**TRƯỜNG ĐẠI HỌC BÁCH KHOA**

**KHOA CÔNG NGHỆ THÔNG TIN**



**ĐỒ ÁN LẬP TRÌNH TÍNH TOÁN**

**XÂY DỰNG ỨNG DỤNG CHUYỂN ĐỔI CƠ SỐ ĐẾM**

**GVHD: GVC.ThS. TRẦN HỒ THỦY TIÊN**

**SINH VIÊN THỰC HIỆN:**

**Nguyễn Thị Mỹ Lệ LỚP: 22T\_DT5 NHÓM: 22Nh14**

**Huỳnh Thị Thanh Nhàn LỚP: 22T\_DT5 NHÓM: 22Nh14**

**Đà Nẵng 06/2023**

[MỤC LỤC](#_Toc138518890)

MỤC LỤC HÌNH ẢNH  [2](#_Toc138518891)

[LỜI CẢM ƠN 3](#_Toc138518891)

[MỞ ĐẦU….. 4](#_Toc138518892)

[1. TỔNG QUAN ĐỀ TÀI 7](#_Toc138518894)

[\*Ý tưởng 7](#_Toc138518895)

[2. CƠ SỞ LÝ THUYẾT 7](#_Toc138518896)

[\*Mảng 1 chiều 8](#_Toc138518897)

[3. TỔ CHỨC CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN 9](#_Toc138518899)

[3.1. Phát biểu bài toán 9](#_Toc138518900)

[3.2. Cấu trúc dữ liệu 10](#_Toc138518901)

[3.3. Thuật toán 10](#_Toc138518902)

[4.CHƯƠNG TRÌNH VÀ KẾT QUẢ 21](#_Toc138518917)

[4.1. Tổ chức chương trình 21](#_Toc138518918)

[4.2. Ngôn ngữ cài đặt 22](#_Toc138518919)

[4.3. Kết quả 22](#_Toc138518920)

[4.3.1. Giao diện chính của chương trình 22](#_Toc138518921)

[4.3.2. Kết quả thực thi của chương trình 23](#_Toc138518926)

[4.3.3. Nhận xét đánh giá 24](#_Toc138518933)

[5. KẾT LUẬN VÀ HƯỚNG P HÁT TRIỂN 25](#_Toc138518934)

[5.1. Kết luận 25](#_Toc138518935)

[5.2. Hướng phát triển 25](#_Toc138518936)

[TÀI LIỆU THAM KHẢO 26](#_Toc138518937)

[PHỤ LỤC.. 27](#_Toc138518938)

**MỤC LỤC HÌNH ẢNH**

[Hình 1. Hình ảnh mở đầu 4](#_Toc138516830)

[Hình 2. Mô phỏng mảng 1 chiều có kích thước là 5 8](#_Toc138516835)

[Hình 3. Sơ đồ khối hàm so\_am 11](#_Toc138516840)

[Hình 4. Sơ đồ khối hàm thap\_phan 11](#_Toc138516841)

[Hình 5. Sơ đồ khối hàm nhan 12](#_Toc138516842)

[Hình 6. Sơ đồ khối hàm dau\_cach 13](#_Toc138516844)

[Hình 7. Sơ đồ khối hàm ket\_qua 14](#_Toc138516845)

[Hình 8. Sơ đồ khối hàm dao\_bit\_cong\_1 15](#_Toc138516846)

[Hình 9. Sơ đồ khối hàm kiem\_tra 16](#_Toc138516847)

[Hình 10. Sơ đồ khối hàm chuong\_ trinh 20](#_Toc138516851)

[Hình 11. Sơ đồ vibộ chương trình 21](#_Toc138516853)

[Hình 12. Giao diện mở đầu chương trình 23](#_Toc138516859)

[Hình 13. Giao diện khi nhập đúng lựa chọn có trong MENU 23](#_Toc138516860)

[Hình 14. Giao diện khi nhập lựa chọn không có trong MENU 23](#_Toc138516861)

[Hình 15. Giao diện khi nhập số không hợp lệ 23](#_Toc138516862)

[Hình 16. Giao diện khi nhập số hợp lệ 23](#_Toc138516864)

[Hình 17. Giao diện khi lựa chọn Thoát 24](#_Toc138516865)

[Hình 18. Giao diện khi đổi từ số âm sang hệ 2 24](#_Toc138516866)

[Hình 19. Giao diện khi đổi từ số âm sang hệ 8,10,16 24](#_Toc138516867)

[Hình 20. Giao diện khi đổi từ số thập phân 24](#_Toc138516868)

[Hình 21. Giao diện khi đổi từ số âm thập phân 24](#_Toc138516869)

LỜI CẢM ƠN

Đầu tiên, chúng em xin gửi lời cảm ơn chân thành tới ThS Trần Hồ Thủy Tiên đã giúp đỡ chúng em hoàn thành tốt đề tài đồ án Cơ sở này. Trong quá trình hướng dẫn chúng em làm đồ án, cô đã chỉ bảo và hướng dẫn tận tình cho chúng em những lý thuyết, cũng như các kỹ năng trong lập trình, cách giải quyết các vấn đề trong đề tài,…

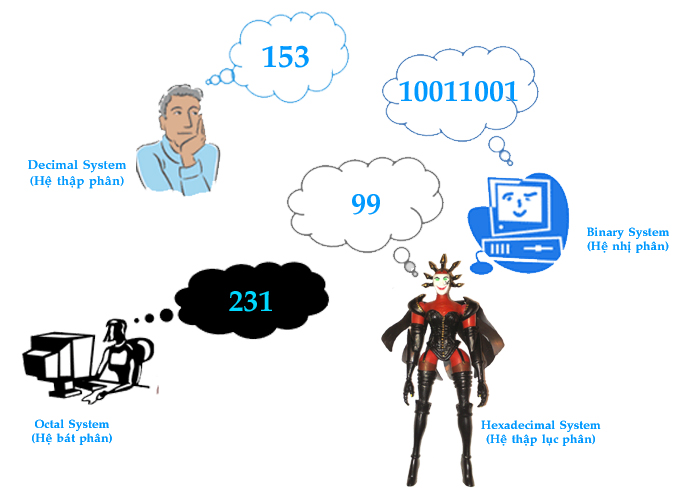
Chân thành cảm ơn các bạn bè trong lớp đã hỗ trợ tôi để hoàn thành tốt đề tài cùng bản báo cáo này.

Mặc dù đã cố gắng hoàn thành đồ án trong phạm vi và khả năng cho phép nhưng chắc chắn sẽ không tránh khỏi những thiếu sót. Chúng em rất mong nhận được sự thông cảm, góp ý và tận tình chỉ bảo của quý Thầy Cô.

**Nguyễn Thị Mỹ Lệ**

**Huỳnh Thị Thanh Nhàn**

MỞ ĐẦU



# Hình 1. Hình ảnh mở đầu

**1. Mục đích thực hiện đề tài** :

-Trong thời điểm hiện nay, các ngành công nghệ thông tin đang ngày càng mở rộng và phát triển , thu hút khá nhiều nguồn nhân lực . Cùng với sự phát triển của nó thì cũng xuất hiện nhiều nhu cầu , khó khăn cần có phương án để giải quyết, trong đó vấn đề về hệ đếm và chuyển đổi giữa các hệ đếm được nhiều người quan tâm. Việc chuyển đổi giữa các hệ đếm xuất hiện thường xuyên trong các dự án, các bài lập trình từ lớn đến nhỏ . Tuy nhiên chúng khá phức tạp đòi hỏi các lập trình viên phải nắm vững kiến thức lí thuyết, vận dụng một cách linh hoạt và nhanh chóng để xây dựng giải thuật rồi mới viết chương trình hoàn chỉnh. Trường gặp những bài chuyển đổi hệ đếm phức tạp và dài dòng thì tốn khá nhiều thời gian ảnh hưởng nghiêm trọng đến các tiến độ lập trình của họ.

