**ĐẠI HỌC QUỐC GIA TPHCM**

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**KHOA CÔNG NGHỆ THÔNG TIN**

**---------🙠🙠🙠---------**



**BÁO CÁO**

**ĐỒ ÁN: BIỂU DIỄN VÀ TÍNH TOÁN SỐ HỌC TRÊN MÁY TÍNH**

Lớp: 17CTT5

Thông tin thành viên nhóm:

1. Đỗ Tấn Tài MSSV: 1712737
2. Huỳnh Công Sinh MSSV: 1712724
3. Trần Thiên Quàng MSSV: 1712709
4. Nguyễn Văn Thìn MSSV: 1712787
5. Phan Đại Hải MSSV: 1712410

GVHD: ThS. Lê Viết Long

Hồ Chí Minh, Tháng 03-2019

**ĐẠI HỌC QUỐC GIA TPHCM**

**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**

**KHOA CÔNG NGHỆ THÔNG TIN**

**---------🙠🙠🙠---------**



**BÁO CÁO**

**ĐỒ ÁN: BIỂU DIỄN VÀ TÍNH TOÁN SỐ HỌC TRÊN MÁY TÍNH**

Lớp: 17CTT5

Thông tin thành viên nhóm:

1. Đỗ Tấn Tài MSSV: 1712737
2. Huỳnh Công Sinh MSSV: 1712724
3. Trần Thiên Quàng MSSV: 1712709
4. Nguyễn Văn Thìn MSSV: 1712787
5. Phan Đại Hải MSSV: 1712410

GVHD: ThS. Lê Viết Long

Hồ Chí Minh, Tháng 03-2019

# MỤC LỤC

[MỤC LỤC 1](#_Toc4337622)

[NỘI DUNG 1](#_Toc4337623)

[**I. PHÂN TÍCH VÀ Ý TƯỞNG THIẾT KẾ CÁC THUẬT TOÁN, CẤU TRÚC 1**](#_Toc4337624)

[**1. Cấu trúc của kiểu QInt 1**](#_Toc4337625)

[**2. Ý tưởng của các thuật toán thực hiện chức năng chuyển đổi qua lại giữa các hệ cơ số và các toán tử trên kiểu Qint 1**](#_Toc4337626)

[**3. Cấu trúc của kiểu QFloat 3**](#_Toc4337627)

[**4. Ý tưởng của các thuật toán thực hiện chức năng chuyển đổi qua lại giữa các hệ cơ số và các toán tử trên kiểu QFloat 4**](#_Toc4337628)

[**II. PHẠM VI BIỂU DIỄN CỦA CÁC KIỂU DỮ LIỆU ĐÃ THIẾT KẾ 5**](#_Toc4337629)

[**1. QInt 5**](#_Toc4337630)

[**2. QFloat 5**](#_Toc4337631)

[**III. GIAO DIỆN ỨNG VỚI CÁC BỘ TESTCASE 5**](#_Toc4337632)

[**1. QInt 5**](#_Toc4337633)

[**2. QFloat 8**](#_Toc4337634)

[**IV. ĐÁNH GIÁ MỨC ĐỘ HOÀN THÀNH 8**](#_Toc4337635)

[**1. Mức độ hoàn thành ứng với các yêu cầu chi tiết của dự án 8**](#_Toc4337636)

[**2. Mức độ hoàn thành ứng với toàn bộ dự án 8**](#_Toc4337637)

[DANH MỤC TÀI LIỆU THAM KHẢO 10](#_Toc4337638)

[BIÊN BẢN PHÂN CÔNG VIỆC NHÓM 11](#_Toc4337639)

[**1. Thông tin chung 11**](#_Toc4337640)

[**2. Bảng phân công công việc 11**](#_Toc4337641)

**NỘI DUNG**

## I. PHÂN TÍCH VÀ Ý TƯỞNG THIẾT KẾ CÁC THUẬT TOÁN, CẤU TRÚC

1. Cấu trúc của kiểu QInt

\* Mô tả cấu trúc kiểu dữ liệu QInt.

