

Chương 9 – THIẾT KẾ LỚP HÌNH CẦU

- Nguyễn Hữu Lợi
- Đoàn Chánh Thống
- ThS. Nguyễn Thành Hiệp
- ThS. Trương Quốc Dũng
- ThS. Võ Duy Nguyên
- ThS. Nguyễn Văn Toàn
- TS. Nguyễn Duy Khánh
- TS. Nguyễn Tấn Trần Minh Khang

Thiết kế lớp CHìnhCau

- Thuộc tính
 - + Tâm.
 - + Bán kính.



Thiết kế lớp CHinhCau

- Các nhóm phương thức
 - + Nhóm phương thức khởi tạo.
 - + Nhóm phương thức cung cấp thông tin.
 - + Nhóm phương thức cập nhật thông tin.
 - + Nhóm phương thức kiểm tra.
 - + Nhóm phương thức xử lý.



Thiết kế lớp CHinhCau

- Nhóm phương thức khởi tạo
 - + Phương thức Nhập.
 - + Toán tử vào (operator >>).
 - + ...



Thiết kế lớp CHinhCau

– Nhóm phương thức khởi tạo (tiếp tục)

+ ...

+ Phương thức thiết lập mặc định.

+ Phương thức thiết lập sao chép.

+ Phương thức thiết lập khi biết đầy đủ thông tin.

+ ...



Thiết kế lớp CHinhCau

– Nhóm phương thức khởi tạo (tiếp tục)

+ ...

+ Phương thức thiết lập khi biết tâm, bán kính lấy mặc định.

+ Phương thức thiết lập khi biết bán kính, tâm lấy mặc định.

+ ...



Thiết kế lớp CHinhCau

- Nhóm phương thức khởi tạo (tiếp tục)
 - + ...
 - + Phương thức khởi tạo mặc định.
 - + Phương thức khởi tạo sao chép.
 - + Phương thức khởi tạo khi biết đầy đủ thông tin.
 - + ...



Thiết kế lớp CHinhCau

– Nhóm phương thức khởi tạo (tiếp tục)

+ ...

+ Phương thức khởi tạo khi biết tâm, bán kính lấy mặc định.

+ Phương thức khởi tạo khi biết bán kính, tâm lấy mặc định.

+ Kết thúc nhóm phương thức khởi tạo



Thiết kế lớp CHinhCau

- Nhóm phương thức cung cấp thông tin
 - + Phương thức Xuất.
 - + Toán tử ra (operator <<).
 - + ...
- + Lớp CHinhCau có bao nhiêu thuộc tính.
- + Trả lời: Lớp CHinhCau có 2 thuộc tính.



Thiết kế lớp CHinhCau

– Nhóm phương thức cung cấp thông tin (tiếp tục)

+ ...

+ Phương thức cung cấp tâm.

+ Phương thức cung cấp bán kính.

+ Kết thúc nhóm phương thức cung cấp thông tin.



Thiết kế lớp CHinhCau

- Nhóm phương thức cập nhật thông tin
 - + Toán tử gán (operator =).
 - + ...
- + Lớp CHinhCau có bao nhiêu thuộc tính.
- + Trả lời: Lớp CHinhCau có 2 thuộc tính.



Thiết kế lớp CHinhCau

– Nhóm phương thức cập nhật thông tin (tiếp tục)

+ ...

+ Phương thức cập nhật tâm.

+ Phương thức cập nhật bán kính.

+ Kết thúc nhóm phương thức cung cấp thông tin.



Thiết kế lớp CHinhCau

- Nhóm phương thức kiểm tra
 - + Toán tử so sánh bằng.
 - + Toán tử so sánh khác.
 - + Toán tử so sánh lớn hơn.
 - + Toán tử so sánh nhỏ hơn.
 - + Toán tử so sánh lớn hơn bằng.
 - + Toán tử so sánh nhỏ hơn bằng.
 - + ...



Thiết kế lớp CHinhCau

– Nhóm phương thức kiểm tra (tiếp tục)

+ ...

+ Kiểm tra hình cầu có đi qua gốc tọa độ không?

+ Kiểm tra hình cầu có chứa gốc tọa độ không?

