

KHỞI TẠO ĐỐI TƯỢNG HÀM BẠN VÀ LỚP BẠN (TT)

C++



Microsoft®

Visual Studio®

HÀM BẠN, LỚP BẠN

- ❖ GIẢ SỬ CÓ LỚP VECTOR, LỚP MATRIX
 - ❖ CẦN VIẾT HÀM **NHÂN** VECTOR VỚI MỘT MATRIX
 - ❖ HÀM **NHÂN**:
 - KHÔNG THỂ THUỘC LỚP VECTOR
 - KHÔNG THỂ THUỘC LỚP MATRIX
 - KHÔNG THỂ TỰ DO
- GIẢI PHÁP: XÂY DỰNG HÀM TRUY CẬP DỮ LIỆU?

HÀM BẠN (FRIEND FUNCTION)

- ❖ HÀM BẠN KHÔNG THUỘC LỚP. TUY NHIÊN, CÓ QUYỀN TRUY CẬP CÁC THÀNH VIÊN PRIVATE.
- ❖ KHI ĐỊNH NGHĨA MỘT LỚP, CÓ THỂ KHAI BÁO MỘT HAY NHIỀU HÀM “BẠN” (BÊN NGOÀI LỚP)
- ❖ ƯU ĐIỂM:
 - KIỂM SOÁT CÁC TRUY NHẬP Ở CẤP ĐỘ LỚP – KHÔNG THỂ ÁP ĐẶT HÀM BẠN CHO LỚP NẾU ĐIỀU ĐÓ KHÔNG ĐƯỢC DỰ TRÙ TRƯỚC TRONG KHAI BÁO CỦA LỚP.

HÀM BẠN (FRIEND FUNCTION)

❖ CÁC TÍNH CHẤT CỦA QUAN HỆ FRIEND:

- PHẢI ĐƯỢC CHO, KHÔNG ĐƯỢC NHẬN
 - LỚP B LÀ BẠN CỦA LỚP A, LỚP A PHẢI KHAI BÁO RÕ RÀNG B LÀ BẠN CỦA NÓ
- KHÔNG ĐỐI XỨNG
- KHÔNG BẮC CẦU

VÍ DỤ

```
class COUNTERCLASS{  
    int Counter;  
public:  
    char CounterChar;  
    void Init( char );  
    void AddOne( ){  
        Counter++;  
    }  
    friend int Total (int);  
};
```

VÍ DỤ

```
COUNTERCLASS MyCounter[26];  
  
int Total(int NumberObjects)  
{  
    for (int i=0, sum=0; i<NumberObjects; i++)  
        sum += MyCounter[i].Counter  
    //Tính tổng số ký tự trong số các Objects ký tự  
    return sum;  
}
```

LỚP BẠN (FRIEND CLASS)

- ❖ MỘT LỚP **CÓ THỂ** TRUY CẬP ĐẾN CÁC THÀNH PHẦN CÓ THUỘC TÍNH **PRIVATE** CỦA MỘT LỚP KHÁC.
- ❖ ĐỂ THỰC HIỆN ĐƯỢC ĐIỀU NÀY, CHÚNG TA CÓ THỂ LẤY TOÀN BỘ MỘT LỚP LÀM BẠN (HÀM FRIEND) CHO LỚP KHÁC.



VÍ DỤ

```
class TOM{  
public:  
    friend class JERRY; //Có lớp bạn là JERRY  
private:  
    int SecretTom;      //Bí mật của TOM  
};  
class JERRY{  
public:  
    void Change(TOM T){  
        T.SecterTom++; //Bạn nên có thể thay thế  
    }  
};
```


GIAO DIỆN VÀ CHI TIẾT CÀI ĐẶT

- ❖ LỚP CÓ HAI PHẦN TÁCH RỜI
 - ❖ PHẦN GIAO DIỆN KHAI BÁO TRONG PHẦN PUBLIC ĐỂ NGƯỜI SỬ DỤNG “THẤY” VÀ SỬ DỤNG.
 - ❖ CHI TIẾT CÀI ĐẶT BAO GỒM DỮ LIỆU KHAI BÁO TRONG PHẦN PRIVATE CỦA LỚP VÀ CHI TIẾT MÃ HÓA CÁC HÀM THÀNH PHẦN, VÔ HÌNH ĐỐI VỚI NGƯỜI DÙNG.
- ❖ LỚP THOIDIEM CÓ THỂ ĐƯỢC CÀI ĐẶT VỚI CÁC THÀNH PHẦN DỮ LIỆU LÀ GIỜ, PHÚT, GIÂY HOẶC TỔNG SỐ GIÂY TÍNH TỪ 0 GIỜ.

