#include <stdio.h>

#include <iostream>

using namespace std;

class A {

protected:

int x = 0;

public:

virtual void positive(){x = 1; }

virtual void negative(){x = 0; }

int state() { return x; }

};

class B : virtual public A {

private:

int xx;

public:

int y = 1;

virtual void positive() {y = 1;}

virtual void negative() {y = 0;}

const int state() { return y; }

};

class C : virtual public A {

public:

int z = 1;

virtual void positive() {z = 1;}

virtual void negative() {z = 0;}

int state() {return z;}

};

class D : public B, public C {

public:

const int a = 2;

virtual void positive() {

cout << (x == y) << endl;

x == 1 ? ((C\*)this)->negative() : ((A\*)this)->positive();

y == 0 ? ((C\*)this)->negative() : ((B\*)this)->positive();

((C\*)this)->positive();

}

virtual void negative() {

cout << ((x + y) == 2) << endl;

((A\*)this)->negative();

y == 1 ? ((A\*)this)->positive() : ((B\*)this)->positive();

((C\*)this)->positive();

}

int state() {

return B::state() + C::state() + A::state();

}

};

//op1

// int main() {

// A a;

// B b;

// C c;

// D d;

// d.positive();

// d.negative();

// printf("%d\n", a.state());

// printf("%d\n", b.state());

// printf("%d\n", c.state());

// printf("%d\n", d.a);

// return 0;

// }

//op2

// int main() {

// A a;

// B b;

// C c;

// D d;

// for(;d.state() != 3; b.positive()) {

// switch(b.state()) {

// case 1:

// switch(c.state()) {

// case 1:

// cout << "Good job" << endl;

// break;

// default:

// b.positive();

// }

// c.negative();

// case 0:

// c.positive();

// break;

// default:

// b.y = 69;

// }

// }

// printf("%d\n", a.state());

// printf("%d\n", b.state());

// printf("%d\n", c.state());

// printf("%d\n", d.a);

// return 0;

// }

//op3

// int main() {

// A a;

// B b;

// C c;

// D d;

// if (a.x == 0) {

// cout << "===\n" << endl;

// }

// else if (b.y == 0) {

// cout << "Why tho" << endl;

// }

// printf("%d\n", a.state());

// printf("%d\n", b.state());

// printf("%d\n", c.state());

// printf("%d\n", d.a);

// return 0;

// }

int main() {

A a;

B b;

C c;

D d;

while(true) {

try {

if (d.state() == 2) {

throw d.state();

}

else if ((d.state() == 3) || (d.state() == 1)) {

d.positive();

}

else d.negative();

cout << d.state() <<endl;

}

catch(int x) {

break;

}

}

}