|  |  |
| --- | --- |
| Học trực tuyến CNTT, học lập trình từ cơ bản đến nâng cao |  |

**IOT301 – LẬP TRÌNH C CƠ BẢN**

**THUYẾT MINH ASSIGNMENT 1**

**LẬP TRÌNH GIẢI HÌNH TAM GIÁC**

|  |  |
| --- | --- |
| Tên học viên:  Mã học viên:  Ngày báo cáo: | Hoàng Văn Trường  FX32766  17/02/2024 |

**HÀ NỘI, 02/2024**

**MỤC LỤC**

[1. Danh sách thư viện 1](#_Toc159055459)

[2. Danh sách các biến toàn cục và các macro 1](#_Toc159055460)

[3. Danh sách các hàm 2](#_Toc159055461)

[4. Hàm nhapToaDoDiem(char diem[], double toado[]) 2](#_Toc159055462)

[5. Hàm doDaiCanh(char diemDau, char diemCuoi, char diem[], double toado[]) 3](#_Toc159055463)

[6. Hàm bool kiemtra\_tamgiac(char diem[], double toado[]) 4](#_Toc159055466)

[7. Hàm goccanh\_tamgiac(char diem[], double toado[]) 5](#_Toc159055469)

[8. Hàm xet\_tamgiac(char diem[], double gocCanh[]) 6](#_Toc159055472)

[9. Hàm dientich\_tamgiac(double gocCanh[]) 6](#_Toc159055475)

[10. Hàm duongcao\_tamgiac(char diem[], double gocCanh[]) 7](#_Toc159055478)

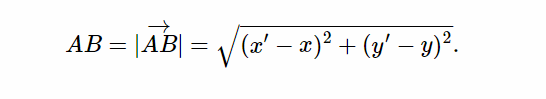
[11. Hàm trungtuyen\_tamgiac(char diem[], double gocCanh[]) 7](#_Toc159055481)

[12. Hàm tam\_tamgiac(double toado[]) 9](#_Toc159055484)

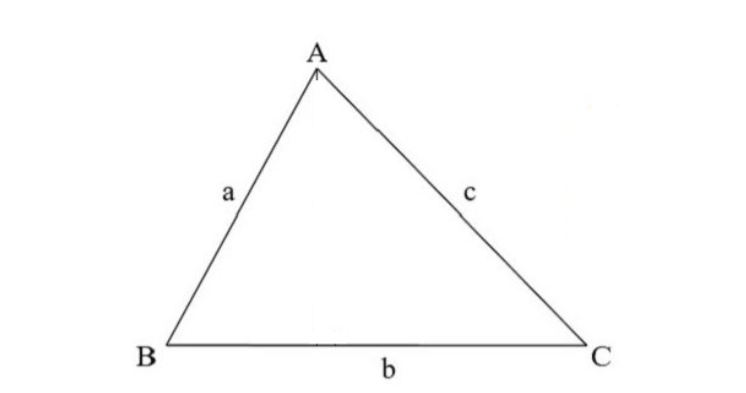
[13. Hàm giaima\_tamgiac(char diem[], double gocCanh[], double toado[]) 9](#_Toc159055487)

[14. Luồng xử lý code 10](#_Toc159055490)

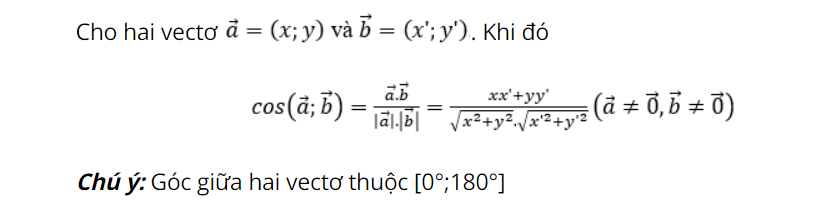
**DANH MỤC HÌNH VẼ, BẢNG BIỂU**

****

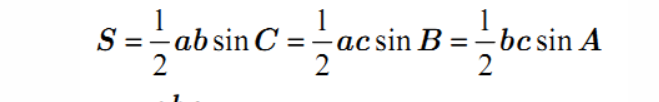
Hình 1: Độ dài 2 điểm trên mặt phẳng tọa độ Oxy



