#include <pthread.h>
#include <iostream>

using namespace std;

const int NUM_THREADS = 5;

void* PrintHello(void* threadid);

void* PrintHello(void* threadid) {
    long tid;
    tid = reinterpret_cast<long>(threadid);
    cout << "It's me big daddy thread " << tid << endl;
    pthread_exit(NULL);
}

int main (void) {
    pthread_t threads[NUM_THREADS];
    int rc;
    long t;
    for (t = 0; t < NUM_THREADS; t++) {
        cout << "Creating child " << t << endl;
        rc = pthread_create(&threads[t], NULL, PrintHello, reinterpret_cast<void*>(t));
        if(rc) {
            cerr << "ERROR: return code from pthread_create() is " << rc << endl;
            exit(-1);
        }
    }
    pthread_exit(NULL);  //won't exit until all children are dead
}

/*
pukeko:pthreads jsr$ ./1pthread
Creating child 0
Creating child 1
It's me big daddy thread 0
Creating child 2
It's me big daddy thread 1
Creating child 3
It's me big daddy thread 2
Creating child 4
It's me big daddy thread 3
It's me big daddy thread 4
pukeko:pthreads jsr$
*/