+ Kiểm tra hình cầu có tiếp xúc mặt phẳng Oxy không?

+ Kiểm tra hình cầu có tiếp xúc mặt phẳng Oyz không?

+ Kiểm tra hình cầu có tiếp xúc mặt phẳng Ozx không?

+ Kiểm tra hình cầu là hình cầu đơn vị không?

+ ...

Thiết kế lớp CHinhCau

- Nhóm phương thức kiểm tra (tiếp tục)
 - + ...
 - + Kiểm tra hai hình cầu có nằm ngoài nhau không?
 - + Kiểm tra hai hình cầu có tiếp xúc ngoài không?
 - + Kiểm tra hai hình cầu có cắt nhau không?
 - + Kiểm tra hai hình cầu có tiếp xúc trong không?
 - + Kiểm tra hai hình cầu có nằm trong nhau không?
 - + ...

Thiết kế lớp CHinhCau

– Nhóm phương thức kiểm tra (tiếp tục)

+ ...

+ Kiểm tra hai hình cầu có trùng nhau không?

+ Kiểm tra hai hình cầu có không trùng nhau không?

+ Kiểm tra hai hình cầu có đồng tâm không?

+ ...

Thiết kế lớp CHinhCau

- Nhóm phương thức kiểm tra (tiếp tục)
 - + ...
 - + Kiểm tra một điểm có nằm trong hình cầu không?
 - + Kiểm tra một điểm có nằm trên hình cầu không?
 - + Kiểm tra một điểm có nằm ngoài hình cầu không?
- + Kết thúc nhóm phương thức kiểm tra.

Thiết kế lớp CHinhCau

- Nhóm phương thức xử lý
 - + Phương thức phá hủy.
 - + ...



Thiết kế lớp CHinhCau

- Nhóm phương thức xử lý (tiếp tục)
 - + ...
 - + Tính khoảng cách đến gốc tọa độ.
 - + Tính thể tích hình cầu.
 - + Tính diện tích xung quanh hình cầu.
 - + Kết thúc nhóm phương thức xử lý.



KHAI BÁO LỚP

Thiết kế lớp CHìnhCau

- Thuộc tính
 - + Tâm.
 - + Bán kính.



Khai báo lớp CHinhCau

```
11.class CHinhCau
12.{
13.    private:
14.        CDiemKhongGian I;
15.        float R;
16.    public:
17.        ///// Nhóm phương thức khởi tạo
18.        ...
```

Thiết kế lớp CHinhCau

- Nhóm phương thức khởi tạo
 - + Phương thức Nhập.
 - + Toán tử vào (operator >>).
 - + ...



Khai báo lớp CHinhCau

```
11. | ...
12. | public:
13. |     ///// Nhóm phương thức khởi tạo
14. |     ///// Chương 03 - Lập Trình HĐT với C++
15. |     void Nhap();
16. |     ///// Chương 04 - Iostream Cơ Bản
17. |     friend istream& operator >>(istream&,
18. |                                   CHinhCau&);
19. |     ...
```

Thiết kế lớp CHinhCau

– Nhóm phương thức khởi tạo (tiếp tục)

+ ...

+ Phương thức thiết lập mặc định.

+ Phương thức thiết lập sao chép.

+ Phương thức thiết lập khi biết đầy đủ thông tin.

+ ...



Thiết kế lớp CHinhCau

– Nhóm phương thức khởi tạo (tiếp tục)

+ ...

+ Phương thức thiết lập khi biết tâm, bán kính lấy mặc định.

+ Phương thức thiết lập khi biết bán kính, tâm lấy mặc định.

+ ...



Khai báo lớp CHinhCau

```
11.  // /// Nhóm phương thức khởi tạo
12.  // /// Chương 05 - Phương thức thiết lập -
13.  // /// Phương thức phá hủy
14.  ...
15.  CHinhCau();
16.  CHinhCau(const CHinhCau&);
17.  CHinhCau(CDiemKhongGian, float);
18.  CHinhCau(CDiemKhongGian);
19.  CHinhCau(float);
```

Thiết kế lớp CHinhCau

- Nhóm phương thức khởi tạo (tiếp tục)
 - + ...
 - + Phương thức khởi tạo mặc định.
 - + Phương thức khởi tạo sao chép.
 - + Phương thức khởi tạo khi biết đầy đủ thông tin.
 - + ...