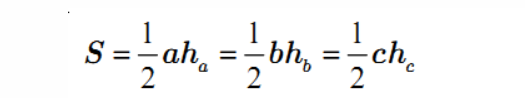
Hình 2: Bất đẳng thức tam giác

****

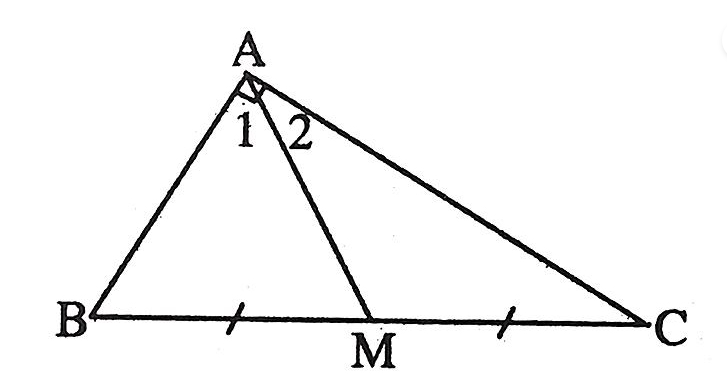
Hình 3: Công thức tính góc giữa 2 vector

****

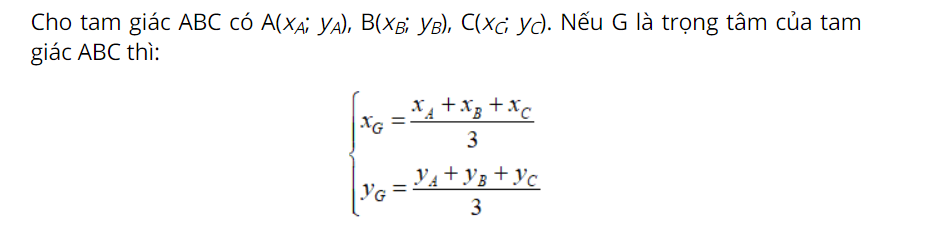
Hình 4: Công thức tính diện tích tam giác theo định lý sin

****

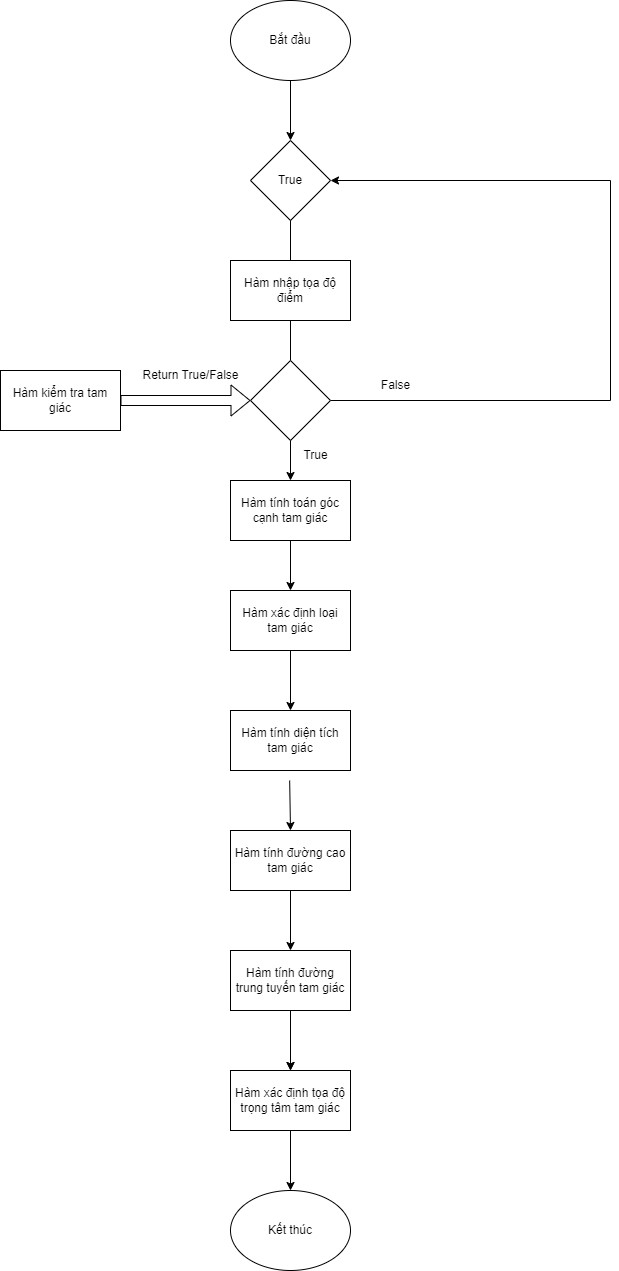
Hình 5: Công thức tính diện tích tam giác dựa trên đường cao