* Xây dựng từ cấu trúc dữ liệu bitset có sẵn trong thư viện chuẩn STL của C++.
* Kiểu dữ liệu QInt được xây dựng chính là một bitset<128> dùng để lưu 128 bit biểu diễn dạng nhị phân của một số nguyên.

\* Lí do chọn bitset<128> để biểu diễn số nguyên 16 byte.

* Thỏa mãn yêu cầu đề bài, sizeof của bitset<128> là 16 byte.
* Dễ dàng truy cập đến bit thứ k trong dãy 128 bit, thuận tiện cho việc lập trình các chức năng.
* Hỗ trợ sẵn một số toán tử thông dụng.

### 2. Ý tưởng của các thuật toán thực hiện chức năng chuyển đổi qua lại giữa các hệ cơ số và các hàm, toán tử trên kiểu Qint

**\* Hàm nhập, xuất**

* Hàm nhập nhận tham số là một string, Chia dần số thập phân (chia chuỗi bằng hàm tự viết) cho 2 cho đến khi kết quả bằng 0, mỗi lần chia lấy số dư phép chia gán cho bit theo thứ tự. Nếu là số âm thì đảo bit kết quả nhận được rồi + 1 vào kết quả vừa nhận được, ta thu được dạng bù 2 biểu diễn số âm. Lưu kết quả nhận được vào bitset<128>.
* Hàm xuất nhận tham số là 1 biến QInt, kiểm tra xem có âm hay không bằng việc xét bit đầu tiên, nếu âm thì lấy chuỗi bit -1 và đảo bit để nhận được dãy nhị phân biễu diễn giá trị tuyệt đối của số nguyên đó. Sau đó chuyển sang hệ 10 như việc thực hiện tính tay với sự hỗ trợ của hàm cộng hai chuỗi và hàm tính lũy thừa của 2 trả về 1 chuỗi, thêm dấu - nếu là số âm ta nhận được kết quả.

**\* Các phép chuyển hệ cơ số**

* Chuyển đổi từ hệ cơ số 10 sang hệ cơ số 2: Chia dần số thập phân (chia chuỗi bằng hàm tự viết) cho 2 cho đến khi kết quả bằng 0, mỗi lần chia lấy số dư phép chia gán cho bit theo thứ tự. Nếu là số âm thì đảo bit kết quả nhận được rồi + 1 vào kết quả vừa nhận được, ta thu được dạng bù 2 biểu diễn số âm.
* Chuyển đổi từ hệ cơ số 2 sang hệ thập 10: Kiểm tra số hệ 2 có âm hay không bằng việc xét bit đầu tiên, nếu âm thì lấy chuỗi bit -1 và đảo bit để nhận được dãy nhị phân biễu diễn giá trị tuyệt đối của số nguyên đó. Sau đó chuyển sang hệ 10 như việc thực hiện tính tay với sự hỗ trợ của hàm cộng hai chuỗi và hàm tính lũy thừa của 2 trả về 1 chuỗi, thêm dấu - nếu là số âm ta nhận được kết quả.
* Chuyển đổi từ hệ cơ số 16 sang hệ cơ số 2: Với mỗi kí tự trong số hệ 16, chuyển thành 4 bit tương ứng trong hệ nhị phân, lặp lại cho đến hết.
* Chuyển đổi từ hệ cơ số 2 sang hệ cơ số16: Xét mỗi 4 bit trong số hệ 2 rồi chuyển thành mã hexa tương ứng, lặp lại cho đến hết.

**\* Các toán tử: Tất cả quá trình tính toán đều thực hiện ở hệ cơ số 2 rồi đổi vệ hệ cơ số mà người dùng yêu cầu.**

* Phép gán =: chuyển hết về nhị phân, sau đó set lại các bit theo chuỗi nhị phân đó.
* Phép cộng: Đi từng bit từ phải qua trái, thực hiện cộng 2 bit của 2 QInt truyền vào theo quy tắc cộng ở hệ nhị phân, đồng thời sử dụng một biến nhớ để lưu giá trị nhớ và lần lượt thực hiện như trên cho đến hết.
* Phép trừ: Ta coi a - b như a + (-b). Vì vậy, chuyển b thành –b bằng cách chuyển b về dạng bù 2 (nhờ sự hỗ trợ của toán tử ~ và hàm +1 vào dãy bit). Sau đó sử dụng toán tử + ở trên để thực hiện phép toán a + (-b), ta nhận được kết quả phép trừ.
* Phép nhân: Sử dụng thuật toán Booth với sự hỗ trợ của toán tử dịch phải.
* Phép chia: Kết hợp hàm chia 2 số nguyên dương với việc xét dấu số bị chia và số chia, ta nhận được kết quả như mong muốn.
* Toán tử AND &: Đi từng bit của QInt và thực hiện & từng 2 bit đôi 1.
* Toán tử OR |: Đi từng bit của QInt và thực hiện & từng 2 bit đôi 1.
* Toán tử XOR ^: Đi từng bit của QInt và thực hiện & từng 2 bit đôi 1.
* Toán tử NOT~: Đi từng bit của QInt và thực hiện NOT từng bit một.
* Phép toán dịch trái: Dịch chuyển bit ở vị trí 1 đến bit ở vị trí 127 lần lượt theo thứ tự vào vị trí 0 đến vị trí 126 và cập nhật lại bit ở vị trí 127 là 0. Lặp lại n lần việc làm trên để dịch trái n lần.
* Phép toán dịch phải (dịch số học): Dịch chuyển bit ở vị trí 0 đến bit ở vị trí 126 lần lượt theo thứ tự vào vị trí 1 đến vị trí 127 và cập nhật lại bit ở vị trí 0 là 0 nếu là số dương, 1 nếu là số âm. Lặp lại n lần việc làm trên để dịch phải n lần.
* Phép toán ROL: Dịch chuyển bit ở vị trí 1 đến bit ở vị trí 127 lần lượt theo thứ tự vào vị trí 0 đến vị trí 126 và cập nhật lại bit ở vị trí 127 là bit ở vị trí 0 của dãy bit đầu vào. Lặp lại n lần việc làm trên để xoay trái n lần.
* Phép toán ROR: Dịch chuyển bit ở vị trí 0 đến bit ở vị trí 126 lần lượt theo thứ tự vào vị trí 1 đến vị trí 127 và cập nhật lại bit ở vị trí 0 là bit ở vị trí 127 của dãy bit đầu vào. Lặp lại n lần việc làm trên để xoay phải n lần.
* Phép toán so sánh >: Nếu cùng là 2 số dương hoặc 2 số âm (xét bit đầu để kiểm tra), đi từng bit từ trái sang đến vị trí k sao cho bit tại vị trí k của số thứ nhất khác bit tại vị trí k của số thứ 2, kiểm tra nếu bit tại vị trí k của số thứ nhất là 1 thì trả về 1 ngược lại trả về 0. Nếu 1 số là âm 1 số là dương, kiểm tra số thứ nhất nếu là dương thì trả về 1, ngược lại trả về 0. Nếu không rơi vào các trường hợp trên trả về 0 (trường hợp 2 số bằng nhau).
* Phép toán so sánh ==: Nếu hai số không có bit nào khác nhau thì trả về 1, ngược lại trả về 0.
* Phép toán so sánh <: Nếu số thứ nhất không lớn hơn và không bằng số thứ 2 (sử dụng toán tử >, = đã viết ở trên để hỗ trợ) thì số thứ nhất bé hơn số thứ 2, trả về 1, ngược lại trả về 0.
* Phép toán so sánh >=: Nếu số thứ nhất lớn hơn hoặc bằng số thứ 2 (dùng toán tử > và == để kiểm tra) thì trả về 1, ngược lại trả về 0.
* Phép toán so sánh <=: Nếu số thứ nhất bé hơn hoặc bằng số thứ 2 (dùng toán tử < và == để kiểm tra) thì trả về 1, ngược lại trả về 0.

