

第八章

8.11

这个会输出4个hello输出行。

最开始的父进程会打印一个helloworld；在父进程下， $i = 0$ 时会fork一个新的分支，会打印一个Helloworld， $i = 1$ 时会fork一个新的分支，打印一个Helloworld；在 $i = 0$ 的子分支下，会再次fork一个子分支的子分支，打印一个Helloworld，共四次。

8.15

共5个Hello

父进程打印一个hello，在执行doit（）时，在判断时会fork一个子进程1，子进程的返回值为0，故进入判断的执行部分，再fork一个子进程2，打印一个hello，执行doit完毕后打印子分支1的main中的Hello。子进程2会执行与子进程1从判断的下一行开始同样的流程，故也打印2个hello。共5个。

8.19

2^n 个。

最初的父进程会输出一个hello。

其余的，对于 $i = k$ 时的子进程，会输出 $(n-k)$ 个hello。

共计 $sum = n + (n-1) + (n-2) + \dots + 1 + 1 = 2^n$ 。

8.23

在同一时间，只会有一个正在等待的信号，其余相同的信号会被取消。如果要消除这种现象，可以将

`sleep (1) ;`

语句去掉。