Hình 6: Đường trung tuyến tam giác

****

Hình 7: Tọa độ trọng tâm tam giác



Hình 8: Lưu đồ thuật toán

**NỘI DUNG THUYẾT MINH**

# Danh sách thư viện

|  |  |  |
| --- | --- | --- |
| **STT** | **Thư viện** | **Mô tả** |
| 1 | stdio.h | Thư viện có các hàm nhập xuất  + printf: In dữ liệu ra màn hình  + scanf: Nhập liệu |
| 2 | stdbool.h | Thư viện định nghĩa kiểu dữ liệu logic  + true: 1  + false: 0 |
| 3 | stdlib.h | Thư viện liên quan đến các hoạt động thực thi hệ thống và quản lý bộ nhớ  + system (thực thi lệnh hệ thống windows, ví dụ: cls, mkdir, dir, …)  + … |
| 4 | math.h | Thư viện toán học  + cos: Tính cos của góc  +... |
| 5 | windows.h | Thư viện chứa các định nghĩa và hàm API trong windows  + Sleep (Tạm dừng thực thi của một chương trình trong một khoảng thời gian cụ thể theo đơn vị mili giây)  + … |

# Danh sách các biến toàn cục và các macro

|  |  |  |
| --- | --- | --- |
| **STT** | **Biến toàn cục và macro** | **Mô tả** |
| 1 | PI 3.14159265 | Giá trị số PI |
| 2 | e 0.01 | Giá trị sai số nằm trong khoảng [0, 0.01] |
| 3 | double goc\_canh[6]; | Mảng lưu 3 góc và 3 cạnh của tam giác |
| 4 | double duongCao[3]; | Mảng lưu 3 đường cao của tam giác |
| 5 | double dienTich; | Biến toàn cục lưu trữ thông số diện tích tam giác |
| 6 | double trungTuyen[3] | Mảng lưu 3 đường trung tuyến của tam giác |
| 7 | double trongTam[2] | Mảng lưu 2 tọa độ trọng tâm của tam giác |

# Danh sách các hàm

|  |  |  |
| --- | --- | --- |
| **STT** | **Danh sách hàm** | **Mô tả** |
| 1 | int main() | Hàm chính của chương trình |
| 2 | double doDaiCanh(char diemDau, char diemCuoi, char diem[], double toado[]); | Tính toán độ dài cạnh giữa 2 điểm |
| 3 | void nhapToaDoDiem(char diem[], double toado[]); | Hàm nhập tọa độ 3 điểm và in ra màn hình |
| 4 | bool kiemtra\_tamgiac(char diem[], double toado[]); | Hàm kiểm tra tọa độ 3 điểm nhập vào có tạo thành tam giác không |
| 5 | void goccanh\_tamgiac(char diem[], double toado[]); | Hàm tính toán góc, cạnh của tam giác |
| 6 | void xet\_tamgiac(char diem[], double gocCanh[]); | Xác định loại tam giác (vuông, vuông cân, tù, tù cân, đều, ...) |
| 7 | void dientich\_tamgiac(double gocCanh[]); | Hàm tính diện tích tam giác |
| 8 | void duongcao\_tamgiac(char diem[], double gocCanh[]); | Hàm tính đường cao tam giác |
| 9 | void trungtuyen\_tamgiac(char diem[], double gocCanh[]); | Hàm tính đường trung tuyến tam giác |
| 10 | void tam\_tamgiac(double toado[]); | Hàm xác định tọa độ trọng tâm tam giác |
| 11 | void giaima\_tamgiac(char diem[], double gocCanh[], double toado[]); | Hàm chung gộp các hàm con lại, tạo vòng lặp đến khi người dùng nhập chính xác tọa độ 3 điểm là tam giác |

# Hàm nhapToaDoDiem(char diem[], double toado[])

1. *Mô tả*

Hàm nhapToaDoDiem(char diem[], double toado[]) yêu cầu người dùng nhập vào tọa độ 3 điểm A (xA, yA), B (xB, yB), C(xC, yC). Tọa độ 3 điểm sau khi được nhập vào sẽ được lưu vào một mảng cục bộ 6 phần tử với kiểu dữ liệu double. Sau khi nhận giá trị nhập vào tư người dùng, tiếp tục thực hiện vòng lặp in ra thông số tọa độ 3 điểm A, B, C đã lưu.

