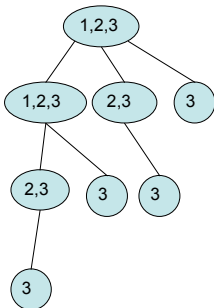


CS 537 Section 5: Quiz 2

Michael Swift

Quiz 2

Quiz 1



```
/* fork a child process */  
  
int val = 1;  
fork();  
printf("Val = %d\n",val);  
/* fork another child process */  
val = 2;  
fork();  
printf("Val = %d\n",val);  
/* and fork another */  
val = 3;  
fork();  
printf("Val = %d\n",val);
```

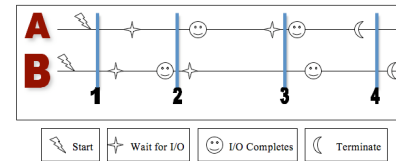
Project 1 questions?

- Sample code is available under `~cs537-1/projects/p0-sample`

Question 2: what to store

- Hardware state = data stored in hardware
 - Stack
 - Program counter
 - mode bit
- Process state is not in hardware; it is in PCB
- Hardware support?
 - Yes: to store PC
 - these get replaced with a PC of the interrupt handler

Question 3



1. A:Ready, B:Running, Ready:A, Waiting:
2. A:waiting, B:Running, Ready:, Waiting:A
3. A:waiting, B:waiting, Ready:, Waiting:A,B
4. A:terminated, B:running, Ready:, Waiting:

Questions on project?

- Hints are on web page for:
 - waiting for all processes to terminate before exiting shell
- TA provided sample shell scripts to test with
- Tricky Problems:
 - using strtok, how do you tell if the line ends in an '&'?
 - Suggestion:


```
int c,j = strlen(command);
end_in_amp = 0;
for (c = j-1; c > 0; c++)
    if (command[c] == '\n' || command[c] == '\t' || command[c] == ' ')
        continue;
    else if (command[c] == '&')
        ends_in_amp = 1;
break;
```

Handling built-ins correctly

- Basic structure (ignoring &)


```
read command line
parse command line
if (strcmp(command,"exit") == 0)
    exit(1);
else if (strcmp(command,"pwd") == 0)
    getpwd()
else if (strcmp(command,"cd") == 0)
    chdir
else {
    fork
    exec
}
```

Handling '&' correctly

- Read in command line
 - a ; b ; c & d ; e ; f
- char * tmp, *command, *next_command;
- command = strtok_r(line, "&", &tmp)
- if (command != NULL) {
 - do {
 - next_command = strtok(NULL, "&", &tmp);
 - if (next_command == NULL) && (!ends_with_ampersand)
 - execute_command_list(command);
 - else
 - pid = fork()
 - if (pid == 0)
 - » execute_command_list (command);
 - command = next_command;
 - } while (command != NULL);