## 需要的头文件：

#include <sys/types.h>

#include <unistd.h>

## pid\_t fork(void);

功能：

用于从一个已存在的进程中创建一个新进程，新进程称为子进程，原进程称为父进程。

参数：

无

返回值：

成功：子进程中返回 0，父进程中返回子进程 ID。pid\_t，为无符号整型。

失败：返回 -1。

失败的两个主要原因是：

1）当前的进程数已经达到了系统规定的上限，这时 errno 的值被设置为 EAGAIN。  
2）系统内存不足，这时 errno 的值被设置为 ENOMEM。

测试示例如下：

#include <stdio.h>

#include <unistd.h>

#include <sys/types.h>

int main(int argc, char \*argv[])

{

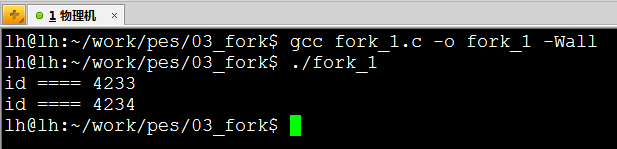
    fork();

    printf("id ==== %d\n", getpid());   // 获取进程号

    return 0;

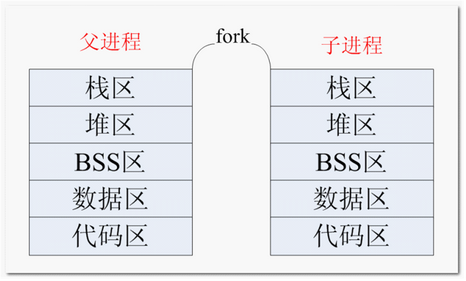
}

运行结果：



## 那子进程长什么样的呢？

使用 fork() 函数得到的子进程是父进程的一个复制品，子进程从父进程复制了一切。因此，使用 fork() 函数的代价是很大的。



测试代码：

#include <stdio.h>

#include <unistd.h>

#include <sys/types.h>

int main(int argc, char \*argv[])

{

    pid\_t pid;

    pid = fork();

    if( pid < 0 ){  // 没有创建成功

        perror("fork");

    }

    if(0 == pid){ // 子进程

        while(1){

            printf("I am son\n");

            sleep(1);

        }

    }else if(pid > 0){ // 父进程

        while(1){

            printf("I am father\n");

            sleep(1);

        }

    }

    return 0;

}