1. *Tham số truyền vào*

Mảng char có 3 phần tử lưu trữ 3 ký tự ‘A’, ‘B’, ‘C’. Mảng char này được sử dụng để tùy biến cho các đoạn code sau này với mục đích là thay đổi ký tự 3 đỉnh nhập vào. (Ví dụ: M, N, P)

Mảng double có 6 phần tử lưu trữ thông số tọa độ 3 điểm A(xA, yA), B (xB, yB), C(xC, yC) được nhập từ người dùng.

1. *Giá trị trả về*

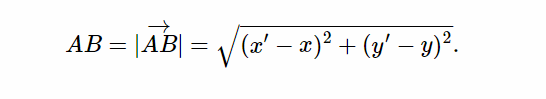
Không có giá trị trả về. Hàm yêu cầu người dùng nhập tọa độ 3 điểm A, B, C và in ra màn hình thông số tọa độ 3 điểm đã nhập.

# Hàm doDaiCanh(char diemDau, char diemCuoi, char diem[], double toado[])

1. *Mô tả*

Hàm doDaiCanh(char diemDau, char diemCuoi, char diem[], double toado[]) thực hiện tính toán độ dài giữa hai điểm.

Độ dài giữa 2 điểm A(x,y), B(x′,y′) được tính với công thức:

****

Hình 1: Độ dài 2 điểm trên mặt phẳng tọa độ Oxy

Tham khảo: [Công thức tính độ dài vectơ, đoạn thẳng trong hệ tọa độ Oxy và Oxyz - Toán Học Việt Nam (mathvn.com)](https://www.mathvn.com/2023/02/cong-thuc-tinh-o-dai-vecto-oan-thang.html)

# *Tham số truyền vào*

Biến char điểm đầu và điểm cuối để xác định giới hạn độ dài đoạn cần tính (AB, BC, CA – với 3 điểm A,B,C)

Mảng char có 3 phần tử lưu trữ 3 ký tự ‘A’, ‘B’, ‘C’. Mảng char này được sử dụng để so sánh với biến char điểm đầu và điểm cuối để tính chính xác độ dài thông qua thông số đã lưu trong mảng toado vì mảng tọa độ lưu trữ thông số lần lượt là (Ax, Ay, Bx, By, Cx, Cy).

Mảng double có 6 phần tử lấy thông số tọa độ 3 điểm A(xA, yA), B (xB, yB), C(xC, yC) đã lưu để tính toán độ dài.

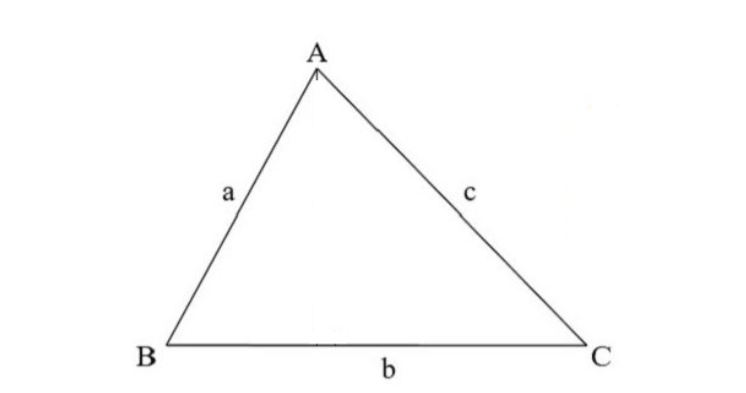
# *Giá trị trả về*

Trả về giá trị độ dài giữa 2 điểm truyền vào với kiểu dữ liệu double.

# Hàm bool kiemtra\_tamgiac(char diem[], double toado[])

1. *Mô tả*

Hàm kiemtra\_tamgiac(char diem[], double toado[]) thực hiện việc kiểm tra tọa độ 3 điểm A(xA, yA), B (xB, yB), C(xC, yC) nhập vào có tạo thành tam giác không dựa vào bất đẳng thức tam giác.