-Trong quá trình tìm hiểu và khảo sát nhiều sinh viên, lập trình viên, nhóm em thấy rằng vấn đề chuyển đổi các hệ đếm trong tin học là điều vô cùng cần thiết và hữu ích phù hợp với thời đại hiên nay.Với những lí do trên, nhóm em quyết định lựa chọn đề tài **“Xây dựng ứng dụng chuyển đổi giữa các hệ đếm”** làm đề tài của chính của dự án lần này để giúp mọi người tăng hiệu quả trong học tập , làm việc.

**2. Mục tiêu đề tài :**

**-**Dự án hướng đến mục tiêu xây dựng được một ứng dụng đơn giản, gọn nhẹ và giúp người dùng dễ dàng chuyển đổi các hệ đếm một cách nhanh, chính xác, đáp ứng nhu cầu công việc , học tập của mọi người.

**3. Phạm vi dự án :**

-Ứng dụng được sử dụng với phạm vi là ở trường đại học, các công ty kĩ thuật.

**4. Đối tượng nghiên cứu**:

- Dự án được triển khai nghiên cứu phù hợp với nhiều người.

- Các hệ đếm và cách chuyển đổi giữa các hệ đếm.

- Ngôn ngữ lập trình C

- Các học sinh và sinh viên học toán học hoặc theo học các ngành công nghệ và kĩ thuật, các lập trình viên , …

**5. Phương pháp nghiên cứu:**

- Phương pháp phân tích, xây dựng cơ sở lý thuyết.

- Thu thập thông tin trên Internet, kiến thức và thực tế trong học tập.

- Sử dụng phần mềm Dev-C++ để viết chương trình.

**6.Cấu trúc đồ án môn học:**

**I. MỞ ĐẦU**

1. Mục đích thực hiện đề tài :

2. Mục tiêu thực hiện đề tài

3. Phạm vi :

4. Đối tượng nghiên cứu:

5. Phương pháp nghiên cứu:

6. Cấu trúc đồ án môn học

**II.PHẦN THÂN**

1. Tổng quan đề tài

2. Cơ sở lý thuyết

2.1.Ý tưởng

2.2. Cơ sở lí thuyết

3. Tổ chức cáu trúc dữ liệu và thuật toán

3.1. Phát biểu bài toán

3.2. Cấu trúc dữ liệu

3.3. Thuật toán

4. Chương trình và kết quả

4.1. Tổ chức chương trình

4.2. Ngôn ngữ cài đặt

4.3. Kết quả

4.3.1. Giao diện chính của chương trình

4.3.2. Kết quả thực thi của chương trình

4.3.3. Nhận xét đánh giá

5. Kết quả và hướng phát triển

5.1. Kết luận

5.2. Hướng phát triển

Tài liệu tham khảo.

Phụ lục

# TỔNG QUAN ĐỀ TÀI

* Hệ thống cơ số là cách thức biểu diễn số trong máy tính. Việc chuyển đổi giữa các hệ cơ số là một kỹ năng quan trọng trong lĩnh vực toán học và khoa học máy tính.
* Với mục tiêu của đề tài là phát triển một ứng dụng để giúp người dùng chuyển đổi giữa các hệ thống cơ số phổ biến, bao gồm hệ thống thập phân, nhị phân, bát phân và thập lục phân.
* Đề tài lần này nghiên cứu dành cho những người sử dụng máy tính và lập trình viên có nhu cầu sử dụng các hệ thống cơ học khác nhau để thực hiện các phép tính số học và tính toán.
* Các tính năng chính của ứng dụng bao gồm: chuyển đổi giữa các hệ thống cơ số biến phổ biến, tính năng hiển thị kết quả chính xác, rõ ràng .  Sản phẩm được thiết kế với giao diện thân thiện và đơn giản để cung cấp cho người dùng trải nghiệm sử dụng tiện lợi và dễ dàng.

## \*Ý tưởng

* Sử dụng các phép toán trong môn kĩ thuật lập trình đã học .
* Sử dụng các kiến thức về chuỗi kí tự, mảng, con trỏ, hướng đối tượng.
* Sử dụng các kiến thức đã học về cách đổi các hệ đếm.

# CƠ SỞ LÝ THUYẾT

- Hệ cơ số là cách thức biểu diễn số học trong máy tính, các hệ cơ số phổ biến bao gồm hệ thập phân, nhị phân, bát phân và thập lục phân. Cách thức biểu diễn số học trong mỗi hệ cơ số khác nhau tùy thuộc vào số lượng các ký tự được sử dụng để biểu diễn các giá trị số và quy tắc cộng trừ trong hệ cơ số đó.

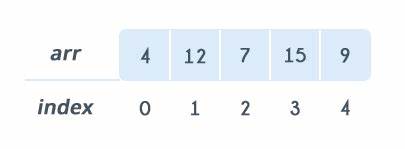
- Các thuật toán chuyển đổi giữa các hệ cơ số là rất quan trọng để thực hiện các phép chuyển đổi tính toán học từ một hệ cơ số sang hệ cơ số khác. Các thuật toán chuyển đổi bao gồm:

* Chuyển đổi từ hệ thống phân tích sang hệ thống cơ số khác: phương pháp chia lấy dư
* Chuyển đổi từ hệ nhị phân sang hệ cơ số khác: phương pháp chia 2 và lấy số dư
* Chuyển đổi từ hệ bát phân sang hệ cơ số khác: phương pháp chia lấy dư và lấy giá trị tương ứng với các ký tự bát phân
* Chuyển đổi từ hệ thống thập lục phân sang hệ cơ số khác: phương pháp chia lấy dư và lấy giá trị tương ứng với các ký tự thập lục phân.

Ngoài ra, đề tài cũng cần phải tìm hiểu về các thuật toán tính toán cơ bản như phép cộng, phép trừ, phép nhân và phép chia trong các hệ thống cơ số khác nhau.

## \*Mảng 1 chiều

* Mảng một chiều là tập hợp tuần tự các phần tử có cùng kiểu dữ liệu và các phần tử được lưu trữ trong một dãy các ô nhớ liên tục trên bộ nhớ. Mỗi phần tử của mảng một chiều được truy xuất bởi tên mảng và vị trí trong mảng gọi là “ chỉ mục ”, được đặt trong cặp dấu ngoặc vuông [ ]. Vị trí trong mảng luôn bắt đầu từ 0 đến tổng số phần tử trừ đi 1.
* Cách truy cập phần tử đầu tiên : a[0]
* Ví dụ với kích thước là 5, tương ứng với 5 phần tử.



# Hình 2. Mô phỏng mảng 1 chiều có kích thước là 5

* Ưu điểm của mảng 1 chiều:

+ Dễ hiểu và dễ sử dụng: chỉ cần khai báo <kiểu dữ liệu> tên mảng[kích thước].