Thiết kế lớp CHinhCau

– Nhóm phương thức khởi tạo (tiếp tục)

+ ...

+ Phương thức khởi tạo khi biết tâm, bán kính lấy mặc định.

+ Phương thức khởi tạo khi biết bán kính, tâm lấy mặc định.

+ Kết thúc nhóm phương thức khởi tạo



Khai báo lớp CHinhCau

```
11.  // /// Nhóm phương thức khởi tạo
12.  // /// Chương 09 - Thiết kế lớp
13.  ...
14.  void KhoiTao();
15.  void KhoiTao(const CHinhCau&);
16.  void KhoiTao(CDiemKhongGian, float);
17.  void KhoiTao(CDiemKhongGian);
18.  void KhoiTao(float);
19.  // /// Nhóm phương thức cung cấp thông tin
```

Thiết kế lớp CHinhCau

- Nhóm phương thức cung cấp thông tin
 - + Phương thức Xuất.
 - + Toán tử ra (operator <<).
 - + ...
- + Lớp CHinhCau có bao nhiêu thuộc tính.
- + Trả lời: Lớp CHinhCau có 2 thuộc tính.



Khai báo lớp CHinhCau

```
11. ...  
12. ///// Nhóm phương thức cung cấp thông tin  
13. ///// Chương 03 - Lập Trình HĐT với C++  
14. void Xuat();  
15. ///// Chương 04 - Iostream Cơ Bản  
16. friend ostream& operator <<(ostream&  
17.                               CHinhCau&);  
18. ...
```

Thiết kế lớp CHinhCau

– Nhóm phương thức cung cấp thông tin (tiếp tục)

+ ...

+ Phương thức cung cấp tâm.

+ Phương thức cung cấp bán kính.

+ Kết thúc nhóm phương thức cung cấp thông tin.



Khai báo lớp CHinhCau

```
11.  // /// Nhóm phương thức cung cấp thông tin
12.  // /// Chương 09 - Thiết kế lớp
13.  ...
14.  CDiemKhongGian getI();
15.  float getR();

16.  // /// Nhóm phương thức cập nhật thông tin
17.  ...
```

Thiết kế lớp CHinhCau

- Nhóm phương thức cập nhật thông tin
 - + Toán tử gán (operator =).
 - + ...
- + Lớp CHinhCau có bao nhiêu thuộc tính.
- + Trả lời: Lớp CHinhCau có 2 thuộc tính.



Thiết kế lớp CHinhCau

– Nhóm phương thức cập nhật thông tin (tiếp tục)

+ ...

+ Phương thức cập nhật tâm.

+ Phương thức cập nhật bán kính.

+ Kết thúc nhóm phương thức cung cấp thông tin.



Khai báo lớp CHinhCau

```
11. ...  
12. ///// Nhóm phương thức cập nhật thông tin  
13. ///// Chương 06 - Toán tử gán  
14. CHinhCau& operator =(const CHinhCau&);  
15. ///// Chương 09 - Thiết kế lớp  
16. void setI(CDiemKhongGian);  
17. void setR(float);  
18. ///// Nhóm phương thức kiểm tra  
19. ...
```

Thiết kế lớp CHinhCau

- Nhóm phương thức kiểm tra
 - + Toán tử so sánh bằng.
 - + Toán tử so sánh khác.
 - + Toán tử so sánh lớn hơn.
 - + Toán tử so sánh nhỏ hơn.
 - + Toán tử so sánh lớn hơn bằng.
 - + Toán tử so sánh nhỏ hơn bằng.
 - + ...



Khai báo lớp CHinhCau

```
11.      ///// Nhóm phương thức kiểm tra
12.      ///// Chương 08 - Toán tử so sánh
13.      bool operator ==(const CHinhCau&);
14.      bool operator !=(const CHinhCau&);
15.      bool operator >(const CHinhCau&);
16.      bool operator <(const CHinhCau&);
17.      bool operator >=(const CHinhCau&);
18.      bool operator <=(const CHinhCau&);
19.      ...
```

Thiết kế lớp CHinhCau

– Nhóm phương thức kiểm tra (tiếp tục)

+ ...