Hình 2: Bất đẳng thức tam giác

Bất đẳng thức tam giác: Tổng độ dài 2 cạnh bất kỳ của một tam giác luôn lớn hơn độ

dài cạnh còn lại. Công thức:

**a + b > c và a + c > b và b + c > a**

Tham khảo:

+ [Kiểm tra 3 cạnh nhập vào có tạo thành hình tam giác trong lập trình C/C++ (tuicocach.com)](https://tuicocach.com/kiem-tra-3-canh-nhap-vao-co-tao-thanh-hinh-tam-giac-trong-lap-trinh-c-c/) (sử dụng bất đẳng thức tam giác có vẻ tối ưu hơn)

+ [Chứng minh ba điểm thẳng hàng, điểm thuộc đường thẳng - VnHocTap.com](https://vnhoctap.com/chung-minh-ba-diem-thang-hang-diem-thuoc-duong-thang/) (cách chứng minh 3 điểm không thẳng hàng không tối ưu do không các định được các điểm trùng nhau, hoặc các vecto có tọa độ là 0)

# *Tham số truyền vào*

Mảng char có 3 phần tử lưu trữ 3 ký tự ‘A’, ‘B’, ‘C’. Mảng char này được sử dụng để gọi hàm doDaiCanh để tính toán độ dài các cạnh và sử dụng bất đẳng thức tam giác để xác định tọa độ 3 điểm nhập vào có tạo thành tam giác không.

Mảng double có 6 phần tử lấy thông số tọa độ 3 điểm A(xA, yA), B (xB, yB), C(xC, yC) đã lưu để tính toán độ dài.

# *Giá trị trả về*

Kiểu dữ liệu trả về là bool. Trong đó

+ true : Là tam giác

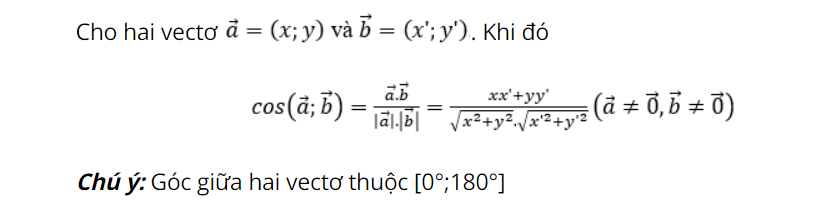
+ false: Không là tam giác

# Hàm goccanh\_tamgiac(char diem[], double toado[])

1. *Mô tả*

Hàm goccanh\_tamgiac(char diem[], double toado[]) thực hiện tính toán độ dài các cạnh và góc của tam giác sau đó lưu thông số vào mảng gocCanh được khai báo toàn cục.

Góc giữa hai cạnh được tính theo công thức

****

Hình 3: Công thức tính góc giữa 2 vector

Tham khảo:

+ [Công thức, cách tính góc giữa hai vecto (cực hay, chi tiết) (vietjack.com)](https://www.vietjack.com/toan-lop-10/cong-thuc-cach-tinh-goc-giua-hai-vecto-cuc-hay.jsp)

+ [Hàm cos() trong C | Thư viện C chuẩn (vietjack.com)](https://www.vietjack.com/thu-vien-c/ham-cos-trong-c.jsp)

# *Tham số truyền vào*

Mảng char có 3 phần tử lưu trữ 3 ký tự ‘A’, ‘B’, ‘C’. Mảng char này được sử dụng để gọi hàm doDaiCanh để tính toán độ dài các cạnh.

Mảng double có 6 phần tử lấy thông số tọa độ 3 điểm A(xA, yA), B (xB, yB), C(xC, yC) đã lưu để tính toán độ dài, góc của tam giác.

# *Giá trị trả về*

Không có giá trị trả về. Hàm in ra thông số cạnh và góc của tam giác.

# Hàm xet\_tamgiac(char diem[], double gocCanh[])

1. *Mô tả*

Hàm xet\_tamgiac(char diem[], double gocCanh[]) thực hiện xác định loại tam giác (vuông, vuông cân, tù, tù cân, đều, ...). Hàm thực hiện dựa trên việc so sánh góc lớn nhất với 90 độ. Nếu góc lớn nhất bằng 90 độ (sai số 0,01) thì tam giác đó là tam giác vuông. Nếu góc lớn nhất lớn hơn 90 độ thì là tam giác tù, các trường hợp còn lại thì là tam giác nhọn.