+ Truy cập đến các phần tử trong mảng nhanh: chúng ta có thể truy cập tới bất kì phần tử nào trong mảng bằng cách chỉ định chỉ số cho phần tử đó. Ví dụ: mảng A[100] gồm 100 phân tử ( từ A[0] đến A[99]), để truy xuất tới phần tử thứ i ta chỉ cần gọi giá trị A[i]. Thời gian truy cập phần tử A[0] và thời gian truy cập phần tử A[1000] là như nhau.

* Nhược điểm của mảng 1 chiều:

+ Kích thước của mảng phải là cố định: Trong cấp phát mảng tĩnh, mảng cần được khai báo với kích thước xác định trước khi chạy chương trình. (vùng nhớ cho mảng được cấp phát khi biên dịch). Trong cấp phát động, vùng nhớ được cấp phát khi chạy chương trình. Như các bạn đã biết, vùng nhớ giành cho mỗi chương trình thường không dự đoán được trước. Nếu khai báo mảng với kích thước lớn, không sử dụng hết sẽ gây lãng phí bộ nhớ, ngược lại nếu kích thước vùng nhớ không đủ dùng, chúng ta không thể mở rộng vùng nhớ thêm được, dẫn đến buffer overrun ( tràn vùng nhớ).

+Việc chèn và xóa phần tử của mảng mất nhiều thời gian: vì vùng nhớ cấp phát cho mảng được sắp xếp liên tục nên việc chèn một phần tử mới vào hoặc xóa phần tử trong mảng trở lên khó khăn. Ví dụ: cho mảng A[100], chúng ta muốn chèn thêm phần tử mới vào vị trí i, tất cả các phần tử thứ i trở đi phải dịch sang vị trí kế tiếp để chèn giá trị vào vị trí thứ i. Việc xóa phần tử trong mảng cũng tương tự như vậy, dịch tất cả các phần tử từ vị trí thứ i+1 sang vị trí liền trước nó.

# TỔ CHỨC CẤU TRÚC DỮ LIỆU VÀ THUẬT TOÁN

## 3.1. Phát biểu bài toán

-Sử dụng các hàm xử lý chuỗi.

-Nhiều vấn đề cần giải quyết với các biến, vòng lặp và ra quyết định.

-Thuật toán và cấu trúc dữ liệu.

-Viết chương trình bằng C/C++ để giải quyết bài toán.

-Viết các hàm để xây dựng các chức năng.

Chương trình gồm menu các chức năng sau:

1 - Đổi từ hệ thập phân sang hệ nhị phân.

2 - Đổi từ hệ thập phân sang hệ bát phân.

3 - Đổi từ hệ thập phân sang hệ thập lục phân.

4 - Đổi từ hệ nhị phân sang hệ thập phân.

5 - Đổi từ hệ bát phân sang hệ thập phân.

6 - Đổi từ hệ thập lục phân sang hệ thập phân.

7 - Đổi từ hệ nhị phân sang hệ thập lục phân.

8 - Đổi từ hệ nhị phân sang hệ bát phân.

9 - Đổi từ hệ bát phân sang hệ nhi phân.

10 - Đổi từ hệ thập lục phân sang hệ nhị phân.

11 - Thoát

## 3.2. Cấu trúc dữ liệu

char b[] = ”0123456789ABCDEF”

- Mảng b[] là mảng toàn cục có độ dài 16 kí tự.

- Từ b[0] đến b[9] tương ứng với các kí tự ‘0’ đến ‘9’.

- Từ b[10] đến b[15] tương ứng với các kí tự ‘A’ đến ‘F’.

## 3.3. Thuật toán

3.3.1. Thuật toán chính của chương trình

-Bước 1: Nhập chuỗi c, sau đó lưu chuỗi c qua a (nếu có dấu ‘-‘ phía trước thì làm mất dấu ‘-‘).

-Bước 2: Đổi chuỗi a thành số thực ở hệ 10.

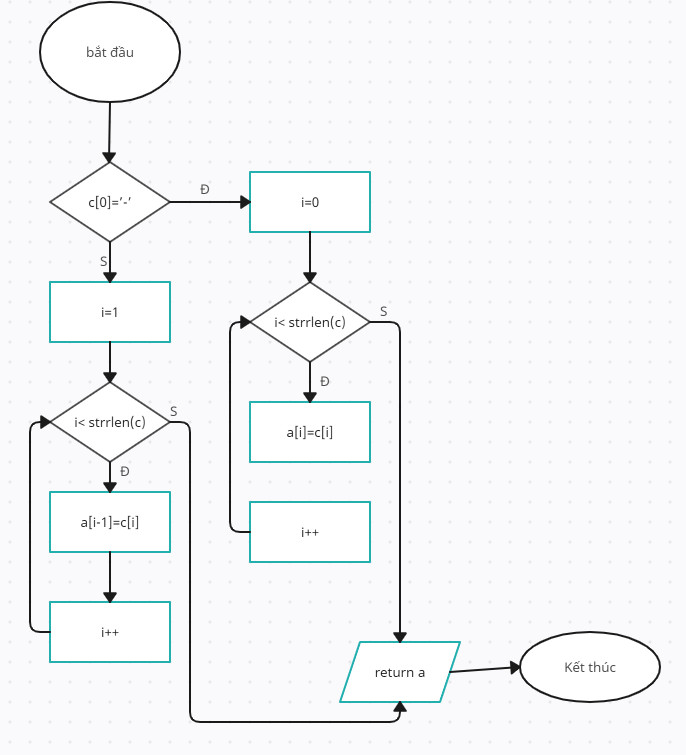
-Bước 3: + Chia số vừa đổi cho cơ số muốn đổi bằng cách chia lấy dư.

+ Đối với số âm đổi sang hệ 2 thì phải đảo bit cộng 1.

+ Có hàm kiểm tra dấu phẩy đối với số thập phân sau đó tách phần nguyên ra chia lấy dư như bình thường, còn phần thập phân thì nhân cho cơ số cần chuyển lấy nguyên (sau mỗi lần nhân trừ đi cho 1).

-Bước 4: Xuất kết quả ra màn hình.

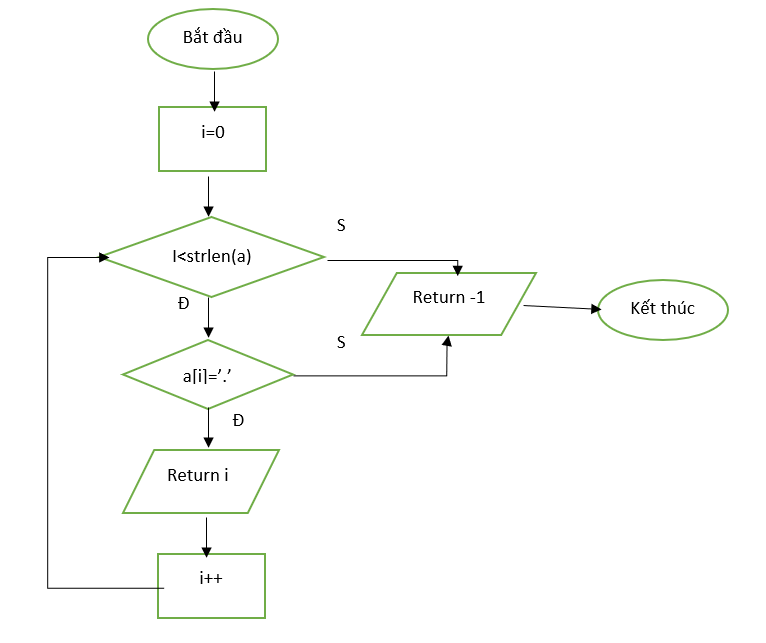
3.3.2. Thuật toán của hàm so\_am



# Hình 3. Sơ đồ khối hàm so\_am

* Độ phức tạp của thuật toán O(n).

