

COSC 519
Spring 2014
Emanuel Rivera
Homework 3

Using Fork() and Exec() or Clone(), create **four** child processes. Load the Hello” program in each process after creation (same program). Each child program should print its own PID or some other parameters distinguishing its execution. Each process should run and return to the parent process where the parent is waiting to terminate. The parent should only terminate after all children returned from execution.

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/types.h>
#include <time.h>

#define BUF_SIZE 100
int retTime = 0;
void ChildProcess(void);      /* child process prototype */
void ParentProcess(void);    /* parent process prototype */

void waitFor (unsigned int secs) {
    retTime = time(0) + secs;  // Get finishing time.
    while (time(0) < retTime); // Loop until it arrives.
}

int main()
{
    pid_t pid;
    int i = 0;

    for(i = 0; i < 4; i++) {
        pid = fork();

        if(pid < 0) {
            printf("Error");
            exit(1);
        } else if (pid == 0) {
            ChildProcess();
        } else {
            ParentProcess();
        }
    }
}
```

```
}
```

```
void ChildProcess(void)
```

```
{
```

```
    char  buf[BUF_SIZE];
```

```
    sprintf(buf, "Hello World from child! from pid %d, value = %d\n", getpid(), 1);
```

```
    write(1, buf, strlen(buf));
```

```
    sleep(1);
```

```
    exit(0);
```

```
}
```

```
void ParentProcess(void)
```

```
{
```

```
    char  buf[BUF_SIZE];
```

```
    wait(NULL);
```

```
    sprintf(buf, "Hello World from Parent ! from pid %d, value = %d\n", getpid(), 1);
```

```
    write(1, buf, strlen(buf));
```

```
}
```

```
/*
```

```
 * Output:
```

```
Hello World from child! from pid 15403, value = 1
```

```
Hello World from Parent ! from pid 15402, value = 1
```

```
Hello World from child! from pid 15404, value = 1
```

```
Hello World from Parent ! from pid 15402, value = 1
```

```
Hello World from child! from pid 15405, value = 1
```

```
Hello World from Parent ! from pid 15402, value = 1
```

```
Hello World from child! from pid 15406, value = 1
```

```
Hello World from Parent ! from pid 15402, value = 1
```

```
*/
```