Pthreads

Anandha Gopalan

November 24^{th} 2017

Imperial College
London

What are Pthreads?

- POSIX (Portable Operating System Interface) Threads
- Implementation depends on the system
- Implemented as a library on POSIX compliant Unix systems $\rightarrow$ libpthread
  - Since Kernel version 2.6, implemented in Linux as the Native POSIX Thread Library $\Rightarrow$ 1-1 relationship between threads created and Kernel threads
- Provides APIs for creating and managing threads

Creation

```
#include <pthread.h>

int pthread_create (pthread_t *thread, const pthread_attr_t *attr,
                    void *(*start_routine) (void *), void *arg)
```

pthread_create (ID, Attributes, Function Pointer, Argument)

- ID → ID of newly created thread (set by the system)
- Attributes → Generally set to NULL for default attributes
- Function Pointer → Pointer to the function which will be executed by the newly created thread
- Argument → Argument to the Function

Create and pass parameters (Example from CW)

Pass address of producerid so that we get id of thread created back

```
pthread_t producerid;
int parameter = 5;
pthread_create (&producerid, NULL, producer,
                (void *) &parameter);
```

Have to cast to correct data type in function

```
void *producer (void *parameter)
{
  int *param = (int *) parameter;
  cout << "Parameter = " << *param << endl;
  pthread_exit(0);
}
```

Create and pass parameters

How do we create multiple threads?

Hint: Using an array to hold multiple thread IDs

How do we pass multiple parameters?

Can use global variables

Hint: Using structures $\Rightarrow$ *struct* data type declaration and passing the structure as a parameter is a more elegant solution

Exit

Exit from Thread

```
#include <pthread.h>
void pthread_exit (void *retval)
pthread_exit (0);
```

Wait for thread to finish and exit main programme

```
#include <pthread.h>
int pthread_join (pthread_t thread, void **retval)
pthread_join (producerid, NULL);
```