3. Cấu trúc của kiểu Qfloat

\* Mô tả cấu trúc kiểu dữ liệu QFloat.

* Xây dựng từ cấu trúc dữ liệu bitset có sẵn trong thư viện chuẩn STL của C++.
* Kiểu dữ liệu QFloat được xây dựng chính là một bitset<128> dùng để lưu 128 bit biểu diễn dạng nhị phân của một số chấm động với:
* 1 bit đầu tiên là phần dấu (S).
* 15 bit tiếp theo là phần số mũ (Exponent).
* 112 bit còn lại là phần định trị (Significand).

\* Lí do chọn bitset<128> để biểu diễn **số chấm động có độ chính xác cao.**

* Thỏa mãn yêu cầu đề bài, sizeof của bitset<128> là 16 byte.
* Dễ dàng truy cập đến bit thứ k trong dãy 128 bit, thuận tiện cho việc lập trình các chức năng.
* Hỗ trợ sẵn một số toán tử thông dụng.

### 4. Ý tưởng của các thuật toán thực hiện chức năng chuyển đổi qua lại giữa các hệ cơ số và các hàm trên kiểu Qfloat

**\* Hàm nhập, xuất**

* Hàm nhập yêu cầu nhập vào một string số ở dạng thập phân, rồi lấy string đó truyền vào hàm floatingPoint ở trên để chuyển sang dạng bitset<128> và lưu vào QFloat x.
* Hàm xuất nhận vào tham số là một số chấm động x. Xét từng bit của x và xuất ra ở dạng chấm động. Chuyển về dạng thập phân nhờ vào hàm floatingPointToDec và xuất ra.

**\* Các phép chuyển hệ cơ số**

* bitset<128> floatingPoint(string s): Hàm lấy tham số là một string số ở dạng thập phân. Sau đó kiểm tra kí tự đầu để xét dấu. Sau đó chia string này thành 2 phần nguyên và phần thập phân. Chuyển cả 2 sang dạng nhị phân. Chuyển sang dạng 1.F \* 2^E. Tìm E, sau đó cộng với 2^14 – 1 và chuyển sang nhị phân để lưu vào phần Exp. Sau đó lấy E cuối của dạng nhị phân phần nguyên, ghép với dạng nhị phân của phần thập phân để lưu vào phần Sign của dấu chấm động.
* string floatingPointToDec(QFloat x): Đầu tiên, tạo một biến tạm để lưu các bit của số chấm động x. Sau đó, dùng các biến tạm khác để lưu các phần dấu, Exp, Sign trong dãy bit để dễ quản lý và chuyển đổi. Xét bit dấu để tìm dấu của kết quả trả về. Tính giá trị của phần bit Exp rồi trừ cho 2^14 – 1 để tìm ra số E. Lấy E bit đầu của dãy bit Sign, thêm một bit 1 vào đầu dãy bit vừa có, tính ra dạng thập phân để có được phần nguyên của kết quả. 112 – E bit còn lại trong phần Sign là giá trị của phần thập phân trong kết quả.Tạo một string kết quả thập phân có giá trị là “0”. Ta xét từng bit một. Bên cạnh đó, tạo một string temp có giá trị “1”. Mỗi lần xét bit, ta thêm một số 0 vào temp, chia đôi string đó, xét giá trị của bit, nếu là 1 thì cộng thêm vào phần thập phân giá trị của temp. Lặp lại cho đến khi hết bit. Việc ta tạo một biến temp có giá trị là “1”, thêm số 0 vào temp, rồi chia đôi nó, có ý nghĩa tương tự như số 1 chia đôi được 0.5. Tiếp tục thêm 0 vào temp và chia đôi khi lặp tương tự như 0.50 chia đôi được 0.25. Tương tự như thế lần lượt có được 2^-1, 2^-2,…
* bool \*DecToBin(QFloat x): Tạo một biến bool\* lưu địa chỉ của 128 biến bool. Xét số x, lần lượt xét giá trị của các bit, sau đó lưu giá trị đó vào biến bool[i]. Kết thúc, ta có được biến bool\* lưu giá trị các bit của x.
* QFloat BinToDec2(bool \*bit): Hàm lấy tham số là một biến bool\*. Tạo một bitset<128> temp để lưu tạm giá trị của dãy bit. Xét các giá trị của biến bool\* bit để lưu vào temp. Từ temp, ta lưu dãy bit vào số chấm động x. Kết thúc hàm trả về kết quả x.