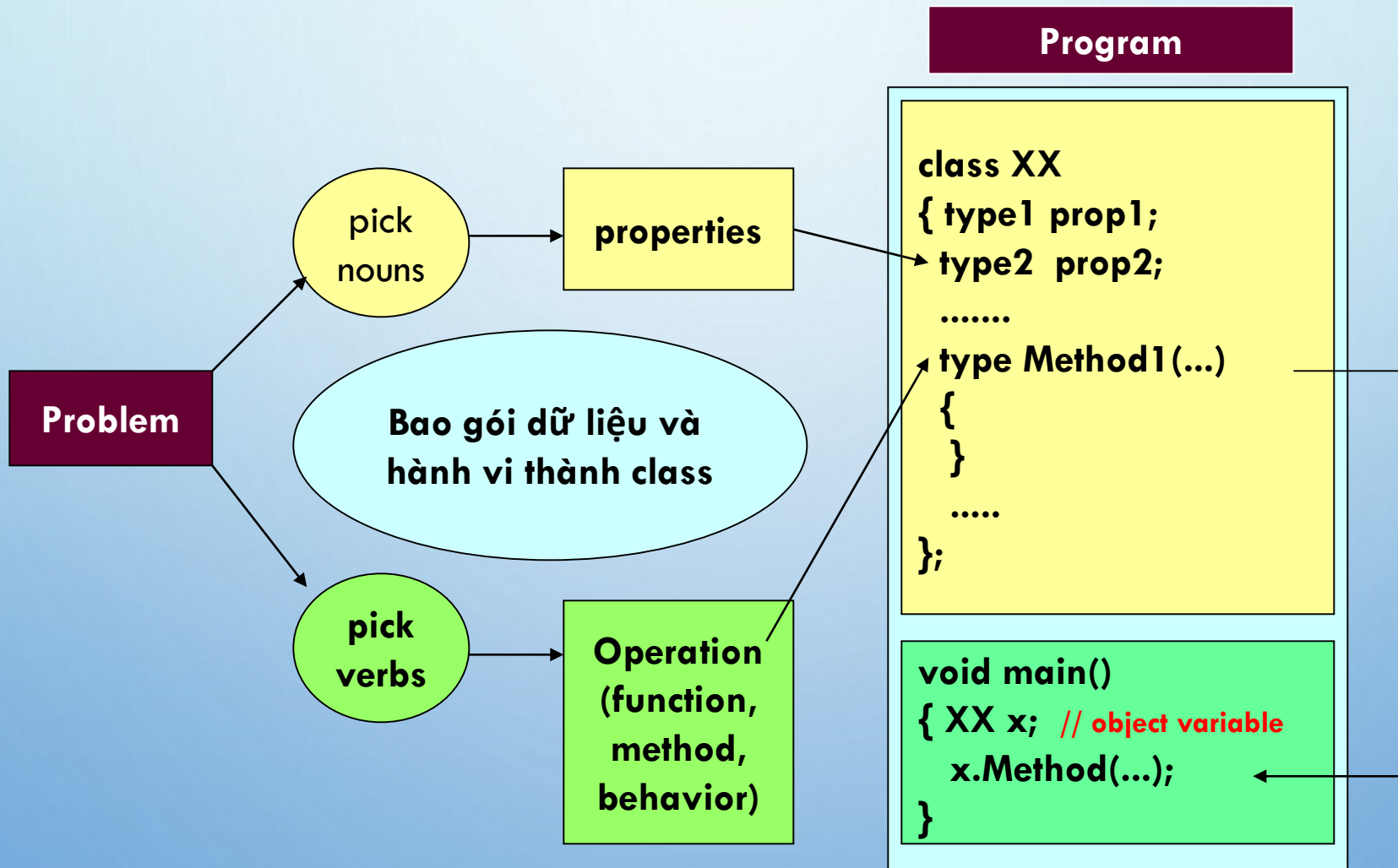
GIAO DIỆN VÀ CHI TIẾT CÀI ĐẶT

- ❖ TA CÓ THỂ THAY ĐỔI UYÊN CHUYỂN CHI TIẾT CÀI ĐẶT, NGHĨA LÀ CÓ THỂ THAY ĐỔI TỔ CHỨC DỮ LIỆU CỦA LỚP, CŨNG NHƯ CÓ THỂ THAY ĐỔI CHI TIẾT THỰC HIỆN CÁC HÀM THÀNH PHẦN (DO SỰ THAY ĐỔI TỔ CHỨC DỮ LIỆU HOẶC ĐỂ CẢI TIẾN GIẢI THUẬT). NHƯNG NẾU BẢO ĐẢM KHÔNG THAY ĐỔI PHẦN GIAO DIỆN THÌ KHÔNG ẢNH HƯỞNG ĐẾN NGƯỜI SỬ DỤNG, VÀ DO ĐÓ KHÔNG LÀM ĐỔ VỠ KIẾN TRÚC CỦA HỆ THỐNG.

CÁC NGUYÊN TẮC XÂY DỰNG LỚP

- ❖ **HÌNH THÀNH LỚP:** KHI TA NGHĨ ĐẾN “NÓ” NHƯ MỘT KHÁI NIỆM RIÊNG LẺ → XÂY DỰNG LỚP BIỂU DIỄN KHÁI NIỆM ĐÓ.
- ❖ **LỚP** LÀ BIỂU DIỄN CỤ THỂ CỦA MỘT KHÁI NIỆM VÌ VẬY **TÊN LỚP LUÔN LÀ DANH TỪ**.
- ❖ **CÁC THUỘC TÍNH** CỦA LỚP LÀ CÁC THÀNH PHẦN DỮ LIỆU NÊN CHÚNG **LUÔN LÀ DANH TỪ**.