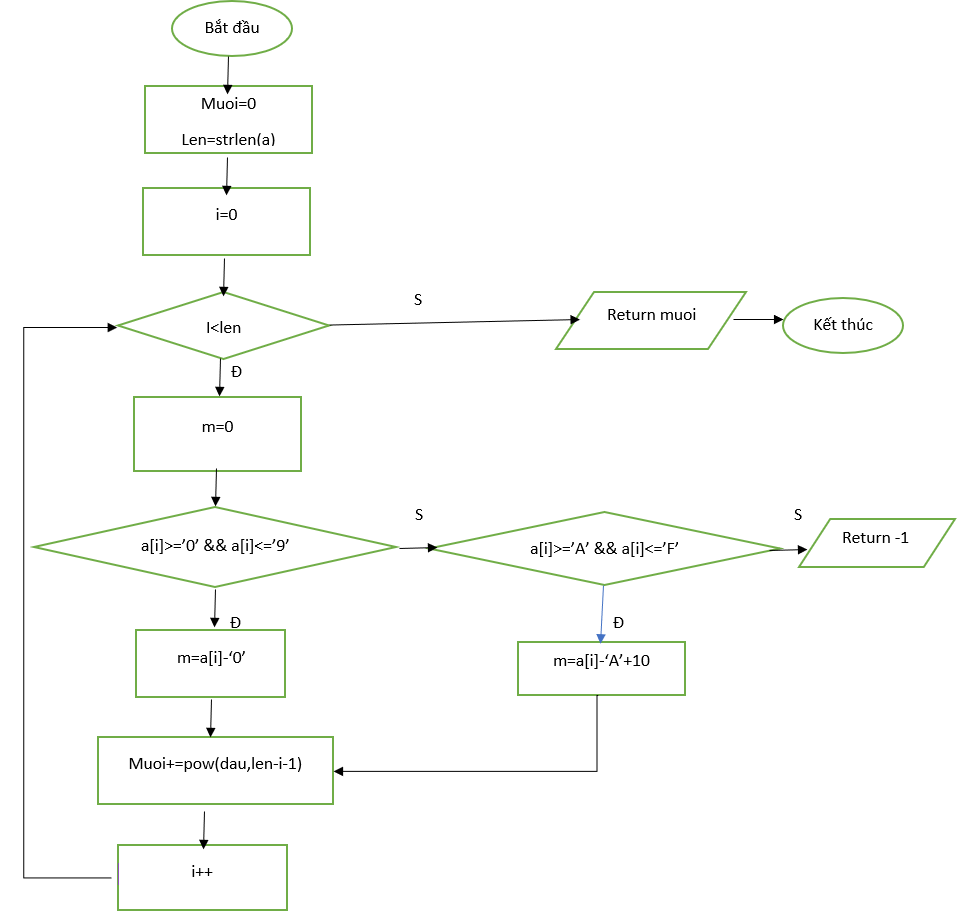
3.3.3. Thuật toán hàm thap\_phan



# Hình 4. Sơ đồ khối hàm thap\_phan

* Độ phức tạp của thuật toán O(n).

3.3.4. Thuật toán hàm nhan



# Hình 5. Sơ đồ khối hàm nhan

* Độ phức tạp của thuật toán O(n).

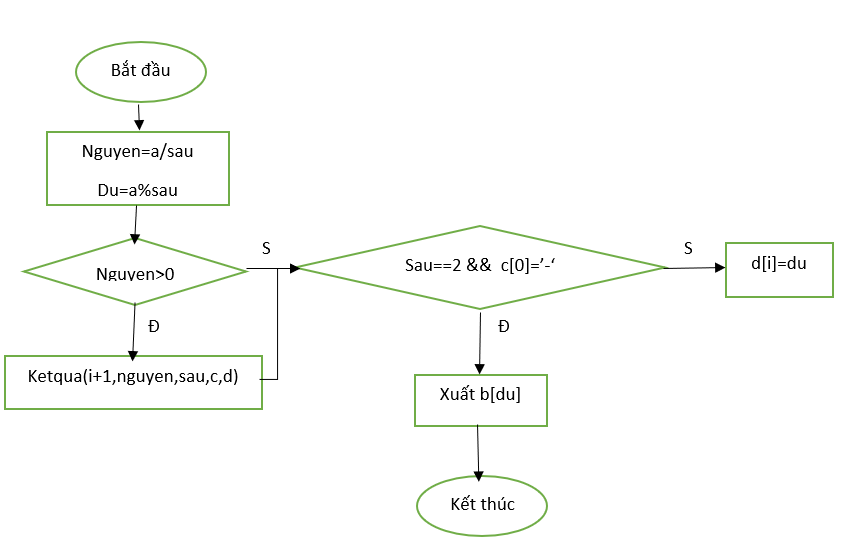
3.3.5. Thuật toán hàm dau\_cach

## 

# Hình 6. Sơ đồ khối hàm dau\_cach

* Độ phức tạp của thuật toán O(1).

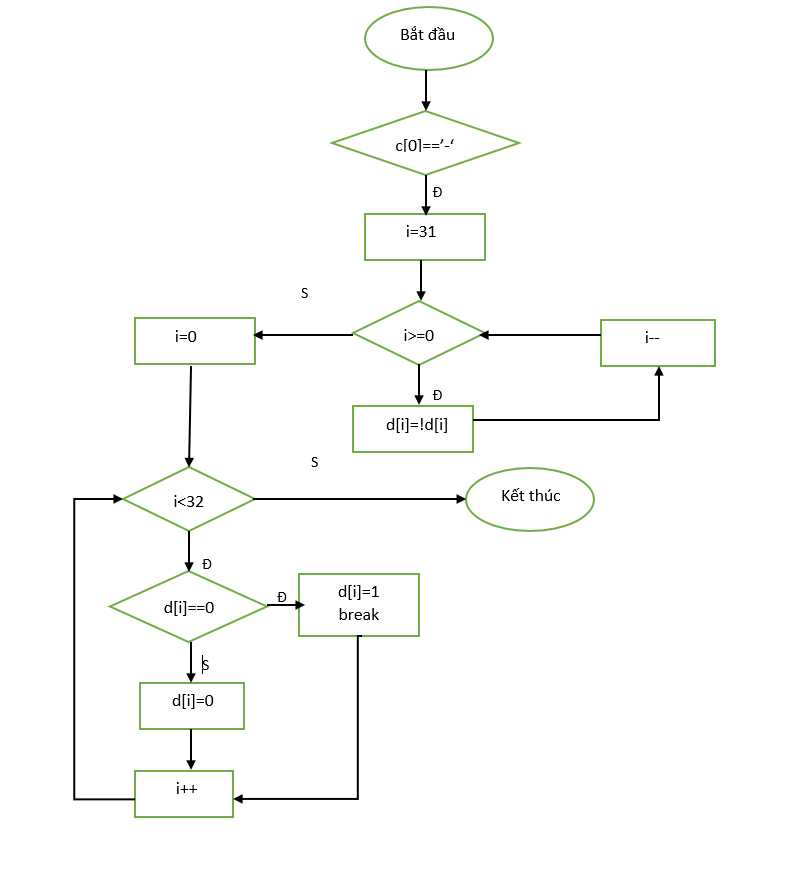
3.3.6. Thuật toán hàm ket\_qua



# Hình 7. Sơ đồ khối hàm ket\_qua

* Độ phức tạp của thuật toán O(log a).

