3 Process and Threads

3.1 Aim

Familiarization and implementation of programs related to Process and thread.

3.2 Theory

A thread is a construct used to execute different portions of a process simultaneously.They are light-weight and can be scheduled individually by the scheduler.The scheduler usually uses a prioritization scheme to determine the order of execution of the the threads. A process,on the other hand is a collection of threads.A process can have multiple threads managing different parts of the process concurrently.

3.3 Algorithm

Algorithm 1 ThreadCallBack Procedure

```
procedure ThreadCallBack(ID)                    ▷ The value of fork method
    threadId ← this_thread :: get_id()
    if id == 0 then
        print "This thread was created by child and thread id is " and threadId
    else
        print "This thread was created by parent and thread id is " and
threadId
    end if
end procedure
```

Algorithm 2 Main Procedure

```
procedure Main()                                          ▷ The main function
    threadNo ← fork()                                ▷ Spawns a child process
    thread1 ← thread(ThreadCallBack, threadNo)
    thread2 ← thread(ThreadCallBack, threadNo)
    thread1.join()                              ▷ Waits for thread1 to terminate
    thread1.join()                              ▷ Waits for thread2 to terminate
end procedure
```

3.4 Code

```
#include<thread>
#include<iostream>
#include<unistd.h>
using namespace std;
void threadcb(int id){
    if(id == 0)
        cout<<"This thread was created by child process and thread id is \n"<<
        this_thread::get_id()<<"\n";
    else
        cout<<"This thread was created by the parent process and thread id is
        \n"<<this_thread::get_id()<<"\n";

}

int main(){
    int threadno = fork();
    thread thread1(threadcb,threadno);
    thread thread2(threadcb,threadno);
    thread1.join();
    thread2.join();
    return 0;
}
```

3.5 Output

```
This thread was created by the parent process and thread id is
This thread was created by the parent process and thread id is
140364165363456
140364156970752
This thread was created by child process and thread id is
140364165363456
This thread was created by child process and thread id is
140364156970752
```

3.6 Result

Implemented a program to show the application of threads and processes in cpp and compiled using g++ version 8.2.1 on arch linux(kernel 4.20.6).The above output was obtained