# *Tham số truyền vào*

Mảng char có 3 phần tử lưu trữ 3 ký tự ‘A’, ‘B’, ‘C’. Mảng char này được sử dụng để in ra các đỉnh tương ứng khi xác định được tam giác cân, vuông, tù tại đỉnh nào.

Mảng double có 6 phần tử lấy thông số góc cạnh đã lưu để tính toán xác định loại tam giác.

# *Giá trị trả về*

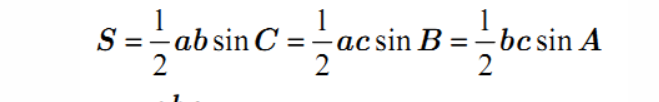
Không có giá trị trả về. Hàm in ra loại tam giác sau khi đã phân tích.

# Hàm dientich\_tamgiac(double gocCanh[])

1. *Mô tả*

Hàm dientich\_tamgiac(double gocCanh[]) thực hiện tính toán độ diện tích của tam giác.

Diện tích tam giác ABC được tính theo định lý sin với công thức:

****

Hình 4: Công thức tính diện tích tam giác theo định lý sin

Tham khảo: [Định lý sin – Wikipedia tiếng Việt](https://vi.wikipedia.org/wiki/%C4%90%E1%BB%8Bnh_l%C3%BD_sin)

# *Tham số truyền vào*

Mảng double có 6 phần tử lấy thông số tọa góc cạnh đã lưu để tính toán diện tích tam giác.

# *Giá trị trả về*

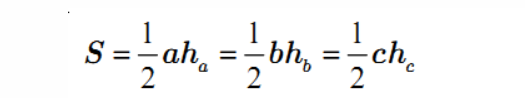
Không có giá trị trả về. Hàm in ra thông số diện tích tam giác đã tính toán.

# Hàm duongcao\_tamgiac(char diem[], double gocCanh[])

1. *Mô tả*

Hàm duongcao\_tamgiac(char diem[], double gocCanh[]) thực hiện tính toán độ dài đường cao tam giác.

Độ dài đường cao tam giác được tính theo công thức tính diện tích:

****

Hình 5: Công thức tính diện tích tam giác dựa trên đường cao

Tham khảo: [Định lí cosin, định lý sin, các công thức tính diện tích tam giác - Toán Học Việt Nam (mathvn.com)](https://www.mathvn.com/2014/02/inh-li-cosin-inh-ly-sin-cac-cong-thuc.html)

# *Tham số truyền vào*

Mảng char có 3 phần tử lưu trữ 3 ký tự ‘A’, ‘B’, ‘C’. Mảng char này được sử dụng để in ra các đỉnh tương ứng khi tính toán đường cao tại đỉnh xác định.

Mảng double có 6 phần tử lấy thông số góc cạnh đã lưu để tính toán độ dài đường cao.

# *Giá trị trả về*

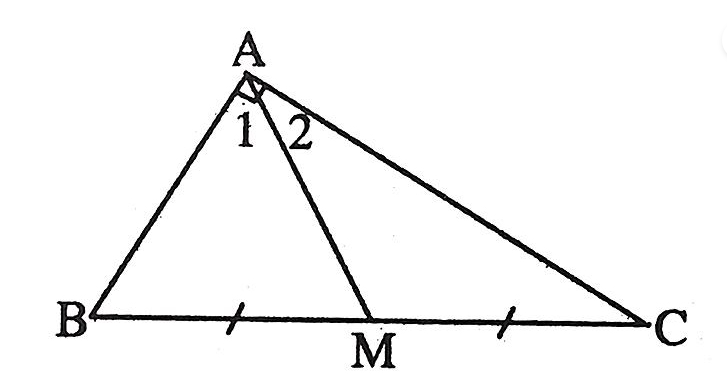
Không có giá trị trả về. Hàm in ra thông số đường cao tam giác tại các đỉnh.