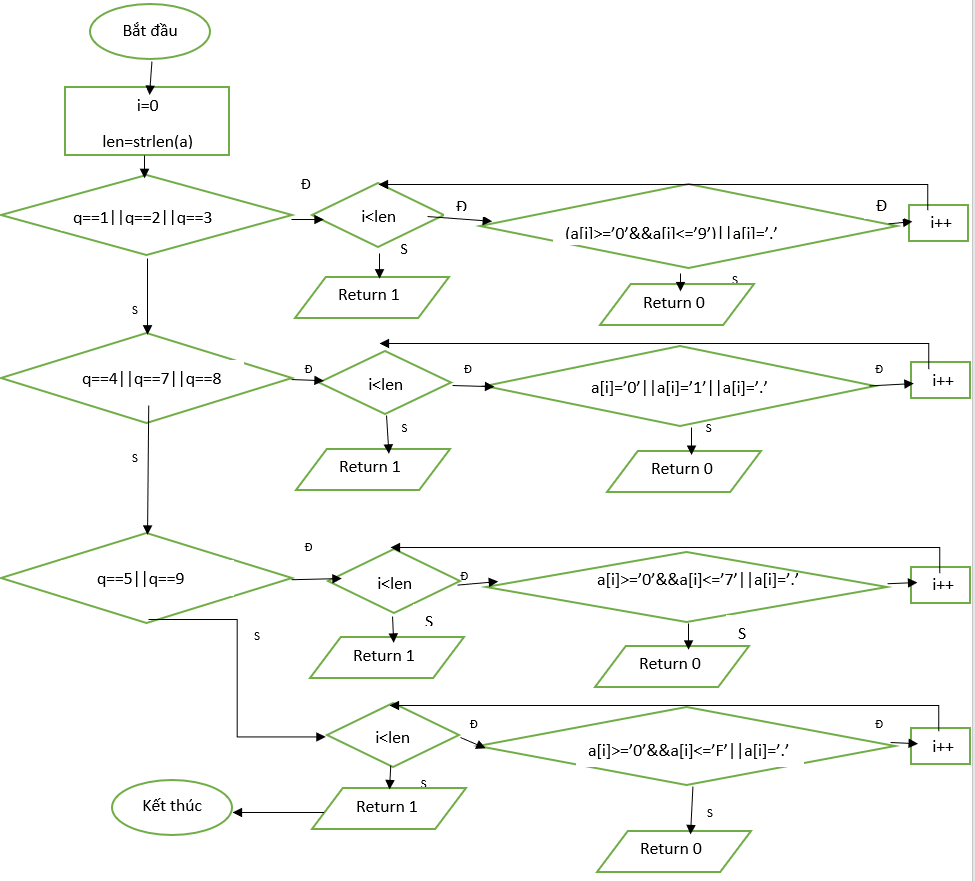
3.3.7. Thuật toán ham dao\_bit\_cong\_1



# Hình 8. Sơ đồ khối hàm dao\_bit\_cong\_1

* Độ phức tạp của thuật toán O(1).

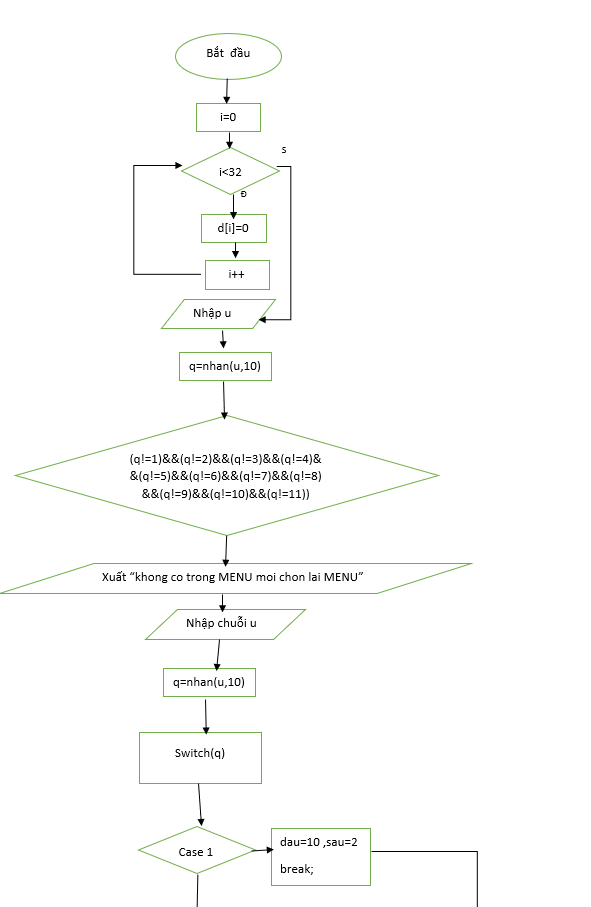
3.3.8. Thuật toán hàm kiem\_tra

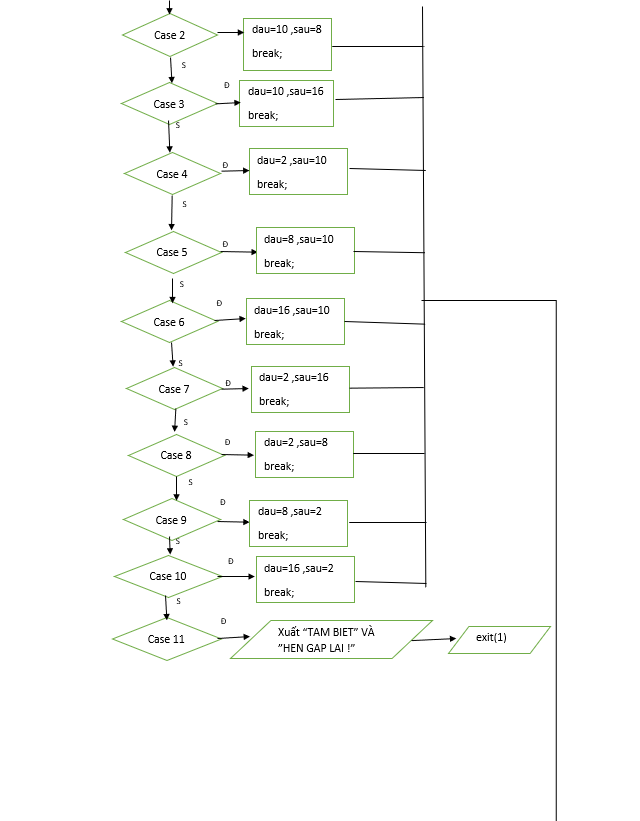


# Hình 9. Sơ đồ khối hàm kiem\_tra

* Độ phức tạp của thuật toán O(n).

3.3.9. Thuật toán hàm chuong\_trinh





### 

### 

### 

# Hình 10. Sơ đồ khối hàm chuong\_ trinh

- Độ phức tạp của thuật toán O(n).

