

With sync:

```
#include <iostream>
#include <pthread.h>
#include <semaphore.h>
#include <unistd.h>
```

```
using namespace std;
```

```
int balance = 500;
int numReader = 0;
pthread_mutex_t mutex;
sem_t wrt;
```

```
void* credit_writer(void* wno) {
    int writerID = *((int*)wno);
    int amount = 500;
    sem_wait(&wrt);
    balance += amount;
    cout << "Writer " << writerID << " credited Rs-" << amount
        << ". Current balance: Rs-" << balance << endl;
    sem_post(&wrt);
    return nullptr;
}
```

```
void* debit_writer(void* wno) {
    int writerID = *((int*)wno);
    int amount = 100;
    sem_wait(&wrt);
    balance -= amount;
    cout << "Writer " << writerID << " debited Rs-" << amount
        << ". Current balance: Rs-" << balance << endl;
    sem_post(&wrt);
    return nullptr;
}
```

```
void* reader(void* rno) {
    int readerID = *((int*)rno);
    pthread_mutex_lock(&mutex);
    numReader++;
    if (numReader == 1)
        sem_wait(&wrt);
    pthread_mutex_unlock(&mutex);
```

```

cout << "Reader " << readerID << " reads balance: Rs-" << balance << endl;

pthread_mutex_lock(&mutex);
numReader--;
if (numReader == 0)
    sem_post(&wrt);
pthread_mutex_unlock(&mutex);
return nullptr;
}

int main() {
    pthread_t readers[10], writers[5];
    char operations[] = {'D', 'C', 'C', 'D', 'C'};
    int ids[10] = {1,2,3,4,5,6,7,8,9,10};

    pthread_mutex_init(&mutex, nullptr);
    sem_init(&wrt, 0, 1);

    for (int i = 0; i < 5; i++) {
        pthread_create(&readers[i], nullptr, reader, &ids[i]);
    }

    for (int i = 0; i < 5; i++) {
        if (operations[i] == 'C')
            pthread_create(&writers[i], nullptr, credit_writer, &ids[i]);
        else
            pthread_create(&writers[i], nullptr, debit_writer, &ids[i]);
    }

    for (int i = 5; i < 10; i++) {
        pthread_create(&readers[i], nullptr, reader, &ids[i]);
    }

    for (int i = 0; i < 10; i++) {
        pthread_join(readers[i], nullptr);
    }

    for (int i = 0; i < 5; i++) {
        pthread_join(writers[i], nullptr);
    }

    pthread_mutex_destroy(&mutex);
    sem_destroy(&wrt);
}

```

```
    return 0;  
}
```

initial balance: 500

Operations for 5 writers:

cpp

Copy

Edit

char operations[] = {'D', 'C', 'C', 'D', 'C'};

Output:

Reader 1 reads balance: Rs-500

Reader 2 reads balance: Rs-500

Reader 3 reads balance: Rs-500

Reader 4 reads balance: Rs-500

Reader 5 reads balance: Rs-500

Writer 1 debited Rs-100. Current balance: Rs-400

Writer 2 credited Rs-500. Current balance: Rs-900

Writer 3 credited Rs-500. Current balance: Rs-1400

Writer 4 debited Rs-100. Current balance: Rs-1300

Writer 5 credited Rs-500. Current balance: Rs-1800

Reader 6 reads balance: Rs-1800

Reader 7 reads balance: Rs-1800

Reader 8 reads balance: Rs-1800

Reader 9 reads balance: Rs-1800

Reader 10 reads balance: Rs-1800