OBJECT ORIENTED PROGRAMING(LẬP TRÌNH HƯỚNG ĐỐI TƯỢNG)

Các tính chất :

Tính kế thừa

Tính đa hình

Tính đóng gói

Tính trừu tượng

|  |
| --- |
| #include <iostream>  using namespace std;  /\*  Trong ví dụ sau, lớp hinh là một lớp trừu tượng, vì nó có một phương thức ảo  virtual .  Điều này có nghĩa là lớp Shape không thể được khởi tạo thành đối tượng trực tiếp,  mà chỉ có thể được sử dụng thông qua con trỏ hoặc tham chiếu.  Cả lớp hinhCN và lớp hinhVuong đều kế thừa từ lớp hinh và định nghĩa lại phương thức các phương của riêng mình.  \*/  class hinh  {      public:      virtual void thietLap()= 0;      virtual void thayDoiDoDaiCanh(unsigned  int x, unsigned  int y)= 0;      virtual void thayDoiDoDaiCanh(unsigned  int x)= 0;      virtual void thongTin()= 0;      virtual unsigned int chuVi()= 0;      virtual unsigned int dienTich()= 0;  };  // tính chất kế thừa  // lớp hinhCN kết thừa các tính chất của hình  // trong kế thừa lớp con có thể kế thừa các phương thức và thuộc tính trong phạm vi protected và public  class hinhCN: public hinh  {      public:      void thietLap() override      {          dai= 0;          rong= 0;      }      void thongTin() override      {          cout<<"chieu dai: "<<dai<<endl;          cout<<"chieu rong: "<<rong<<endl;      }      void thayDoiDoDaiCanh(unsigned  int x) override      {          ;      }      void thayDoiDoDaiCanh(unsigned  int x, unsigned  int y)  override      {          dai= x;          rong= y;      }      unsigned int chuVi()      {          return (dai+rong)\*2;      }      unsigned int dienTich()      {           return (dai\*rong);      }      protected:      unsigned  int dai;      unsigned  int rong;  };  class hinhVuong:public hinh  {      public:      void thietLap() override      {          chieuDaiCanh= 0;      }      void thongTin() override      {          cout<<"chieu dai canh: "<<chieuDaiCanh<<endl;      }  // tính chất đa hình  // trong class hỗ trợ tính chất đa hình  // các hàm có thể nạp chồng  // yêu cầu khác kiểu trong tham số đầu vào hoặc khác số lượng tham số đầu vào      void thayDoiDoDaiCanh(unsigned  int x,unsigned  int y) override      {          ;      }      void thayDoiDoDaiCanh(unsigned  int x) override      {         chieuDaiCanh= x;      }      unsigned int chuVi() override      {          return chieuDaiCanh\*4;      }      unsigned int dienTich() override      {           return (chieuDaiCanh\*chieuDaiCanh);      }      protected:      unsigned int chieuDaiCanh;  };  class chonPhepToan  {      private:      hinh \*hinh1;      hinhCN hinhCN1;      hinhVuong hinhVuong1;      public:      chonPhepToan(): hinh1(nullptr)      {          hinhCN1.thietLap();          hinhVuong1.thietLap();          cout<<"mac dinh cac phep toan bang 0!";      }      // tính chất đóng gói không thể thao tác trực tiếp đến các thuộc tính của phương thức      // phải thông qua các phương thức cho phép thay đổi thuộc tính      // khi các biểu thức trong phương thức thay đổi cũng không làm thay đổi kiểu phương thức không ảnh hưởng      void luaChonPhepToan()      {   uint8\_t chonHinh,chonChucNang;          while(1)          {              cout<<"chon hinh"<<endl;              cout<<"0: hinh vuong"<<endl;              cout<<"1: hinh chu nhat"<<endl;              cin>>chonHinh;              if(chonHinh)              hinh1=&hinhCN1;              else              hinh1=&hinhVuong1;              cout<<"chon chuc nang"<<endl;              cout<<"0: thay doi do dai"<<endl;              cout<<"1: thong tin"<<endl;              cout<<"2: chu vi "<<endl;              cout<<"3: dien tich"<<endl;              cin>>chonChucNang;              switch (chonChucNang)              {              case 0:                  unsigned int x,y;                  cout<<"nhap chieu dai canh 1: "<<endl;                  cin>>x;                  cout<<"nhap chieu dai canh 2: "<<endl;                  cin>>y;                  hinh1->thayDoiDoDaiCanh(x,y);                  hinh1->thayDoiDoDaiCanh(x);                  break;              case 1:                  hinh1->thongTin();                  break;              case 2:                  cout<<"chu vi cua hinh: "<<hinh1->chuVi()<<endl;                  break;              case 3:                  cout<<"dien tich cua hinh: "<<hinh1->dienTich()<<endl;                  break;              default:                  break;              }          }      }  };  int main()  {      chonPhepToan chonPhepToan1;      chonPhepToan1.luaChonPhepToan();      return 0;  } |