

# 第八章 异常控制流 家庭作业

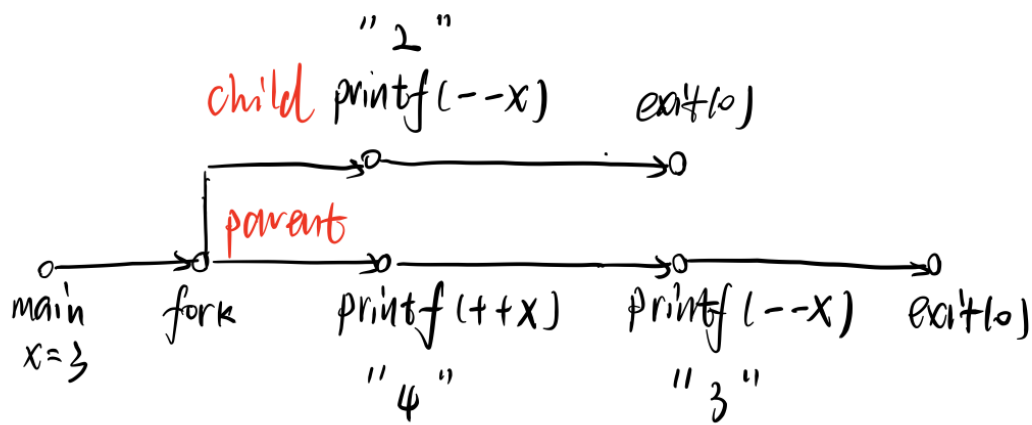
——1170300825 李大鑫

## 8.9

进程对	并发地?
AB	否
AC	是
AD	是
BC	是
BD	是
CD	是

## 8.13

该程序的进程图如下：

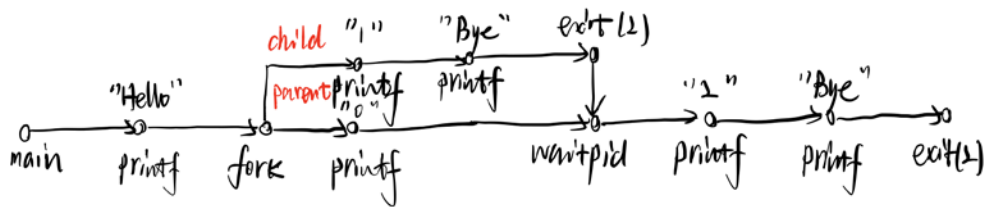


由进程图可以得出，只要满足 4 在 3 之前的序列都是可能的输出，所有的可能输出为（下面答案中的空格在实际输出中是换行）：

- 1) 4 3 2
- 2) 2 4 3
- 3) 4 2 3

## 8.17

列举练习题 8.4 中程序所有可能的输出。

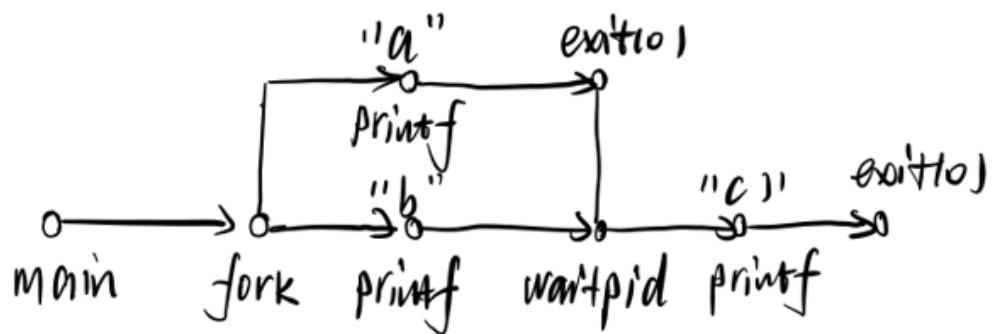


由上面的进程图可以得出，程序的所有可能输出如下：

- 1) Hello 0 1 Bye 2 Bye
- 2) Hello 1 0 Bye 2 Bye
- 3) Hello 1 Bye 0 2 Bye

## 8.21

该程序的进程图如下：



由进程图可以得出，所有的输出情况如下：

- 1) abc
- 2) bac

## 8.25

分析：

首先由“5 秒钟之后会超时”的要求可以想到我们之前学过的 SIGALRM 信号以及 alarm 函数，alarm(secs)函数的作用是在 secs 秒之后向调用进程发送一个 SIGALRM 信号。接收到

SIGALRM 信号之后，我们处理逻辑应该是在 tfgets 函数中返回空值 NULL；如果在 5 秒钟之内我们在屏幕上成功获得输入，则应该在 tfgets 函数返回调用 fgets 函数从 stdin 读入相应输入。

如何从 SIGALRM 处理函数中跳转到 tfgets 函数中我们可以想到非本地跳转。利用 setjmp 的性质：在 longjmp 返回之前 setjmp 值一直为 0，longjmp 可以返回指定返回值。所以对 setjmp 进行 if 判断，如果为 0 代表 longjmp 没有调用，此时我们调用 fgets 进行读入，如果读入成功则 return，此时 SIGALRM 处理函数也就没有了回调的机会；为 1 时，代表 SIGALRM 处理函数被调用，此时返回 NULL。

代码如下：

```
//
// Created by DaxinLi on 2018/12/10.
//
#include <stdio.h>
#include <signal.h>
#include <unistd.h>
#include <setjmp.h>

#define TIMELIMIT ((unsigned int)5)
#define SIZEOFBUF 1024

jmp_buf buf;

void SIGALRM_handler(int signum) {
    longjmp(buf, 1);    //1是setjump函数的返回值
}

char *tfgets(char *s, int size, FILE *stream)
{
    if (signal(SIGALRM, SIGALRM_handler) == SIG_ERR) { //设置SIGALRM的
信号处理函数
        perror("ERROR WHILE SET SIGALRM RECALL FUNC");
        return NULL;
    } else {
        alarm(TIMELIMIT); //在TIMEOUT时间 之后 向调用进程发送一个SIGALRM信号
触发信号处理函数。
    }

    if (!setjmp(buf)) { //利用非本地跳转 buf中存放运行上下文
        return fgets(s, size, stream);
    } else {
        return NULL;
    }
}
```

```

}

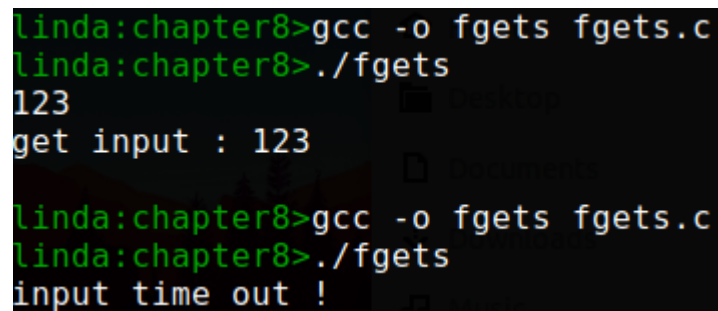
int main(int argc, char const *argv[])
{
    char temp_bufer[SIZEOFBUF];
    char *result = tfgets(temp_bufer, SIZEOFBUF, stdin);

    if (result) {
        printf("get input : %s\n", result);
    } else {
        fprintf(stderr, "input time out !\n");
    }

    return 0;
}

```

代码测试:



```

linda:chapter8>gcc -o fgets fgets.c
linda:chapter8>./fgets
123
get input : 123

linda:chapter8>gcc -o fgets fgets.c
linda:chapter8>./fgets
input time out !

```