## II. PHẠM VI BIỂU DIỄN CỦA CÁC KIỂU DỮ LIỆU ĐÃ THIẾT KẾ

### 1. QInt

* Từ -2127 đến 2127 – 1.

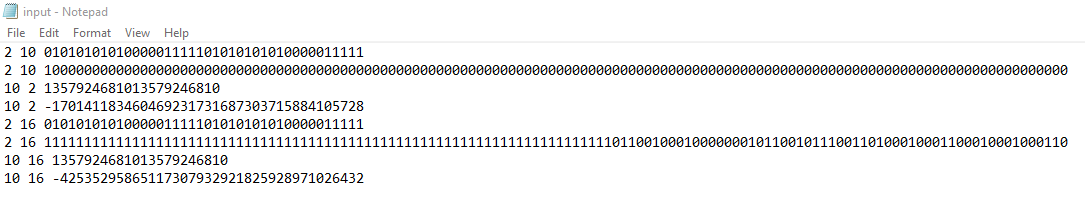
### 2. QFloat

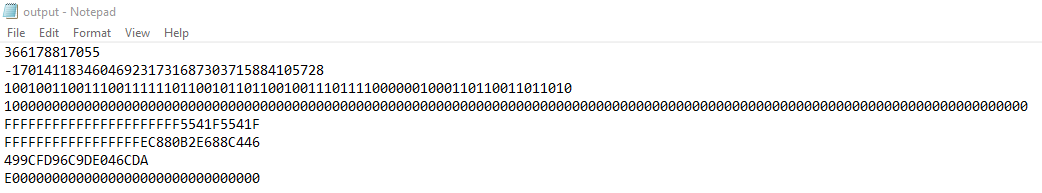
* Tầm hiển thị của số dấu phẩy động phụ thuộc vào số bit hay số chữ số của phần định trị và của phần số mũ. Một số dấu phẩy động nhị phân định dạng ‘QFloat’ (128 bit) có phần định trị 112 bit, phần số mũ gồm 15 bit và 1 bit dấu. Tầm số dương mà định dạng này có thể biểu diễn được là khoảng từ 10−4932 đến 104932 (vì 4932 xấp xỉ bằng 16383 × log10(2) và tầm số mũ của định dạng độ chính xác là [−16383,16384]). Toàn tầm hiển thị của định dạng độ chính xác là khoảng từ −104932 đến +104932.

