The header:

struct absp {

int a;

int b;

int s;

int p;

};

Program A:

#include "header.h"

#include <stdlib.h>

#include <sys/ipc.h>

#include <sys/shm.h>

/\*program A : responsible for the creation and deletion of the shared memory\*/

int main() {

int shmid; /\*here we'll store the id, the key of the ipc\*/

int k = 0;

struct absp\* x;

/\*how to create a shared memory (an ipc)\*/

shmid = shmget(1234, sizeof(struct absp), IPC\_CREAT | 0600);

/\* 1234 : the id of the ipc which was specified explicitly it wasn't chosen by the system

\* !!!

\* If the id is already used it will result in an error or unexpected behaviour

\* -> make sure u use something unique

\* -> or create it first in the command line and use schmget(1234-the id, 0, 0) to access it

\* -> and delete it from the command line when you are done

\* !!!

\* sizeof(struct absp) : the size of the shared memory you want to create

\* IPC\_CREA : create the shared memory

\* 600 : the permission

\*/

/\*how to attach to the shared memory area - hoe to get the pointer to that memry area\*/

x = shmat(shmid, 0, 0);

/\* shmid : the id of the shared memory area

\* x : the pointer to the sharem memory area

\*/

while(1) {

x->a = k++ % 100;

x->b = k++ % 100;

if (x->p == x->s)

break;

}

/\*how to detach from the shared memory area\*/

shmdt(x);

/\*shmid : the id of the shared memory area\*/

/\*how to delete the shared memory area\*/

shmctl(shmid, IPC\_RMID, NULL);

/\* shmid : the id of the shared memory area

\* IPC\_RMID : delete the shared memory

\* NULL : the parameters for the function

\*/

return 0;

}

Program B:

#include "header.h"

#include <stdlib.h>

#include <sys/ipc.h>

#include <sys/shm.h>

/\*program B : only connects and then leaves\*/

int main() {

int shmid; /\*here we'll store the id, the key of the ipc\*/

struct absp\* x;

/\*how to connect to a shared memory (an ipc)\*/

shmid = shmget(1234, 0, 0);

/\* 1234 : the id of the shared memory area

\* 0 : we don't care about the size as it is already exists

\* 0 : we don;t create it nor do we give it permission

\*/

/\*how to attach to the shared memory area - hoe to get the pointer to that memry area\*/

x = shmat(shmid, 0, 0);

/\* shmid : the handle to the shared memory area

\* x : the pointer to the memory area

\*/

while(1) {

x->s = x->a + x->b;

x->p = x->a \* x->b;

if (x->p == x->s)

break;

}

/\*how to detach from the shared memory area\*/

shmdt(x);

/\*shmid : the handle to the shared memory area\*/

return 0;

}