**/\* A simple child/parent signalling example. - main-signal.c \*/**

**#include <stdio.h>**

**#include <pthread.h>**

**int done = 0;**

**void\* worker(void\* arg) {**

**printf("this should print first\n");**

**done = 1;**

**return NULL;**

**}**

**int main(int argc, char \*argv[]) {**

**pthread\_t p;**

**pthread\_create(&p, NULL, worker, NULL);**

**while (done == 0)**

**;**

**printf("this should print last\n");**

**return 0;**

**}**

**/\***

**vishnu@mannava:~/threads$ cc main-signal.c -lpthread**

**vishnu@mannava:~/threads$ ./a.out**

**this should print first**

**this should print last**

**\*/**

**/\* A more efficient signaling via condition variables. - main-signal-cv.c \*/**

**#include <stdio.h>**

**#include <pthread.h>**

**// simple synchronizer: allows one thread to wait for another structure "synchronizer\_t" has all the needed data methods are:**

**// init (called by one thread)**

**// wait (to wait for a thread)**

**// done (to indicate thread is done)**

**typedef struct \_\_synchronizer\_t {**

**pthread\_mutex\_t lock;**

**pthread\_cond\_t cond;**

**int done;**

**} synchronizer\_t;**

**synchronizer\_t s;**

**void signal\_init(synchronizer\_t \*s) {**

**pthread\_mutex\_init(&s->lock, NULL);**

**pthread\_cond\_init(&s->cond, NULL);**

**s->done = 0;**

**}**

**void signal\_done(synchronizer\_t \*s) {**

**pthread\_mutex\_lock(&s->lock);**

**s->done = 1;**

**pthread\_cond\_signal(&s->cond);**

**pthread\_mutex\_unlock(&s->lock);**

**}**

**void signal\_wait(synchronizer\_t \*s) {**

**pthread\_mutex\_lock(&s->lock);**

**while (s->done == 0)**

**pthread\_cond\_wait(&s->cond, &s->lock);**

**pthread\_mutex\_unlock(&s->lock);**

**}**

**void\* worker(void\* arg) {**

**printf("this should print first\n");**

**signal\_done(&s);**

**return NULL;**

**}**

**int main(int argc, char \*argv[]) {**

**pthread\_t p;**

**signal\_init(&s);**

**pthread\_create(&p, NULL, worker, NULL);**

**signal\_wait(&s);**

**printf("this should print last\n");**

**return 0;**

**}**

**/\***

**vishnu@mannava:~/threads$ cc main-signal-cv.c -lpthread**

**vishnu@mannava:~/threads$ ./a.out**

**this should print first**

**this should print last**

**\*/**