MULTI –THREAD PROJECT REPORT

First of all, the multi-thread project in Linux with C language requires some operating system function in a library called “unistd.h” and the other library is “pthread.h”. We need to include these libraries to call some functions about multi-threading programs.

In this project, as our lecturer declared, we have to use the same object named “counter” in our 3 thread functions. We have to increment this object in every thread calling.

However, there is a big problem in this calling. If two thread wants to access this object at the same time, there going to be a big problem. Why has this problem come out? The reason for this, an incrementation consists of 3 commands. As I researched some information about it, these commands are MOV, INC, and MOV. This means that it cannot be an atomic command. İt is not certain. So we need to take some precautions to avoid this problem.

To avoid this problem, there is a structure to prevent this conflict. This method is ‘mutex’. When we create a shared object or shared writing area, before this code we need to lock it and after the end of the code, we need to unlock it. Therefore, there’s gonna be nothing problem to happen.

I needed to use an argument inside a thread. This argument should be sent with its reference or address. I searched the parameters of the pthread_create function and I found the last parametre of this function. The fourth parameter of the function can take all types from the users but it should be a pointer type. I really struggled at this point because the typecast from (void *) to (int *) was a good experience for me.

This sample is a screenshot from the code output: