#include <iostream>

using namespace std;

double balance;

void init() {

    balance = 0;

}

void deposit(double amount) {

    if (amount <= 0) {

        return;

    }

    balance += amount;

}

void withdraw(double amount) {

    if (amount <= 0) {

        return;

    }

    if (balance - amount < 0) {

        return;

    }

    balance -= amount;

}

#include <QtTest>

class banktest : public QObject {

    Q\_OBJECT

public:

    banktest();

    ~banktest();

private slots:

    void test\_deposit();

    void test\_deposit\_negative();

    void test\_withdraw();

    void test\_withdraw\_negative();

    void test\_withdraw\_overdraw();

};

banktest::banktest() {

}

banktest::~banktest() {

}

void banktest::test\_deposit() {

    init();

    deposit(100);

    QCOMPARE(balance, 100);

}

void banktest::test\_deposit\_negative() {

    init();

    deposit(-100);

    QCOMPARE(balance, 0);

}

void banktest::test\_withdraw() {

    init();

    deposit(100);

    withdraw(50);

    QCOMPARE(balance, 50);

}

void banktest::test\_withdraw\_negative() {

    init();

    deposit(100);

    withdraw(-50);

    QCOMPARE(balance, 100);

}

void banktest::test\_withdraw\_overdraw() {

    init();

    deposit(100);

    withdraw(10000);

    QCOMPARE(balance, 100);

}

QTEST\_APPLESS\_MAIN(banktest)

#include "tst\_banktest.moc"