

4B OS LAB 8 TASKS

Muhammad Danish | cs182019 | 4B

Task-1:

```
task1.cpp task2.cpp task3.cpp task4.cpp task5.cpp task6.cpp task7.cpp c
1 C:\Users\user\Desktop\Semester 4\OS Lab\Lab 08\Tasks\task1.cpp
2
3
4 using namespace std;
5
6
7 void threadFunction()
8 {
9     for(int i=0;i<5;i++)
10     {
11         cout<< "Thread using function pointer." <<endl;
12     }
13 }
14
15 int main()
16 {
17     thread threadObj(threadFunction);
18     cout<<"Display from main thead."<<endl;
19     threadObj.join();
20
21     cout<<"\nExit of main function."<<endl;
22     return 0;
23 }
```

```
Display from main thead.Thread using function pointer.
Thread using function pointer.
Thread using function pointer.
Thread using function pointer.
Thread using function pointer.

Exit of main function.

-----
Process exited after 0.06582 seconds with return value 0
Press any key to continue . . .
```

(Continued...)

Task-2:

```
task1.cpp task2.cpp task3.cpp task4.cpp task5.cpp task6.cpp task7.cpp q1.cpp
1  #include <iostream>
2  #include <thread>
3  using namespace std;
4  class DisplayThread
5  {
6      public:
7          void operator()()
8          {
9              for(int i=0;i<5;i++)
10             {
11                 cout<<"Thread using fucntion object."<<endl;
12             }
13         };
14
15
16 int main()
17 {
18     thread threadObj( ( DisplayThread() ) );
19     for(int i=0;i<5;i++)
20     {
21         cout<<"Display from main thread."<<endl;
22     }
23
24     cout<<"Waiting for thread to complete."<<endl;
25     threadObj.join();
26
27     cout<<"Exiting from main thread"<<endl;
28     return 0;
29 }
```

```
Display from main thread.Thread using fucntion object.
Thread using fucntion object.
Thread using fucntion object.
Thread using fucntion object.
Thread using fucntion object.
```

```
Display from main thread.
Display from main thread.
Display from main thread.
Display from main thread.
Waiting for thread to complete.
Exiting from main thread
```

```
-----
Process exited after 0.07343 seconds with return value 0
Press any key to continue . . .
```

(Continued...)

Task-3:

task1.cpp task2.cpp task3.cpp task4.cpp task5.cpp task6.cpp task7.cpp q1.

```
1  #include <iostream>
2  #include <thread>
3
4  using namespace std;
5  int main()
6  {
7      int x =9;
8      thread threadObj([]{
9          for(int i=0;i<5;i++)
10             {
11                 cout <<"thread using lambda fucntion."<<endl;
12             }
13         });
14
15     for(int i=0;i<5;i++)
16     {
17         cout<<"Display from main thread."<<endl;
18     }
19     threadObj.join();
20     cout<<"Exiting from main thread."<<endl;
21     return 0;
22 }
```

```
Display from main thread.thread using lambda fucntion.
thread using lambda fucntion.
thread using lambda fucntion.
thread using lambda fucntion.
thread using lambda fucntion.
```

```
Display from main thread.
Display from main thread.
Display from main thread.
Display from main thread.
Exiting from main thread.
```

```
-----
Process exited after 0.1314 seconds with return value 0
Press any key to continue . . .
```

(Continued...)