+ Kiểm tra hình cầu có đi qua gốc tọa độ không?

+ Kiểm tra hình cầu có chứa gốc tọa độ không?

+ Kiểm tra hình cầu có tiếp xúc mặt phẳng Oxy không?

+ Kiểm tra hình cầu có tiếp xúc mặt phẳng Oyz không?

+ Kiểm tra hình cầu có tiếp xúc mặt phẳng Ozx không?

+ Kiểm tra hình cầu là hình cầu đơn vị không?

+ ...

Khai báo lớp CHinhCau

```
11.  // /// Nhóm phương thức kiểm tra
12.  // /// Chương 09 - Thiết kế lớp
13.  bool ktQuaGoc();
14.  bool ktChuaGoc();
15.  bool ktTiepXucOxy();
16.  bool ktTiepXucOyz();
17.  bool ktTiepXucOzx();
18.  bool ktDonVi();
19.  ...
```

Thiết kế lớp CHinhCau

- Nhóm phương thức kiểm tra (tiếp tục)
 - + ...
 - + Kiểm tra hai hình cầu có nằm ngoài nhau không?
 - + Kiểm tra hai hình cầu có tiếp xúc ngoài không?
 - + Kiểm tra hai hình cầu có cắt nhau không?
 - + Kiểm tra hai hình cầu có tiếp xúc trong không?
 - + Kiểm tra hai hình cầu có nằm trong nhau không?
 - + ...

Khai báo lớp CHinhCau

```
11.  // /// Nhóm phương thức kiểm tra
12.  // /// Chương 09 - Thiết kế lớp
13.  ...
14.  bool isNamNgoai(const CHinhCau&);
15.  bool isTiepXucNgoai(const CHinhCau&);
16.  bool isGiaoNhau(const CHinhCau&);
17.  bool isTiepXucTrong(const CHinhCau&);
18.  bool isNamTrong(const CHinhCau&);
19.  ...
```


Thiết kế lớp CHinhCau

- Nhóm phương thức kiểm tra (tiếp tục)
 - + ...
 - + Kiểm tra hai hình cầu có trùng nhau không?
 - + Kiểm tra hai hình cầu có không trùng nhau không?
 - + Kiểm tra hai hình cầu có đồng tâm không?
 - + ...

Khai báo lớp CHinhCau

```
11.  //... Nhóm phương thức kiểm tra
12.  //... Chương 09 - Thiết kế lớp
13.  ...
14.  bool ktTrung(const CHinhCau&);
15.  bool ktKhongTrung(const CHinhCau&);
16.  bool ktDongTam(const CHinhCau&);
17.  ...
```

Thiết kế lớp CHinhCau

- Nhóm phương thức kiểm tra (tiếp tục)
 - + ...
 - + Kiểm tra một điểm có nằm trong hình cầu không?
 - + Kiểm tra một điểm có nằm trên hình cầu không?
 - + Kiểm tra một điểm có nằm ngoài hình cầu không?
- + Kết thúc nhóm phương thức kiểm tra.

Khai báo lớp CHinhCau

```
11.  ///// Nhóm phương thức kiểm tra
12.  ///// Chương 09 - Thiết kế lớp
13.  ...
14.  bool isNamTrong(const CDiemKhongGian&);
15.  bool isNamTren(const CDiemKhongGian&);
16.  bool isNamNgoai(const CDiemKhongGian&);

17.  ///// Nhóm phương thức xu ly
```

Thiết kế lớp CHinhCau

- Nhóm phương thức xử lý
 - + Phương thức phá hủy.
 - + ...



Khai báo lớp CHinhCau

```
11. | ...
12. | //// Nhóm phương thức xử lý
13. | //// Chương 05 - Phương thức thiết lập -
14. | //// Phương thức phá hủy
15. | ~CHinhCau();
16. | ...
```

Thiết kế lớp CHinhCau

- Nhóm phương thức xử lý (tiếp tục)
 - + ...
 - + Tính khoảng cách đến gốc tọa độ.
 - + Tính thể tích hình cầu.
 - + Tính diện tích xung quanh hình cầu.
 - + Kết thúc nhóm phương thức xử lý.



