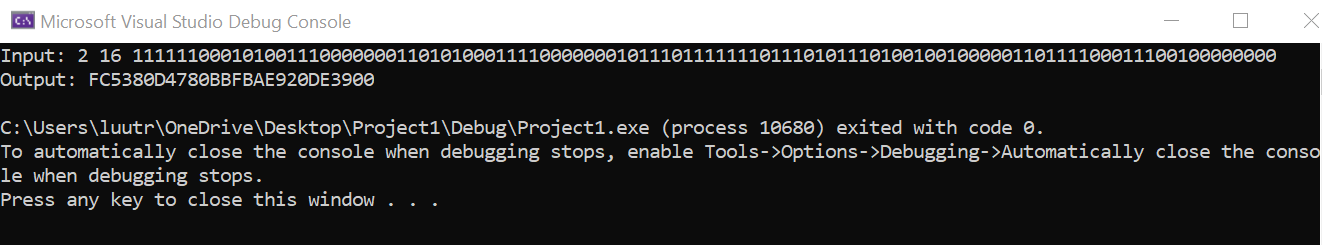
1. **CÁC TEST CASE**
2. -873129580771448538940058586182283

Task: DEC to BIN

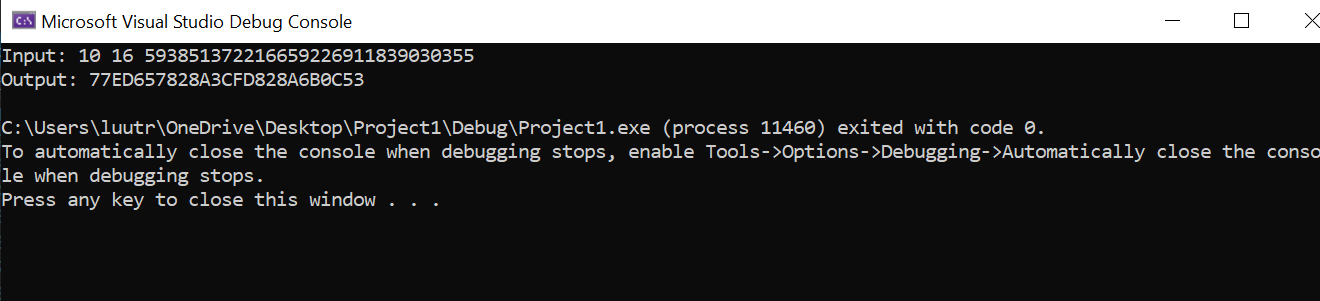
1. 110010111101110010011111101000100100111000110101111000101111101111101101111001100001110010111000000010101100101110100111101000

Task: BIN to DEC

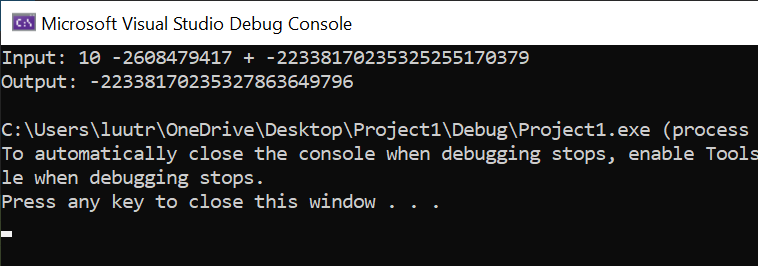
1. 11111100010100111000000011010100011110000000101110111111101110101110100100100000110111100011100100000000