**III. GIAO DIỆN ỨNG VỚI CÁC BỘ TESTCASE**

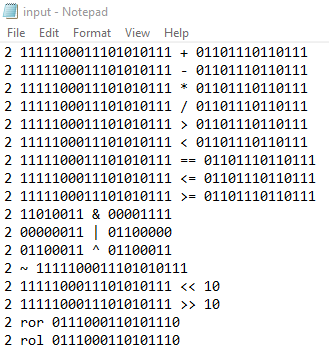
### 1. QInt

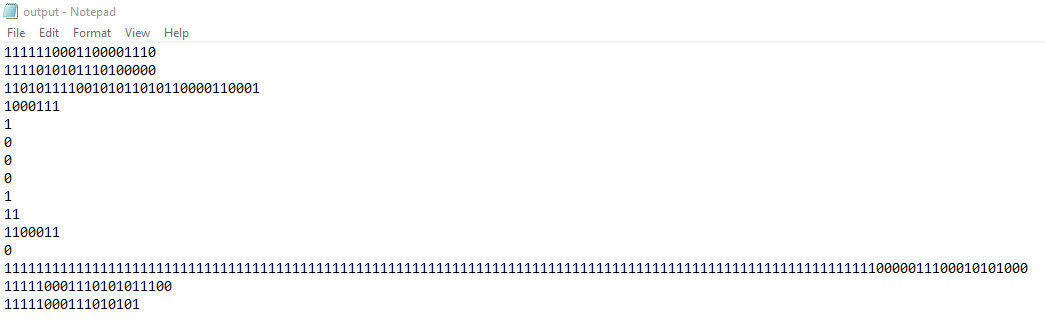
\* Chuyển đổi qua lại giữa các hệ cơ số



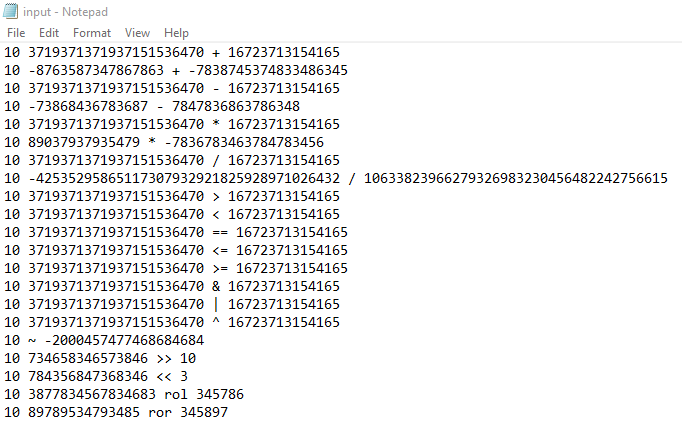


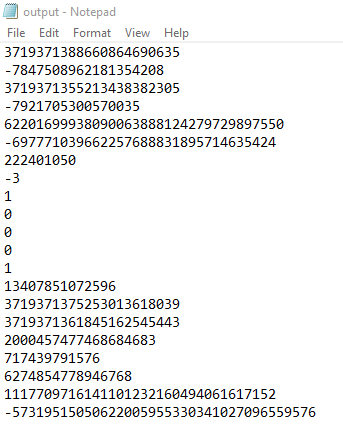
\* Các phép toán tính trên hệ cơ số 2



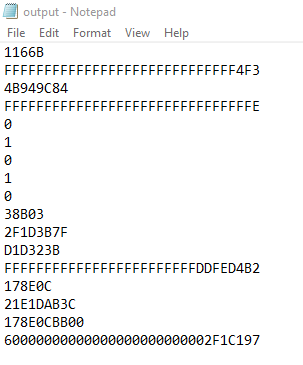
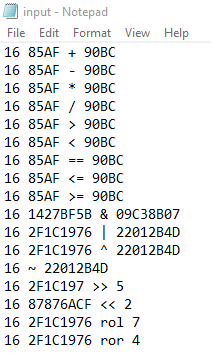