Khai báo lớp CHinhCau

```
11. |      ///// Nhóm phương thức xử lý
12. |      ///// Chương 09 - Thiết kế lớp
13. |      ...
14. |      float KhoangCachGoc();
15. |      float TheTich();
16. |      float DienTichXungQuanh();
17. |};
```

Định nghĩa các phương thức

NHÓM PHƯƠNG THỨC KHỞI TẠO

Định nghĩa phương thức

```
11. void CTamGiac::Nhap()  
12. {  
13.     cout << "Nhap A: \n";  
14.     A.Nhap();  
15.     cout << "Nhap B: \n";  
16.     B.Nhap();  
17.     cout << "Nhap C: \n";  
18.     C.Nhap();  
19. }
```


Định nghĩa phương thức

```
11. istream& operator >>(istream& is, CTamGiac& t)
12. {
13.     cout << "Nhap A: " << endl;
14.     is >> t.A;
15.     cout << "Nhap B: " << endl;
16.     is >> t.B;
17.     cout << "Nhap C: " << endl;
18.     is >> t.C;
19.     return is;
20. }
```

Định nghĩa phương thức

```
11. CTamGiac::CTamGiac()  
12. {  
13.     A = CDiem();  
14.     B = CDiem();  
15.     C = CDiem();  
16. }
```

Định nghĩa phương thức

```
11. CTamGiac::CTamGiac(const CTamGiac& t)
12. {
13.     A = t.A;
14.     B = t.B;
15.     C = t.C;
16. }
```

Định nghĩa phương thức

```
11. CTamGiac::CTamGiac(CDiem AA, CDiem BB)
12. {
13.     A = AA;
14.     B = BB;
15.     C = CDiem();
16. }
```

Định nghĩa phương thức

```
11. CTamGiac::CTamGiac(CDiem AA)
12. {
13.     A = AA;
14.     B = CDiem();
15.     C = CDiem();
16. }
```


Định nghĩa phương thức

```
11. void CTamGiac::KhoiTao()  
12. {  
13.     A = CDiem();  
14.     B = CDiem();  
15.     C = CDiem();  
16. }
```

Định nghĩa phương thức

```
11. void CTamGiac::KhoiTao(const CTamGiac& t)
12. {
13.     A = t.A;
14.     B = t.B;
15.     C = t.C;
16. }
```

Định nghĩa phương thức

```
11. void CTamGiac::KhoiTao(CDiem AA, CDiem BB, CDiem CC)
12. {
13.     A = AA;
14.     B = BB;
15.     C = CC;
16. }
```

Định nghĩa phương thức

```
11. void CTamGiac::KhoiTao(CDiem AA, CDiem BB)
12. {
13.     A = AA;
14.     B = BB;
15.     C = CDiem();
16. }
```

Định nghĩa phương thức

```
11. void CTamGiac::KhoiTao(CDiem AA)
12. {
13.     A = AA;
14.     B = CDiem();
15.     C = CDiem();
16. }
```


Định nghĩa các phương thức

NHÓM PHƯƠNG THỨC CUNG CẤP THÔNG TIN

Định nghĩa phương thức

```
11. void CTamGiac::Xuat()  
12. {  
13.     cout << "\nDiem A vua nhap: ";  
14.     A.Xuat();  
15.     cout << "\nDiem B vua nhap: ";  
16.     B.Xuat();  
17.     cout << "\nDiem C vua nhap: ";  
18.     C.Xuat();  
19. }
```

Định nghĩa phương thức

```
11. ostream& operator <<(ostream& os, CTamGiac& t)
12. {
13.     os << "\nDiem A vua nhap: ";
14.     os << t.A;
15.     os << "\nDiem B vua nhap: ";
16.     os << t.B;
17.     os << "\nDiem C vua nhap: ";
18.     os << t.C;
19.     return os;
20. }
```

Định nghĩa phương thức

```
11. CTamGiac::getA()  
12. {  
13.     return A;  
14. }
```

Định nghĩa phương thức

```
11. CTamGiac::getB()  
12. {  
13.     return B;  
14. }
```

