

```

/*
                                /      r
                               /
                              a \
                               \  f
                               /  \
                              e    l

*/

#include <stdio.h>
#include <sys/types.h>
#include <unistd.h>
#include <stdlib.h>
#include <sys/wait.h>
#include <string.h>

int main (void){
    pid_t childpid = 0;
    int status;
    printf("proc-R, pid %d, ppid %d, acaba de ser criado\n", getpid(), getppid());

    childpid = fork();
    if(childpid == 0){
        printf("proc-A, pid %d, ppid %d, acaba de ser criado\n", getpid(), getppid());
        childpid = fork();

        if(childpid == 0){
            printf("proc-F, pid %d, ppid %d, acaba de ser criado\n", getpid(),
getppid());
            childpid = fork();

            if(childpid == 0){
                printf("proc-E, pid %d, ppid %d, acaba de ser criado\n",
getpid(), getppid());
                printf("proc-E, pid %d, e morreu\n", getpid());
                exit(0);
            }
            else{
                waitpid(childpid, &status, 0);
                childpid = fork();
                if(childpid == 0){
                    printf("proc-L, pid %d, ppid %d, acaba de ser criado\n",
getpid(), getppid());
                    printf("proc-L, pid %d, e morreu\n", getpid());
                    exit(0);
                }
                else{
                    waitpid(childpid, &status, 0);
                    printf("proc-F, pid %d, e morreu\n", getpid());
                    exit(0);
                }
            }
        }
        else{
            waitpid(childpid, &status, 0);
            printf("proc-A, pid %d, e morreu\n", getpid());
            exit(0);
        }
    }
    else{
        waitpid(childpid, &status, 0);
        printf("proc-R, pid %d, e morreu\n", getpid());
        exit(0);
    }
}

```