Task: BIN to HEX

1. 593851372216659226911839030355

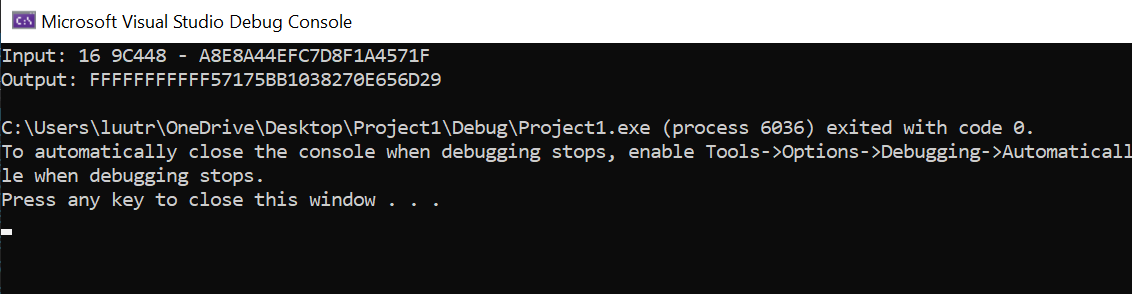
Task: DEC to HEX

1. -2608479417 + -22338170235325255170379

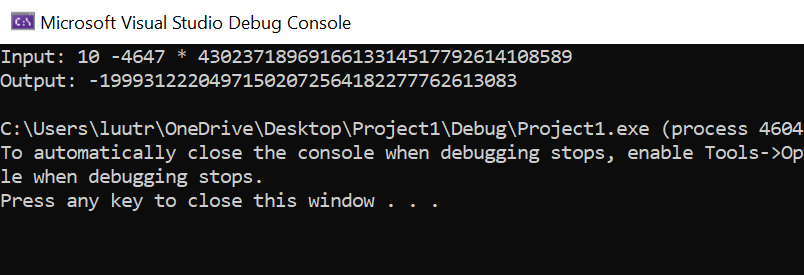
Task: PLUS DEC

1. 9C448 - A8E8A44EFC7D8F1A4571F

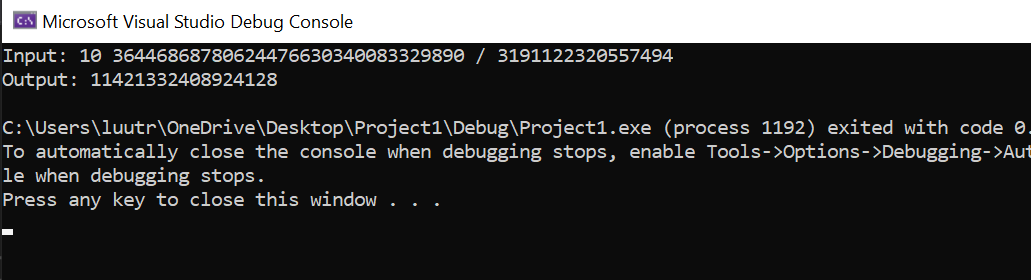
Task: MINUS HEX



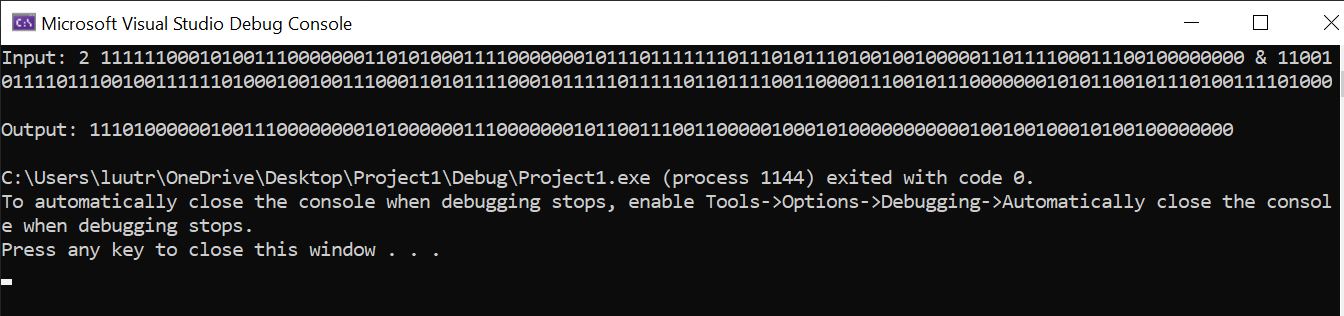
1. -4647 \* 4302371896916613314517792614108589

