OS homework 2nd

Process

1. Describe the actions taken by a kernel to context-switch between processes.

2. 采用下述程序，确定 A、B、C、D 四行中 pid 和 pid1 的值。（假设父进程和子进程的 pid 分别为 2600 和 2603）

```
#include <sys/types.h>
#include <stdio.h>
#include <unistd.h>

int main()
{
   pid_t pid,pid1;

   pid=fork();

   if (pid<0) {
     fprintf(stderr,"fork fail");
     return 1;
   }
   else if (pid==0) {
       pid1=getpid();
       printf("child:pid=%d",pid);              //A
       printf("child:pid1=%d",pid1);          //B
    }
    else{
       pid1=getpid();
       printf("parent:pid=%d",pid);            //C
       printf("parent:pid1=%d",pid1);        //D
       wait(NULL);
    }
   return 0;
}
```

Thread

1. Discuss the difference between user-level thread and kernel level thread.

2. Which of the following components of program state are shared across threads in a multithreaded process?
   A. Register values
   B. Heap memory
   C. Global variables
   D. Stack memory

3. The program shown in Figure 4.11 uses the Pthreads API. What would be output from the program at LINE C and LINE P?

```
include <stdio.h>
#include <pthread.h>

int value=0;
void *runner(void *param);      /* the thread */

int main()
{
   int pid;
   pthread_t tid;
   pthread_attr_t attr;
   pid=fork();
   if(pid==0) {
      pthread_attr_init(&attr);
      pthread_create(&tid, &attr, runner, NULL);
      pthread_join(tid, NULL);
      printf("CHILD: value=%d\n", value);     /* LINE C */
   }
   else if(pid>0) {
      wait(NULL);
      printf("PARENT: value=%d\n",value);    /* LINE P */
   }
}

void *runner(void *param)
{
   value=5;
   pthread_exit(0);
}
```