# Hàm trungtuyen\_tamgiac(char diem[], double gocCanh[])

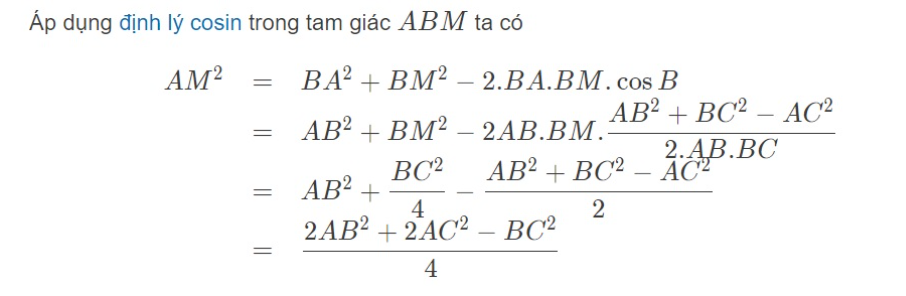
1. *Mô tả*

Hàm trungtuyen\_tamgiac(char diem[], double gocCanh[]) thực hiện tính toán độ dài đường trung tuyến tại các đỉnh tương ứng.

Độ dài đường trung tuyến được tính với công thức:



Hình 6: Đường trung tuyến tam giác

****

Tham khảo: [[Toán 10]: Chứng minh công thức đường trung tuyến chuẩn nhất - Giadinhphapluat.vn](https://giadinhphapluat.vn/toan-10-chung-minh-cong-thuc-duong-trung-tuyen-chuan-nhat/)

# *Tham số truyền vào*

Mảng char có 3 phần tử lưu trữ 3 ký tự ‘A’, ‘B’, ‘C’. Mảng char này được sử dụng để in ra các đỉnh tương ứng khi tính toán đường trung tuyến tại đỉnh xác định.

Mảng double có 6 phần tử lấy thông số góc cạnh đã lưu để tính toán độ dài đường trung tuyến.

# *Giá trị trả về*

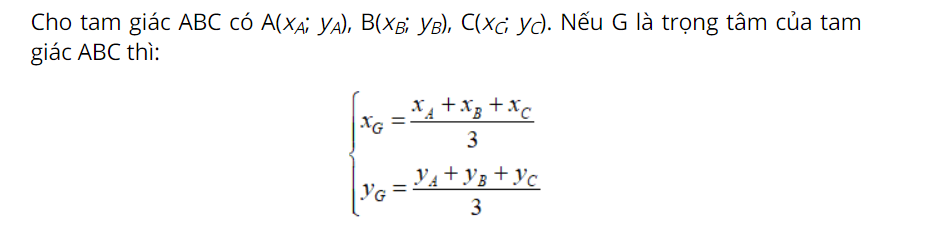
Không có giá trị trả về. Hàm in độ dài đường trung tuyến tại các đỉnh với kiểu dữ liệu double.

# Hàm tam\_tamgiac(double toado[])

1. *Mô tả*

Hàm tam\_tamgiac(double toado[]) thực hiện tính toán tọa độ tâm tam giác trong mặt phẳng tọa độ Oxy.

Tọa độ trọng tâm tam giác được tính với công thức:

****

Hình 7: Tọa độ trọng tâm tam giác

Tham khảo: [Cách tìm tọa độ của trọng tâm tam giác (cực hay, chi tiết) (vietjack.com)](https://vietjack.com/toan-lop-10/cach-tim-toa-do-cua-trong-tam-tam-giac-cuc-hay.jsp)

# *Tham số truyền vào*

Mảng double có 6 phần tử lấy thông số tọa độ 3 điểm A(xA, yA), B (xB, yB), C(xC, yC) đã lưu để tính toán tọa độ trọng tâm tam giác ABC.

# *Giá trị trả về*

Không có giá trị trả về. Hàm in ra thông số tọa độ trọng tâm tam giác ABC.

# Hàm giaima\_tamgiac(char diem[], double gocCanh[], double toado[])

1. *Mô tả*

Hàm giaima\_tamgiac(char diem[], double gocCanh[], double toado[]) thực hiện nhóm các hàm con lại và tạo vòng lặp kiểm tra xem tọa độ người dùng nhập vào có tạo thành tam giác không. Nếu tọa độ nhập vào không tạo thành tam giác thì yêu cầu người dùng nhập lại.

# *Tham số truyền vào*

Mảng char điểm được sử dụng để truyền vào hàm con được gọi.