3.3.10. Thuật toán hàm main

### 

# Hình 11. Sơ đồ vibộ chương trình

# 4.CHƯƠNG TRÌNH VÀ KẾT QUẢ

## 4.1. Tổ chức chương trình

#include <stdio.h>

#include <conio.h>

#include <string.h>

#include <math.h>

#include <stdlib.h>

char b[] = "0123456789ABCDEF";

Hàm in ra phần giới thiệu:

*void title();*

Hàm kiểm tra chuỗi nhập vào có thỏa mãn hệ ban đầu:

*int kiem\_tra(char \*a, int q);*

Hàm để chuyển chuỗi kí tự có dấu ‘ – ‘ thành không dấu:

*char\* so\_am(char \*a, char \*c);*

Hàm kiểm tra có phải là số thập phân ( số chứa dấu phẩy ):

*int thap\_phan(char \*a);*

Hàm trả về giá trị của số nhập vào theo hệ 10 :

*unsigned long long nhan(char \*a, int dau);*

Hàm in dấu cách sau mỗi 4 bit đối với hệ 2:

*void dau\_cach(d);*

Hàm in kết quả đối với các số chuyển thành hệ 8,10,16 và lưu vào mảng d[] đối với hệ 2( trước khi đảo bit cộng 1):

*void ket\_qua(int i,int a, int sau , int d[]);*

Hàm dùng để đảo bit cộng 1:

*void dao\_bit\_cong\_1(int d[]);*

Hàm in ra MENU :

*void menu();*

Hàm chính giải quyết chương trình bao gồm số thập phân:

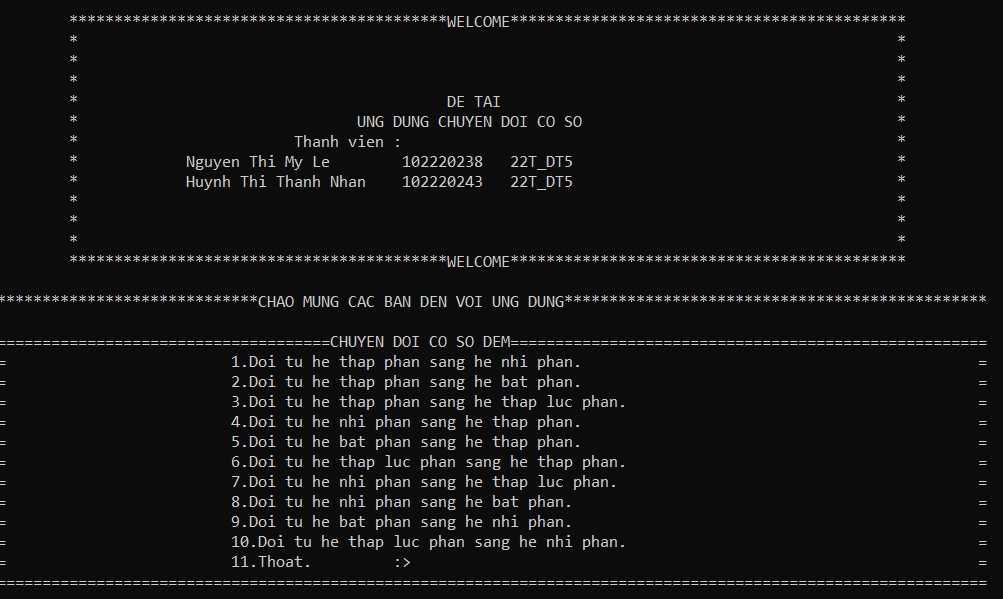
*Void chuong\_trinh();*

## 4.2. Ngôn ngữ cài đặt

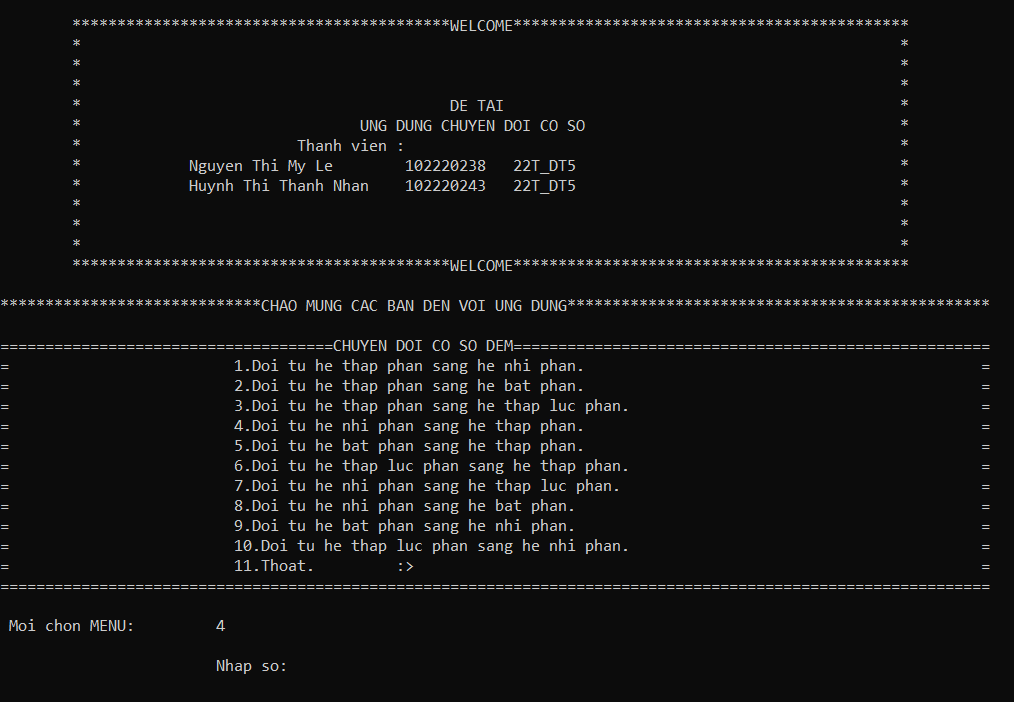
* Ngôn ngữ lập trình C.

## 4.3. Kết quả

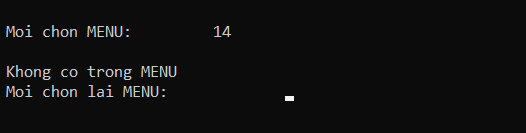
### 4.3.1. Giao diện chính của chương trình



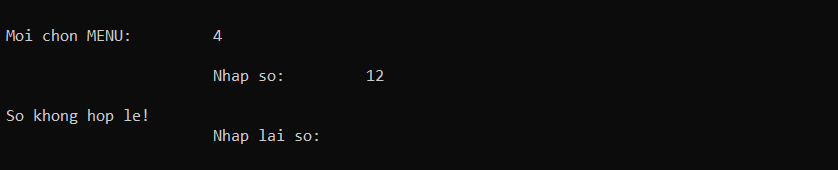
# Hình 12. Giao diện mở đầu chương trình



