

4B OS LAB-10 TASKS | MUHAMMAD DANISH | CS182019 | 4B

Example-01:

```
1  #include<iostream>
2  #include <thread>
3  #include <vector>
4
5  using namespace std;
6  // race condition example
7  class Wallet{
8  private:
9      int money=0;
10
11  public:
12      void addMoney(int amount){
13          for(int i=0; i<amount; i++){
14              money++;
15          }
16      }
17      int getMoney(){
18          return money;
19      }
20  };
21
22  int main(){
23      Wallet walletObj;
24      vector<thread> threads;
25      for(int i=0; i<5; ++i)
26          threads.push_back(thread(Wallet::addMoney, &walletObj, 10000));
27
28      for(int i=4; i>=0; i--){
29          threads[i].join();
30          cout<< walletObj.getMoney()<<endl;
31      }
32  }
```

```
49106
49106
49106
49106
49106
```

```
-----
Process exited after 0.07732 seconds with return value 0
Press any key to continue . . .
```

Example-02:

```
1  #include<iostream>
2  #include <thread>
3  #include <vector>
4  #include <mutex>
5
6  using namespace std;
7  //mutex example
8  class Wallet{
9      private:
10         int money=0;
11         mutex mutex;
12
13     public:
14         void addMoney(int amount){
15             mutex.lock();
16             for(int i=0; i<amount; i++){
17                 money++;
18             }
19             mutex.unlock();
20         }
21         int getMoney(){
22             return money;
23         }
24     };
25
26 int main(){
27     Wallet walletObj;
28     vector<thread> threads;
29     for(int i=0;i<5;++i)
30         threads.push_back(thread(Wallet::addMoney,&walletObj,10000));
31
32     for(int i=4; i>=0; i--){
33         threads[i].join();
34         cout<< walletObj.getMoney()<<endl;
35     }
36
37 }
```

```
50000
50000
50000
50000
50000
```

```
-----
Process exited after 0.07784 seconds with return value 0
Press any key to continue . . .
```

Example-03:

```
1  #include<iostream>
2  #include <thread>
3  #include <vector>
4  #include <mutex>
5
6  using namespace std;
7  //mutex lock_guard example
8  class Wallet{
9      private:
10         int money=0;
11         mutex m;
12
13     public:
14         void addMoney(int amount){
15             // mutex.Lock();
16             lock_guard<mutex> lockGuard(m);
17             for(int i=0; i<amount; i++){
18                 money++;
19             }
20             // mutex.unlock();
21         }
22         int getMoney(){
23             return money;
24         }
25     };
26
27 int main(){
28     Wallet walletObj;
29     vector<thread> threads;
30     for(int i=0; i<5; ++i)
31         threads.push_back(thread(Wallet::addMoney, &walletObj, 10000));
32
33     for(int i=4; i>=0; i--){
34         threads[i].join();
35         cout<< walletObj.getMoney()<<endl;
36     }
37
38 }
```

```
50000
50000
50000
50000
50000
```

Process exited after 0.07784 seconds with return value 0
Press any key to continue . . .