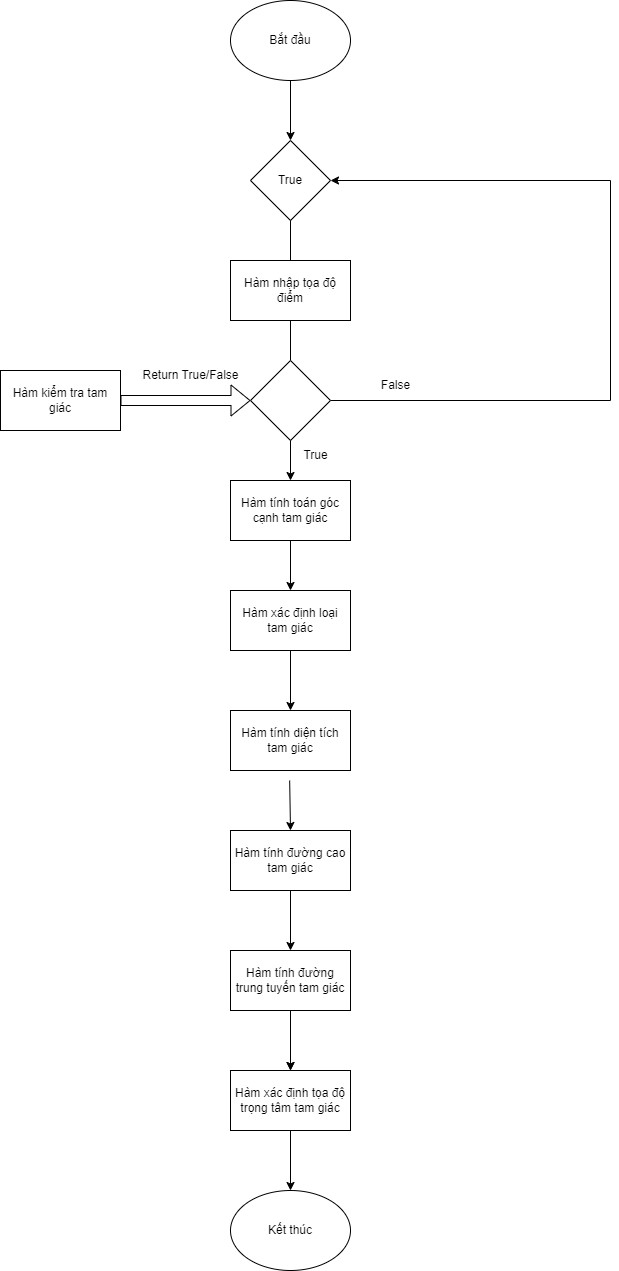
Mảng double có 6 phần tử chứa thông tin góc cạnh của tam giác được sử dụng để truyền vào hàm con được gọi.

Mảng double có 6 phần tử lấy thông số tọa độ 3 điểm A(xA, yA), B (xB, yB), C(xC, yC) đã lưu để truyền vào hàm con được gọi.

# *Giá trị trả về*

Hàm không có giá trị trả về. Hàm thực hiện nhiệm vụ gọi và thực thi các hàm con trong vòng lặp kiểm tra xem thông số tọa độ nhập vào có tạo thành tam giác hay không.

# Luồng xử lý code



Hình 8: Lưu đồ thuật toán

Các bước thực xử lý:

*Bước 1:* Nhập tọa độ 3 điểm của tam giác và lưu vào biến toàn cục toado[6];

Trong đó:

Ax = toado[0];

Bx = toado[1];

Cx = toado[2];

…

*Bước 2:* Kiểm tra 3 điểm nhập vào có phải tam giác hay không?

*Bước 3:* Nếu 3 điểm nhập vào không tạo thành tam giác thì yêu cầu người dùng nhập lại. Nếu không thì thực hiện tính toán góc cạnh của tam giác.

*Bước 4:* Xác định loại tam giác (vuông, vuông cân, tù, tù cân, đều, ...)

*Bước 5:* Tính diện tích tam giác

*Bước 6:* Tính đường cao tam giác

*Bước 7:* Tính đường trung tuyến tam giác

*Bước 8:* xác định tọa độ trọng tâm tam giác

*Bước 9:* Thoát khỏi vòng lặp với toàn bộ thông tin của tam giác