Định nghĩa phương thức

```
11. CDiem CTamGiac::getC()  
12. {  
13.     return C;  
14. }
```


Định nghĩa các phương thức

NHÓM PHƯƠNG THỨC CẬP NHẬT THÔNG TIN

Định nghĩa phương thức

```
11. CTamGiac& CTamGiac::operator =(const CTamGiac& t)
12. {
13.     A = t.A;
14.     B = t.B;
15.     C = t.C;
16.     return *this;
17. }
```

Định nghĩa phương thức

```
11. void CTamGiac::setA(CDiem AA)
12. {
13.     A = AA;
14. }
```

Định nghĩa phương thức

```
11. void CTamGiac::setB(CDiem BB)
12. {
13.     B = BB;
14. }
```

Định nghĩa phương thức

```
11. void CTamGiac::setC(CDiem CC)
12. {
13.     C = CC;
14. }
```

Định nghĩa phương thức

```
11. void CTamGiac::setA(float xx, float yy)
12. {
13.     A.setX(xx);
14.     A.setY(yy);
15. }
```


Định nghĩa phương thức

```
11. void CTamGiac::setB(float xx, float yy)
12. {
13.     B.setX(xx);
14.     B.setY(yy);
15. }
```

Định nghĩa phương thức

```
11. void CTamGiac::setC(float xx, float yy)
12. {
13.     C.setX(xx);
14.     C.setY(yy);
15. }
```

Định nghĩa các phương thức

NHÓM PHƯƠNG THỨC KIỂM TRA

Định nghĩa phương thức

```
11. bool CTamGiac::operator ==(const CTamGiac& t)
12. {
13.     CTamGiac temp(t);
14.     if (DienTich() == temp.DienTich())
15.         return true;
16.     return false;
17. }
```

Định nghĩa phương thức

```
11. bool CTamGiac::operator !=(const CTamGiac& t)
12. {
13.     CTamGiac temp(t);
14.     if (DienTich() != temp.DienTich())
15.         return true;
16.     return false;
17. }
```

Định nghĩa phương thức

```
11. bool CTamGiac::operator >(const CTamGiac& t)
12. {
13.     CTamGiac temp(t);
14.     if (DienTich() > temp.DienTich())
15.         return true;
16.     return false;
17. }
```


Định nghĩa phương thức

```
11. bool CTamGiac::operator <(const CTamGiac& t)
12. {
13.     CTamGiac temp(t);
14.     if (DienTich() < temp.DienTich())
15.         return true;
16.     return false;
17. }
```

Định nghĩa phương thức

```
11. bool CTamGiac::operator >=(const CTamGiac& t)
12. {
13.     CTamGiac temp(t);
14.     if (DienTich() >= temp.DienTich())
15.         return true;
16.     return false;
17. }
```

Định nghĩa phương thức

```
11. bool CTamGiac::operator <=(const CTamGiac& t)
12. {
13.     CTamGiac temp(t);
14.     if (DienTich() <= temp.DienTich())
15.         return true;
16.     return false;
17. }
```

Định nghĩa phương thức

```
11. bool CTamGiac::ktHopLe()  
12. {  
13.     float a = B.KhoangCach(C);  
14.     float b = C.KhoangCach(A);  
15.     float c = A.KhoangCach(B);  
16.     if (a + b > c && a + c > b && b + c > a)  
17.         return true;  
18.     return false;  
19. }
```

Định nghĩa phương thức

```
11. bool CTamGiac::ktDeu()  
12. {  
13.     if (!ktHopLe())  
14.         return false;  
15.     float a = B.KhoangCach(C);  
16.     float b = C.KhoangCach(A);  
17.     float c = A.KhoangCach(B);  
18.     if (a == b && b == c)  
19.         return true;  
20.     return false;  
}
```

Định nghĩa phương thức

```
11. bool CTamGiac::ktCan()  
12. {  
13.     if (!ktHopLe())  
14.         return false;  
15.     float a = B.KhoangCach(C);  
16.     float b = C.KhoangCach(A);  
17.     float c = A.KhoangCach(B);  
18.     if (a == b || b == c || c == a)  
19.         return true;  
20.     return false;  
}
```