Task: MULTI DEC

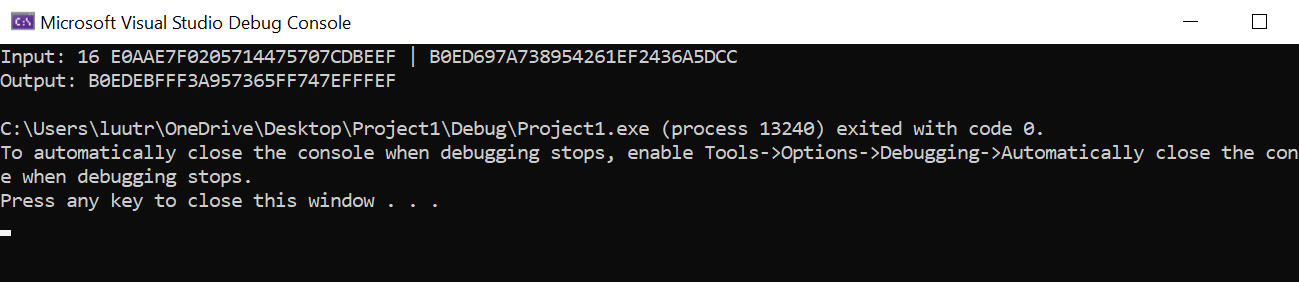
1. 36446868780624476630340083329890 / 3191122320557494

Task: DIV DEC

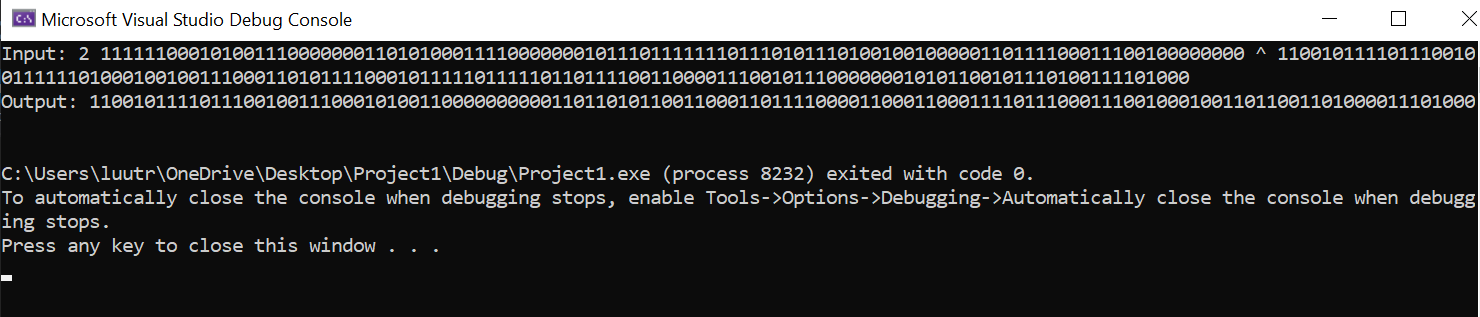
1. 11111100010100111000000011010100011110000000101110111111101110101110100100100000110111100011100100000000 & 110010111101110010011111101000100100111000110101111000101111101111101101111001100001110010111000000010101100101110100111101000

Task: AND (&) BIN

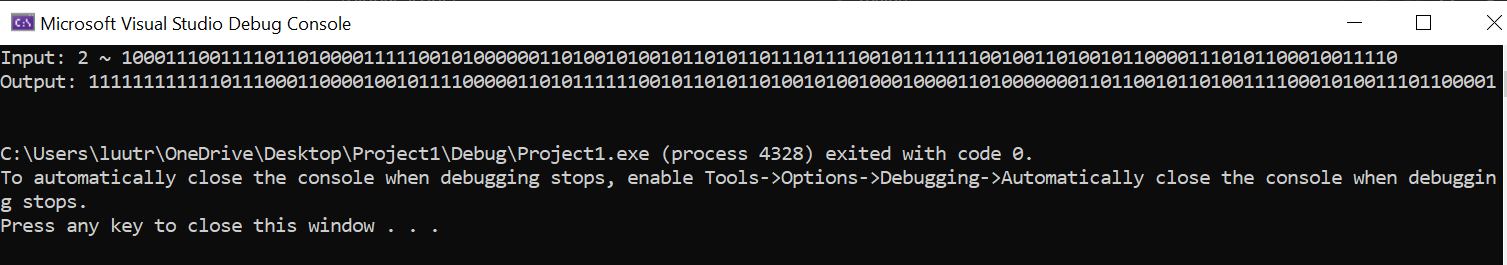
1. E0AAE7F0205714475707CDBEEF | B0ED697A738954261EF2436A5DCC