- ❖ **CÁC HÀM THÀNH PHẦN** (CÁC HÀNH VI) LÀ CÁC THAO TÁC CHỈ RÕ HOẠT ĐỘNG CỦA LỚP NÊN **CÁC HÀM LÀ ĐỘNG TỪ**.

CÁC NGUYÊN TẮC XÂY DỰNG LỚP



CÁC NGUYÊN TẮC XÂY DỰNG LỚP

❖ CÁC THUỘC TÍNH CÓ THỂ SUY DIỄN TỪ NHỮNG THUỘC TÍNH KHÁC THÌ NÊN DÙNG HÀM THÀNH PHẦN ĐỂ THỰC HIỆN TÍNH TOÁN.

```
class TamGiac{  
    Diem A,B,C;  
    double ChuVi;  
    double DienTich;  
public:  
    //...  
};
```

```
class TamGiac{  
    Diem A,B,C;  
public:  
    //...  
    double ChuVi() const;  
    double DienTich() const;  
};
```

CÁC NGUYÊN TẮC XÂY DỰNG LỚP

❖ TUY NHIÊN, NẾU CÁC THUỘC TÍNH SUY DIỄN ĐÒI HỎI NHIỀU TÀI NGUYÊN HOẶC THỜI GIAN ĐỂ THỰC HIỆN TÍNH TOÁN, TA NÊN KHAI BÁO

```
class QuocGia{  
    long DanSo;  
    double DienTich;  
    double TuổiTrungBinh;  
public:  
    double TinhTuoiTB() const;  
    //...  
};
```

