

Tell me the output of the program to the right. Put your answer all on one line -- if there's a newline in the program's output, simply replace it with a space, or even no space. Both will be fine.

So, for example, if the output is:

```
A
B
C
```

Then your answer should be "ABC" or "A B C".

```
#include <iostream>
using namespace std;

class Fred {
public:
    Fred();
    Fred(const Fred &f);
    ~Fred();
};

Fred::Fred()
{
    cout << "A" << endl;
}

Fred::Fred(const Fred &f)
{
    cout << "B" << endl;
}

Fred::~~Fred()
{
    cout << "C" << endl;
}

void proc1(Fred f, int i)
{
    cout << "X" << endl;
    if (i != 0) throw (string) "E";
}

void proc2(const Fred &f)
{
    cout << "Y" << endl;
}

int main()
{
    Fred f;
    Fred *f2;

    try {
        proc1(f, 0);
        proc2(f);
        f2 = new Fred;
        proc1(f, 1);
    } catch (const string &s) {
        cout << s << endl;
        return 0;
    }

    cout << "F" << endl;
    return 0;
}
```

Answer to the Clicker Question

- The very first thing that happens is the f's constructor is called, because it has been declared inside **main()**. That prints "A".
- Next "proc1(f,0)" is called. That calls the copy constructor for the parameter. It prints "B".
- "proc1(f,0)" prints "X".
- "proc1(f,0)" tries the if statement, which is false. It returns, calling the destructor for its parameter: "C".
- "proc2(f)" is called. It doesn't call a constructor, because its parameter f is a reference parameter. It prints "Y" and returns.
- "f2 = new Fred" is called. That calls the constructor for "Fred". It prints "A".
- "proc1(f,1)" is called. That calls the copy constructor for the parameter. It prints "B".
- "proc1(f,1)" prints "X".
- "proc1(f,1)" throws the exception, which makes the procedure return. Before returning it calls the destructor for its parameter f: "C".
- The exception is caught in the "catch" statement. That prints "E".
- The "catch" statement now returns, which calls the destructor for main's f. That prints "C".
- You'll note that the destructor for f2 is not called -- you would have to call "delete" for that.

The answer is "A B X C Y A B X C E C" or "ABXCYABXCEC".