**Thread Management**

**Try-catch block**

**Program:**

#include<iostream>

#include<thread>

using namespace std;

void myfunc()

{

cout<<"Are we here yet?"<<endl;

};

int main()

{

int i;

thread t1(myfunc);

try{

for (i=0;i<10;i++)

{

cout<<"Message from main: "<<i+1<<endl;

}

}

catch(...){

t1.join();

throw;

}

t1.join();

return 0;

}

**Functor**

**Program:**

#include<iostream>

#include<thread>

using namespace std;

class F

{

public:

void operator()()

{

for(int i=0;i<10;i++)

{

cout<<"Starting at t1 "<<i<<endl;

}

}

};

int main()

{

F functor;

int i;

thread t1(functor);

try{

for (i=0;i<10;i++)

{

cout<<"Message from main: "<<i+1<<endl;

}

}

catch(...){

t1.join();

throw;

}

t1.join();

return 0;

}