\* Các phép toán tính trên hệ cơ số 10



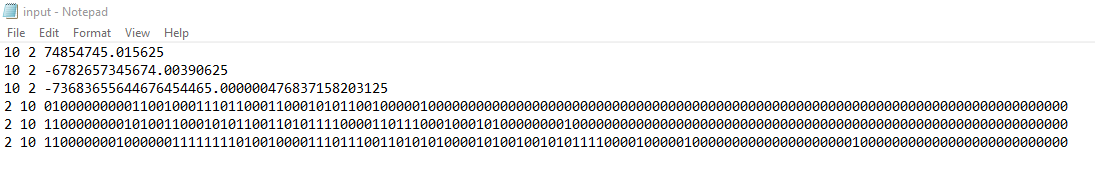


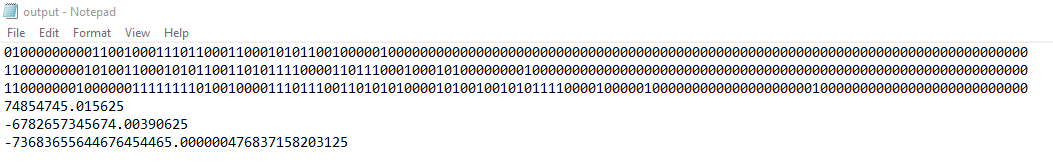
\* Các phép toán tính trên hệ cơ số 16



### 2. QFloat

\* Chuyển đổi qua lại giữa hệ 2 và hệ 10





## IV. ĐÁNH GIÁ MỨC ĐỘ HOÀN THÀNH

### 1. Mức độ hoàn thành ứng với các yêu cầu chi tiết của dự án

* Các hàm nhập xuất, chuyển cơ số, các toán tử: Hoàn thành 100% các hàm theo nguyên mẫu hàm của đề bài, đồng thời có các hàm nạp chồng với chức năng tương tự nhưng tham số và kiểu trả về khác.
* Xử lí được cấu trúc file input theo yêu cầu đề bài và xuất kết quả theo yêu cầu ra file output.
* Báo cáo gồm đầy đủ nội dung được yêu cầu, bố cục rõ ràng.

### 2. Mức độ hoàn thành ứng với toàn bộ dự án: 100%

# DANH MỤC TÀI LIỆU THAM KHẢO

*[1] Phép toán thao tác bit,*

*<https://vi.wikipedia.org/wiki/Ph%C3%A9p\_to%C3%A1n\_thao\_t%C3%A1c\_bit>, xem 12/03/2019.*

*[2] Phạm Tuấn Sơn. Bài giảng học phần Kiến trúc máy tính và hợp ngữ, Trường Đại học Khoa học Tự nhiên TPHCM.*

# BIÊN BẢN PHÂN CÔNG VIỆC NHÓM

## 1. Thông tin chung

Các thành viên có tham gia làm đồ án:

|  |  |  |
| --- | --- | --- |
| **STT** | **MSSV** | **Họ và tên** |
| 1 | 1712737 | Đỗ Tấn Tài |
| 2 | 1712724 | Huỳnh Công Sinh |
| 3 | 1712709 | Trần Thiên Quàng |
| 4 | 1712787 | Nguyễn Văn Thìn |
| 5 | 1712410 | Phan Đại Hải |

## 2. Bảng phân công công việc

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Người phụ trách** | **Mô tả nội dung công việc** | **Kết quả** | **Đánh giá của Nhóm trưởng và thành viên** |
| 1 | Đỗ Tấn Tài | ➀ Code các câu a, b, c, d, e, f, g(\*,/) phần QInt.  ➁ Code phần xử lí input, output và tham số dòng lệnh.  ➂ Làm báo cáo | Hoàn thành 100%, đúng dealine nhóm | Thực hiện tốt vai trò leader nhóm: đề cao giá trị mỗi cá nhân và kết nối các thành viên trong nhóm với tinh thần trách nhiệm cao. |
| 2 | Huỳnh Công Sinh | ➀ Code các câu a, c phần QFloat.  ➁ Hỗ trợ làm báo cáo phần ý tưởng thuật toán QFloat. | Hoàn thành 100%, đúng dealine nhóm | Thường phát hiện ra các ý tưởng xây dựng cấu trúc các kiểu dữ liệu và các hàm chức năng và truyền đạt cho nhóm. |
| 3 | Trần Thiên Quàng | ➀ Code các câu b, d phần QFloat.  ➁ Hỗ trợ làm báo cáo phần phạm vi biểu diễn của QFloat. | Hoàn thành 100%, đúng dealine nhóm | Nhiệt tình trong việc đưa ra các ý kiến xây dựng ý tưởng giải quyết các vấn đề trong đồ án được giao. |
| 4 | Nguyễn Văn Thìn | ➀ Code các câu g(+,-),h phần QInt.  ➁ Chuẩn bị các bộ testcase cho kiểu QInt. | Hoàn thành 100%, đúng dealine nhóm | Hoàn thành tốt các yêu cầu được giao với tinh thần học hỏi cao. |
| 5 | Phan Đại Hải | ➀ Code các câu i, j phần QInt. | Hoàn thành 100%, đúng dealine nhóm | Hoàn thành tốt các yêu cầu được giao với tinh thần học hỏi cao. |