# Hình 13. Giao diện khi nhập đúng lựa chọn có trong MENU

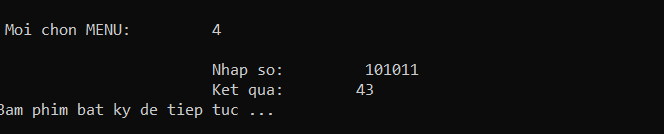


# Hình 14. Giao diện khi nhập lựa chọn không có trong MENU

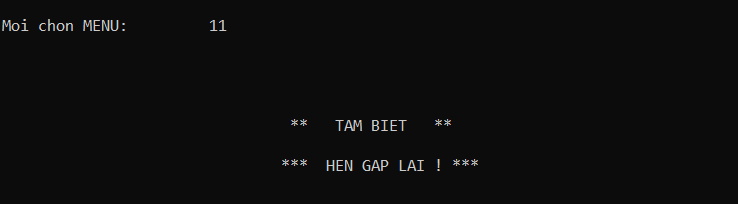


# Hình 15. Giao diện khi nhập số không hợp lệ

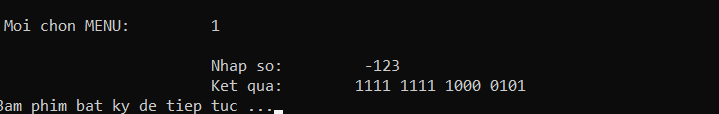
### 4.3.2. Kết quả thực thi của chương trình



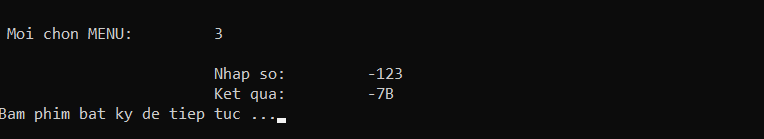
# Hình 16. Giao diện khi nhập số hợp lệ



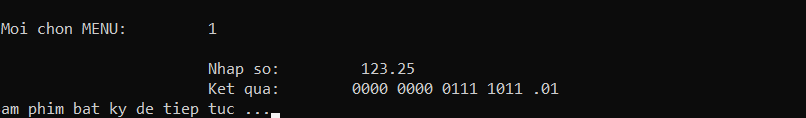
# Hình 17. Giao diện khi lựa chọn Thoát



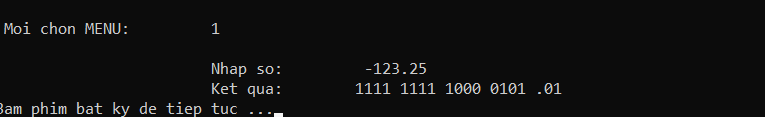
# Hình 18. Giao diện khi đổi từ số âm sang hệ 2



# Hình 19. Giao diện khi đổi từ số âm sang hệ 8,10,16



# Hình 20. Giao diện khi đổi từ số thập phân



# Hình 21. Giao diện khi đổi từ số âm thập phân

### 4.3.3. Nhận xét đánh giá

- Hoàn thành đồ án lập trình tính toán này đã giúp chúng em hiểu rõ hơn về cách xây dựng một chương trình ứng dụng hoàn thiện, cũng như các thao tác với mảng 1 chiều, các thao tác xử lí với chuỗi ,hiểu được cách vẽ sơ đồ khối của thuật toán cũng như cách tổ chức chương trình gọn gàng, ngăn nắp, logic và giúp chúng em hoàn thiện hơn khả năng làm việc nhóm của mình. Dù đã gặp nhiều khó khăn cũng như có nhiều thứ vẫn khá mới lạ vì đây là lần đầu tiên chúng em làm một đồ án lập trình tính toán, nhưng nhờ sự giúp đỡ tận tình, chu đáo của cô Trần Hồ Thủy Tiên đã giúp chúng em hoàn thành đồ án lần này.

# 5. KẾT LUẬN VÀ HƯỚNG P HÁT TRIỂN

## 5.1. Kết luận

- Chương trình của chúng em đã hoạt động theo đúng yêu cầu và ý tưởng của đề tài. Chương trình có phần yêu cầu nhập lại lựa chọn, số mỗi khi nhập sai, có phần chạy chương trình liên tục, phần đổi các số âm hay số thập phân ,...

## 5.2. Hướng phát triển

-Tiếp tục sửa những lỗi còn tồn tại trong chương trình (nếu có).

-Phát triển ứng dụng có sử dụng phần đồ họa để chương trình trở nên thân thiện và dễ dàng sử dụng hơn cho người dùng.

-Phát triển thêm nhiều tính năng nữa như chuyển số có độ dài lớn, đổi số âm hệ 2 qua các hệ, …

TÀI LIỆU THAM KHẢO

Tài liệu lập trình C: https://cuuduongthancong.com/s/lap-trinh-c

Cách chuyển đổi các hệ đếm:

<https://cauvonghoala.blogspot.com/2015/03/caccach-chuyen-oi-giua-cac-he-co-so.html>

Lý thuyết về chuỗi kí tự trong C : <https://cachhoc.net/2014/12/16/lap-trinh-c-bai-8-chuoi-ky-tu-trong-c/>

Lý thuyết về mảng trong C:https://quantrimang.com/hoc/mang-array-trong-cplusplus-156212

