

Shell Lab实验报告

eval函数

1.使用parseline函数解析命令行，如果命令行为空直接返回，之后使用builtin_cmd函数判断所得命令是否为内置命令，如果是内置命令则直接执行后返回。
2.如果不是内置命令，那么先阻塞SIGINT、SIGTSTP、SIGCHLD信号，然后调用fork创建子进程。在子进程中我们解除信号阻塞并设置id，然后调用execve函数来执行命令。
3.父进程最先调用addjob函数将子进程job加入链表中，然后解除阻塞，判断子进程是否在前台运行，若在前台运行则调用waitfg函数等待运行完成，否则其后台工作，就打印进程组jid和子进程pid以及相应的命令行字符串。

builtin_cmd函数

取argv[0]为命令行参数cmd。
1.当cmd为quit时，直接终止tsh。
2.当cmd为jobs时，调用listjobs函数。
3.当cmd为bg或fg时，调用do_bgfg函数执行相应的内置bg和fg命令。
4.其他情况返回0交给eval函数继续处理。

do_bgfg函数

令cmd为argv[0]，arg为argv[1]。
1.判断arg是否带%，若不带则说明传入了pid，若带则说明传入了jid。接着调用getjobxid函数来获得对应的job结构体，注意判断返回结构是否为空。
2.使用strcmp函数判断cmd为bg命令还是fg命令：
若为bg，则使目标进程重新开始工作，设置BG状态，打印进程信息。
若为fg，则使目标进程重新开始工作，设置FG状态，等待进程结束。

waitfg函数

首先设置屏蔽SIGCHLD信号，然后设置一个全局原子变量sign=1，通过while(sign)循环合理调用sigsuspend函数不断循环直至前台进程结束（此时sign在sigchld_handler函数中被置为0），再解除屏蔽返回即可。

sigchld_handler函数

首先用while循环调用waitpid直到它所有的子进程终止，接着检查已回收子进程的退出状态，分别用WSTOPSIG，WTERMSIG，WEXITSTATUS函数来判断以上三个退出状态并进行相应处理，最后输出处理结果。

sigint_handler函数

首先调用函数fgpid返回前台进程curpid，如果curpid不为0，那么调用kill函数发送SIGINT信号给前台进程组，kill返回-1则输出错误表示发送失败。

sigtstp_handler函数

具体操作与sigint_handler类似，用fgpid(jobs)获取前台进程curpid，如果curpid不为0，那么调用kill函数发送SIGTSTP信号给前台进程组，kill返回-1则输出错误表示发送失败。

运行结果

经过人工测试可知，本程序最终通过了16个测试点，完成了所要求。