Task: OR (|) HEX

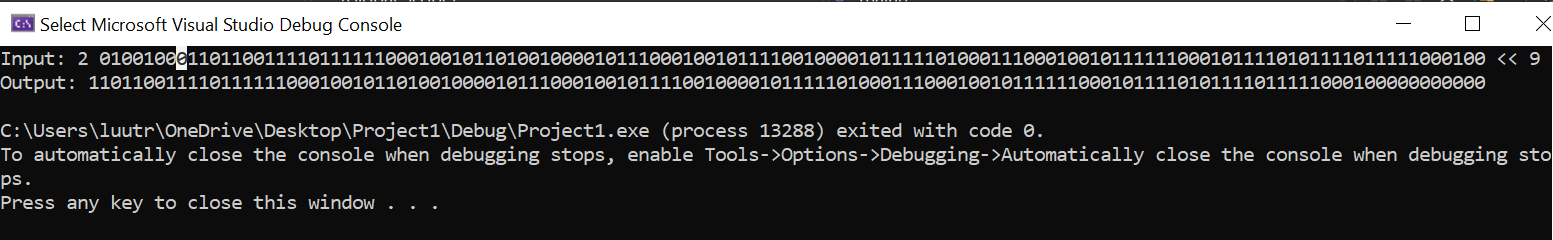
1. 11111100010100111000000011010100011110000000101110111111101110101110100100100000110111100011100100000000 ^ 110010111101110010011111101000100100111000110101111000101111101111101101111001100001110010111000000010101100101110100111101000

Task: XOR (^) BIN

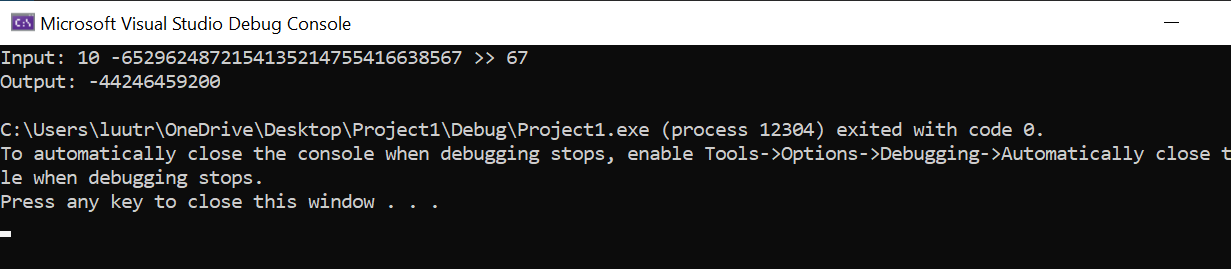
1. ~ 10001110011110110100001111100101000000110100101001011010110111011110010111111100100110100101100001110101100010011110

Task: NOT (~) BIN

1. 010010001101100111101111110001001011010010000101110001001011110010000101111101000111000100101111110001011110101111011111000100 << 9