PHỤ LỤC

|  |
| --- |
| *#include <stdio.h>*  *#include <conio.h>*  *#include <string.h>*  *#include <math.h>*  *#include <stdlib.h>*  *char b[] = "0123456789ABCDEF";*  *void title();*  *void menu();*  *void chuong\_trinh();*  *int kiem\_tra(char \*a,int q);*  *char\* so\_am(char \*a, char \*c);*  *unsigned long long nhan(char \*a, int dau);*  *void ket\_qua(int i,int a, int sau ,int d[]);*  *void he\_hai(int d[],char \*c);*  *int thap\_phan(char \*a);*  *void dau\_cach(int d[]);*  *int main() {*  *char tt;*  *title();*  *while(1){*  *menu();*  *chuong\_trinh();*  *}}*  *void title()*  *{*  *printf("\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*WELCOME\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");*  *printf("\n\t\* \*");*  *printf("\n\t\* \*");*  *printf("\n\t\* \*");*  *printf("\n\t\* DE TAI \*");*  *printf("\n\t\* UNG DUNG CHUYEN DOI CO SO \*");*  *printf("\n\t\* Thanh vien : \*");*  *printf("\n\t\* Nguyen Thi My Le 102220238 22T\_DT5 \*");*  *printf("\n\t\* Huynh Thi Thanh Nhan 102220243 22T\_DT5 \*");*  *printf("\n\t\* \*");*  *printf("\n\t\* \*");*  *printf("\n\t\* \*");*  *printf("\n\t\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*WELCOME\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*");*  *}*  *void menu()*  *{*  *printf("\n\n\*\*\*\*\*\*\*CHAO MUNG CAC BAN DEN VOI UNG DUNG\*\*\*\*\*\*\*\n\n");*  *printf("================CHUYEN DOI CO SO DEM================\n");*  *printf("=\t\t 1. Doi tu he thap phan sang he nhi phan. =\n");*  *printf("=\t\t 2. Doi tu he thap phan sang he bat phan. =\n");*  *printf("=\t\t 3. Doi tu he thap phan sang he thap luc phan. =\n");*  *printf("=\t\t 4. Doi tu he nhi phan sang he thap phan. =\n");*  *printf("=\t\t 5. Doi tu he bat phan sang he thap phan. =\n");*  *printf("=\t\t 6. Doi tu he thap luc phan sang he thap phan. =\n");*  *printf("=\t\t 7. Doi tu he nhi phan sang he thap luc phan. =\n");*  *printf("=\t\t 8. Doi tu he nhi phan sang he bat phan. =\n");*  *printf("=\t\t 9. Doi tu he bat phan sang he nhi phan. =\n");*  *printf("=\t\t 10. Doi tu he thap luc phan sang he nhi phan. =\n");*  *printf("=\t\t 11. Thoat. :> =\n");*  *printf("=====================================================\n");*  *}*  *int kiem\_tra(char \*a,int q)*  *{ int i=0;*  *int len = strlen(a);*  *if((q==1)||(q==2)||(q==3)){*  *while(i<len)*  *{*  *if ((a[i] >= '0' && a[i] <= '9')|| a[i]=='.') i++;*  *else return 0;*  *} return 1;*  *}*  *else if((q==4)||(q==7)||(q==8)){*  *while(i<len)*  *{*  *if ((a[i] == '0' || a[i] == '1')||a[i]=='.') i++;*  *else return 0;*  *} return 1;*  *}*  *else if((q==5)||(q==9) ){*  *while(i<len)*  *{*  *if ((a[i] >= '0' && a[i] <= '7')|| a[i]=='.') i++;*  *else return 0;*  *} return 1;*  *}*  *else {*  *while(i<len)*  *{*  *if ((a[i] >= '0' && a[i] <= 'F')|| a[i]=='.') i++;*  *else return 0;*  *} return 1;*  *}*  *return 0;}*  *char\* so\_am(char \*a, char \*c)*  *{*  *if(c[0]=='-')*  *{*  *for(int i=1;i<strlen(c);i++) a[i-1]=c[i];*  *}*  *else*  *{*  *for(int i=0;i<strlen(c);i++) a[i]=c[i];*  *}*  *}*  *int thap\_phan(char \*a)*  *{*  *for (int i = 0; i < strlen(a); i++) {*  *if (a[i] == '.') return i*  *} return -1;}*  *unsigned long long nhan(char \*a, int dau)*  *{*  *unsigned long long muoi = 0;*  *int len = strlen(a);*  *for (int i = 0; i < len; i++) {*  *int m = 0;*  *if (a[i] >= '0' && a[i] <= '9') m = a[i] - '0';*  *else if (a[i] >= 'A' && a[i] <= 'F') m = a[i] - 'A' + 10;*  *else return -1;*  *muoi += m \* pow(dau , len - i - 1);*  *}return muoi;*  *}*  *void dau\_cach(int d[])*  *{*  *for(int i=31;i>=0;i--)*  *{*  *printf("%d", d[i]);*  *if (i%4==0) printf(" ");*  *}*  *}*  *void ket\_qua(int i,int a, int sau , int d[])*  *{*  *int nguyen = a / sau;*  *int du = a % sau;*  *if (nguyen > 0) ketqua(i+1,nguyen, sau,d);*  *if(sau==2) d[i]=du;*  *else printf("%c", b[du]);*  *}*  *void dao\_bit\_cong\_1(int d[],char \*c)*  *{*  *if(c[0]=='-')*  *{*  *for(int i=31 ;i>=0;i--) d[i]=!d[i];*  *for(int i=0;i<32;i++)*  *{*  *if(d[i]==0)*  *{ d[i]=1;*  *break;*  *}*  *else d[i]=0;*  *}}}*  *void chuong\_trinh()*  *{*  *int q,dau,sau;*  *char a[50],u[50];*  *int d[32];*  *unsigned long long muoi;*  *for(int i=0;i<31;i++) d[i]=0;*  *printf("\n Moi chon MENU:\t\t");*  *memset(u,0,sizeof(u));*  *scanf("%s",&u);*  *q=nhan(u,10);*  *while((q!=1)&&(q!=2)&&(q!=3)&&(q!=4)&&(q!=5)&&(q!=6)&&(q!=7)&&(q!=8)&&(q!=9)&&(q!=10)&&(q!=11))*  *{*  *printf("\n Khong co trong MENU \n Moi chon lai MENU:\t\t");*  *memset(u,0,sizeof(u));*  *scanf("%s",&u);*  *q=nhan(u,10);*  *}*  *switch(q)*  *{*  *case 1 : dau=10 , sau=2 ; break;*  *case 2 : dau=10 , sau=8 ; break;*  *case 3 : dau=10 , sau=16 ; break;*  *case 4 : dau=2 , sau=10 ; break;*  *case 5 : dau=8 , sau=10 ; break;*  *case 6 : dau=16 , sau=10 ; break;*  *case 7 : dau=2 , sau=16 ; break;*  *case 8 : dau=2 , sau=8 ; break;*  *case 9 : dau=8 , sau=2 ; break;*  *case 10: dau=16 , sau=2 ; break;*  *case 11:*  *{*  *system ("color 0");*  *printf("\n\n\n\n\t\t\t\t \*\* TAM BIET \*\* \n\n");*  *printf("\t\t\t\t\*\*\* HEN GAP LAI ! \*\*\* \n");*  *exit(1); }*  *}*  *printf("\n\t\t\tNhap so:\t ");*  *char c[50];*  *memset(c,0,sizeof(c));*  *scanf("%s", c);*  *memset(a,0,sizeof(a));*  *so\_am(a,c);*  *while (!kiem\_tra(a,q))*  *{*  *printf("\n So khong hop le! \n\t\t\tNhap lai so:\t");*  *memset(c,0,sizeof(c));*  *scanf("%s", c);*  *memset(a,0,sizeof(a));*  *so\_am(a,c);*  *}*  *int o=thap\_phan(a);*  *if(o>0)*  *{ int s1=0;*  *float s2=0;*  *for (int k = 0; k < o; k++)*  *{*  *s1 += (a[k] - '0') \* pow(dau, o - k - 1);*  *}*  *for (int l = o + 1; l < strlen(a ); l++)*  *{*  *s2 += (a[l] - '0') \* pow(dau, o - l);*  *}*  *if((sau==2))*  *{*  *printf("\t\t\tKet qua: \t");*  *ket\_qua(0,s1,sau,d);*  *dao\_bit\_cong\_1(d,c);*  *dau\_cach(d);*  *}*  *else if((sau!=2)&&(c[0]=='-'))*  *{*  *printf("\t\t\tKet qua: \t -");*  *ket\_qua(1,s1,sau,d);*  *}*  *else*  *{*  *printf("\t\t\tKet qua: \t");*  *ket\_qua(1,s1,sau,d);*  *}*  *printf(".");*  *int m=1;*  *do*  *{*  *s2 \*= sau;*  *printf("%c", b[(int) s2]);*  *s2 = fmod(s2, 1);*  *if(m%4==0) printf(" ");*  *m++;*  *}while (s2 > 0);*  *}*  *else*  *{*  *muoi = nhan (a, dau);*    *if (muoi != -1)*  *{*  *if((sau==2))*  *{*  *printf("\t\t\tKet qua: \t");*  *ket\_qua(0,muoi,sau,d);*  *dao\_bit\_cong\_1(d,c);*  *dau\_cach(d);*  *}*    *else if((sau!=2)&&(c[0]=='-'))*  *{*  *printf("\t\t\tKet qua: \t -");*  *ket\_qua(1,muoi,sau,d);*  *}*  *else*  *{*  *printf("\t\t\tKet qua: \t");*  *ket\_qua(1,muoi,sau,d);*  *}}}*  *printf("\nBam phim bat ky de tiep tuc ...");*  *getch();}* |