Định nghĩa phương thức

```
11. bool CTamGiac::ktVuong()  
12. {  
13.     if (!ktHopLe())  
14.         return false;  
15.     float a = B.KhoangCach(C);  
16.     float b = C.KhoangCach(A);  
17.     float c = A.KhoangCach(B);  
18.     if ((a * a + b * b == c * c) || (c * c + b * b  
    == a * a) || (a * a + c * c == b * b))  
19.         return true;  
20.     return false;
```

Định nghĩa phương thức

```
11. bool CTamGiac::ktVuongCan()  
12. {  
13.     if (!ktHopLe())  
14.         return false;  
15.     if (ktVuong() && ktCan())  
16.         return true;  
17.     return false;  
18. }
```

Định nghĩa phương thức

```
11. bool CTamGiac::ktTrung(const CTamGiac& t)
12. {
13.     if (A == t.A && B == t.B && C == t.C)
14.         return true;
15.     return false;
16. }
```

Định nghĩa phương thức

```
11. bool CTamGiac::ktKoTrung(const CTamGiac& t)
12. {
13.     if (!(A == t.A && B == t.B && C == t.C))
14.         return true;
15.     return false;
16. }
```

Định nghĩa các phương thức

NHÓM PHƯƠNG THỨC XỬ LÝ

Định nghĩa phương thức

```
11. CTamGiac::~~CTamGiac()  
12. {  
13.     return;  
14. }
```


Định nghĩa phương thức

```
11. CTamGiac CTamGiac::operator +(const CTamGiac& x)
12. {
13.     CTamGiac temp;
14.     temp.A = A + x.A;
15.     temp.B = B + x.B;
16.     temp.C = C + x.C;
17.     return temp;
18. }
```

Định nghĩa phương thức

```
11. CTamGiac CTamGiac::operator -(const CTamGiac& x)
12. {
13.     CTamGiac temp;
14.     temp.A = A - x.A;
15.     temp.B = B - x.B;
16.     temp.C = C - x.C;
17.     return temp;
18. }
```

Định nghĩa phương thức

```
11. CTamGiac& CTamGiac::operator +=(const CTamGiac& t)
12. {
13.     A += t.A;
14.     B += t.B;
15.     C += t.C;
16.     return *this;
17. }
```

Định nghĩa phương thức

```
11. CTamGiac& CTamGiac::operator -=(const CTamGiac& t)
12. {
13.     A -= t.A;
14.     B -= t.B;
15.     C -= t.C;
16.     return *this;
17. }
```

Định nghĩa phương thức

```
11. float CTamGiac::ChuVi()  
12. {  
13.     float a = B.KhoangCach(C);  
14.     float b = C.KhoangCach(A);  
15.     float c = A.KhoangCach(B);  
16.     return a + b + c;  
17. }
```

Định nghĩa phương thức

```
11. float CTamGiac::DienTich()  
12. {  
13.     float a = B.KhoangCach(C);  
14.     float b = C.KhoangCach(A);  
15.     float c = A.KhoangCach(B);  
16.     float p = (a + b + c) / 2;  
17.     return (float)sqrt(p * (p - a) * (p - b) * (p -  
    c));  
18. }
```


Định nghĩa phương thức

```
11. CDiem CTamGiac::TrongTam()  
12. {  
13.     CDiem temp;  
14.     temp.setX((A.getX() + B.getX() + C.getX()) / 3);  
15.     temp.setY((A.getY() + B.getY() + C.getY()) / 3);  
16.     return temp;  
17. }
```

Định nghĩa phương thức

```
11. CDiem CTamGiac::TrucTam()  
12. {  
13.     CDiem temp;  
14.     float d1A = C.getX() - B.getX();  
15.     float d1B = C.getY() - B.getY();  
16.     float d1C = A.getX() * d1A + A.getY() * d1B;  
  
17.     float d2A = C.getX() - A.getX();  
18.     float d2B = C.getY() - A.getY();  
19.     float d2C = B.getX() * d2A + B.getY() * d2B;
```

Cảm ơn quý vị đã lắng nghe

Nhóm tác giả

TS. Nguyễn Tấn Trần Minh Khang