CÁC NGUYÊN TẮC XÂY DỰNG LỚP

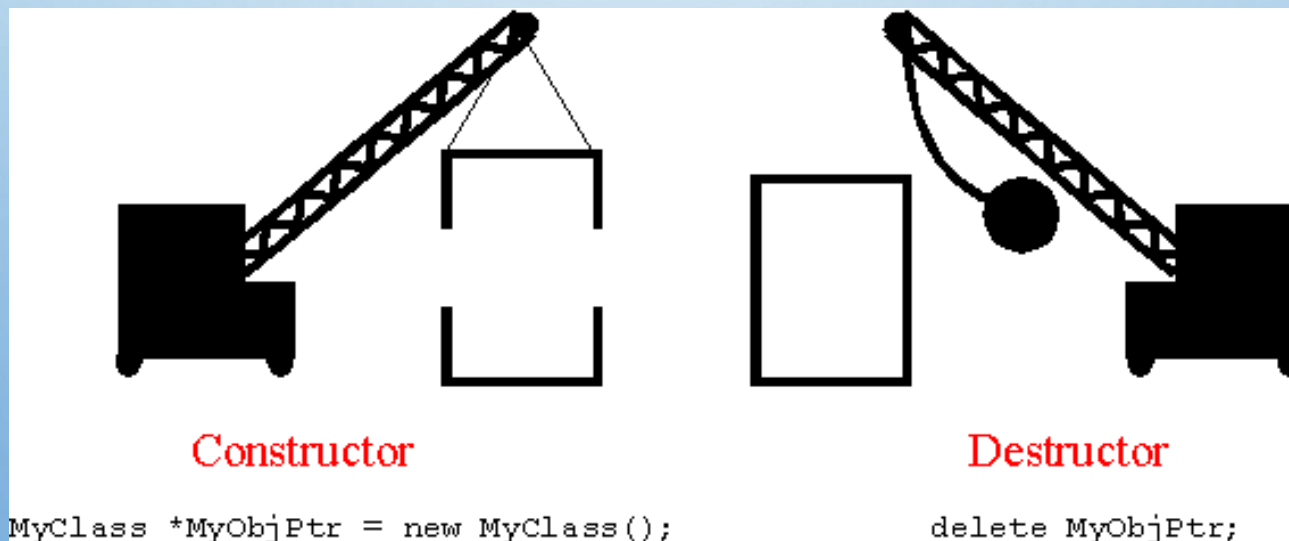
❖ DỮ LIỆU THÀNH PHẦN NÊN ĐƯỢC KẾT HỢP:

```
class TamGiac{  
    Diem A,B,C;  
public:  
    //...  
};  
class HìnhTron{  
    Diem Tam;  
    double BanKinh;  
public:  
    //...  
};
```

```
class TamGiac{  
    double xA, yA;  
    double xB, yB, xC, yC;  
public:  
    //...  
};  
class HìnhTron{  
    double tx, ty, BanKinh;  
public:  
    //...  
};
```


CÁC NGUYÊN TẮC XÂY DỰNG LỚP

- ❖ TRONG MỌI TRƯỜNG HỢP, NÊN CÓ PHƯƠNG THỨC THIẾT LẬP (CONSTRUCTOR) ĐỂ KHỞI ĐỘNG ĐỐI TƯỢNG.
- ❖ NÊN CÓ PHƯƠNG THỨC THIẾT LẬP CÓ KHẢ NĂNG TỰ KHỞI ĐỘNG KHÔNG CẦN THAM SỐ



CÁC NGUYÊN TẮC XÂY DỰNG LỚP

- ❖ NẾU ĐỐI TƯỢNG CÓ NHU CẦU CẤP PHÁT TÀI NGUYÊN THÌ PHẢI CÓ PHƯƠNG THỨC THIẾT LẬP, COPY CONSTRUCTOR ĐỂ KHỞI ĐỘNG ĐỐI TƯỢNG BẰNG ĐỐI TƯỢNG CÙNG KIỂU VÀ CÓ DESTRUCTOR ĐỂ DỌN DẸP. NGOÀI RA CÒN CÓ PHÉP GÁN (CHƯƠNG 5).
- ❖ NẾU ĐỐI TƯỢNG ĐƠN GIẢN KHÔNG CẦN TÀI NGUYÊN RIÊNG → KHÔNG CẦN COPY CONSTRUCTOR VÀ DESTRUCTOR