Task-4:

ask1.cpp task2.cpp task3.cpp task4.cpp task5.cpp task6.cpp task7.cpp

```
1  #include <iostream>
2  #include <string>
3  #include <thread>
4  using namespace std;
5  class DisplayThread
6  {
7  public:
8      void operator()(int x,string str){
9          cout<<"Passed Number = "<<x<<endl;
10         cout<<"Passed String = "<<str<<endl;
11     }
12 };
13
14 void threadFunction(int x,string str){
15     cout<<"Passed Number = "<<x<<endl;
16     cout<<"Passed String = "<<str<<endl;
17 }
18
19 int main(){
20     int x=10;
21     string str ="SAMPLE STRING.";
22     thread threadObj1(threadFunction,x,str);
23     thread threadObj2((DisplayThread()),x,str);
24     thread threadObj3([](int x,string str){
25         cout<<"Passed Number = "<<x<<endl;
26         cout<<"Passed String = "<<str<<endl;
27     },x,str);
28
29     threadObj1.join();
30     threadObj2.join();
31     threadObj3.join();
32 }
```

```
Passed Number = Passed Number = 10
Passed String = SAMPLE STRING.
10
Passed String = SAMPLE STRING.
Passed Number = 10
Passed String = SAMPLE STRING.

-----
Process exited after 0.2331 seconds with return value 0
Press any key to continue . . .
```

(Continued...)

Task-5:

```
task1.cpp task2.cpp task3.cpp task4.cpp task5.cpp task6.cpp task7.cpp q1.cpp q2.cpp
1  #include <iostream>
2  #include <thread>
3  using namespace std;
4
5  void thread_function()
6  {
7      cout<<endl<<"Inside Thread :: ID = "<<this_thread::get_id()<<endl;
8  }
9
10 int main()
11 {
12     thread threadObj1(thread_function);
13     thread threadObj2(thread_function);
14     if(threadObj1.get_id() != threadObj2.get_id())
15     {
16         cout<<endl<<"Both threads have different IDs"<<endl;
17     }
18     cout<<"From main thread :: ID of thread 1 = "<<threadObj1.get_id()<<endl;
19     cout<<"From main thread :: ID of thread 2 = "<<threadObj2.get_id()<<endl;
20     threadObj1.join();
21     threadObj2.join();
22     return 0;
23 }
24 }
```

```
Both threads have different IDs
From main thread :: ID of thread 1 = Inside Thread :: ID = 3
Inside Thread :: ID = 2
2
From main thread :: ID of thread 2 = 3

-----
Process exited after 0.08208 seconds with return value 0
Press any key to continue . . . █
```

(Continued...)

Task-6:

ask1.cpp task2.cpp task3.cpp task4.cpp task5.cpp task6.cpp tas

```
1  #include <thread>
2  #include <iostream>
3  using namespace std;
4  class Summation{
5      private:
6          int sum;
7          int limit;
8      public:
9          Summation(int lim){
10             limit=lim;
11             sum=0;
12         }
13
14         int getSum() const{
15             return sum;
16         }
17
18         void operator()(){
19             for (int i=1;i<=limit;i++)
20             {
21                 sum =sum+i;
22             }
23         }
24     };
25
26 int main(){
27     Summation sumHelper(10);
28     thread thread(ref(sumHelper));
29     thread.join();
30     cout<<"Sum : "<<sumHelper.getSum()<<endl;
31     return 0;
32 }
```

Sum : 55

Process exited after 0.06646 seconds with return value 0
Press any key to continue . . . █

(Continued...)

Task-7:

task1.cpp task2.cpp task3.cpp task4.cpp task5.cpp task6.cpp tasl

```
1  #include<iostream>
2  #include<thread>
3  using namespace std;
4  class Summation{
5  private:
6      int sum;
7      int limit;
8  public:
9      Summation(int lim){
10         limit=lim;
11         sum=0;
12     }
13     int getSum() const{
14         return sum;
15     }
16
17     void operator ()(){
18         for(int i=1;i<=limit;i++){
19             sum =sum+i;
20         }
21     }
22 };
23 int main()
24 {
25     Summation sumHelper(10);
26     thread thread(ref(sumHelper));
27     thread.detach();
28     cout<<"Sum : "<<sumHelper.getSum()<<endl;
29     return 0;
30 }
```

Sum : 0

Process exited after 0.1276 seconds with return value 0
Press any key to continue . . .