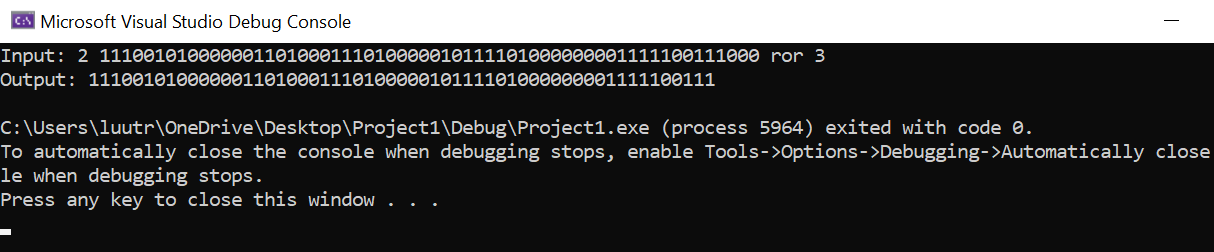
Task: SHIFT LEFT BIN

1. -6529624872154135214755416638567 >> 67

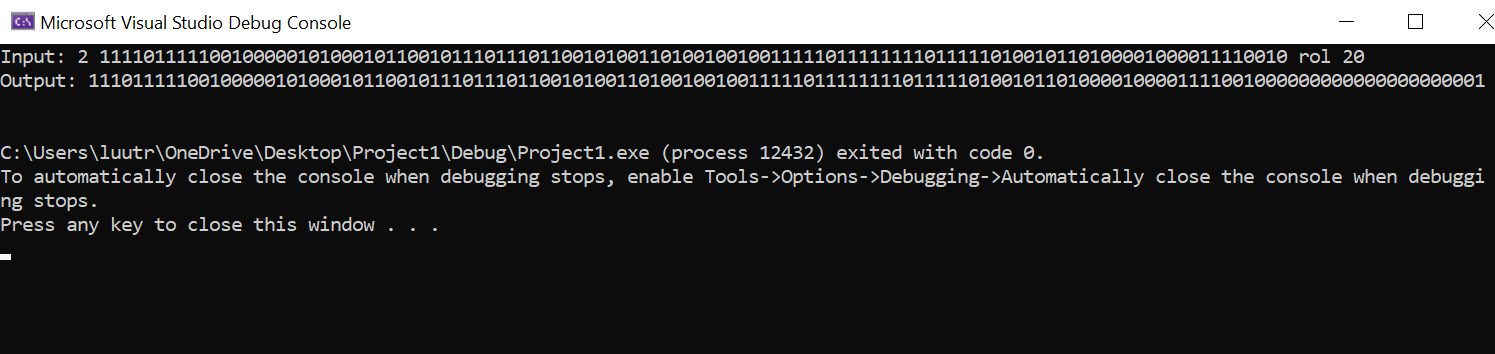
Task: SHIFT RIGHT DEC

1. 111001010000001101000111010000010111101000000001111100111000 ror 3

Task: ROR BIN



1. 111101111100100000101000101100101110111011001010011010010010011111011111111011111010010110100001000011110010 rol 20

Task: ROL BIN

1. **CÁC NGUỒN THAM KHẢO**

* Wolframalpha:[*https://www.wolframalpha.com/*](https://www.wolframalpha.com/)
* Cplusplus:[*http://www.cplusplus.com/*](http://www.cplusplus.com/)
* Các phép toán căn bản trên số nhị phân: [*https://expressmagazine.net/posts/view/1252/cac-phep-toan-can-ban-tren-so-nhi-phan*](https://expressmagazine.net/posts/view/1252/cac-phep-toan-can-ban-tren-so-nhi-phan)
* Slide ch02 - Biểu diễn số nguyên (KHOA CNTT - ĐH KHTN): [*http://www.mediafire.com/file/scbt0q3zuz32cfz/Ch02\_Bieu\_dien\_so\_nguyen.pdf/file*](http://www.mediafire.com/file/scbt0q3zuz32cfz/Ch02_Bieu_dien_so_nguyen.pdf/file)
* Geeksforgeeks (chia số nguyên lớn): [*https://www.geeksforgeeks.org/divide-large-number-represented-string/*](https://www.geeksforgeeks.org/divide-large-number-represented-string/?fbclid=IwAR3_rIGvgDP4Cb-8o4NW5zhnej1_mCQiYWys4GDpL6Z1DsMJHmVqHBLzAlQ)
* Geeksforgeeks (hàm mũ cho số nguyên lớn): [*https://www.geeksforgeeks.org/writing-power-function-for-large-numbers/*](https://www.geeksforgeeks.org/writing-power-function-for-large-numbers/?fbclid=IwAR2WGaYZ1l8i5GVrDIkZJKqYHG76PA0szACqYyoLiewjemARf7Su1wEaBDo)