

CS 537 Section 9 Programming Assignment 4

Michael Swift

© 2004-2007 Ed Lazowska, Hank Levy, Andrea and
Remzi Arpaci-Dusseau, Michael Swift

1

Quiz 6 answers

- Raid 10 vs Raid 5
 - Single block: on Raid 10 you get a choice of two disks, so it can be faster
 - Multiple blocks: you can read from all disks with both
 - On read, RAID assumes the disk detects an error so parity/extra disk aren't read for every block.
- AFS vs NFS
 - Reads always see the data just written on the same machine.
 - With AFS, simultaneous writes are not mingled. With NFS, they may be

© 2004-2007 Ed Lazowska, Hank Levy, Andrea and
Remzi Arpaci-Dusseau, Michael Swift

2

P4

- Due date: Thursday 4/30
- Goal: write a multithreaded program
- Part 1: test the performance of thread create vs fork
- Part 2: write code to use threads

© 2004-2007 Ed Lazowska, Hank Levy, Andrea and
Remzi Arpaci-Dusseau, Michael Swift

3

Creating threads with pthreads

- pthreads is the official API for threads on Unix
- creating a thread:


```
int pthread_create(pthread_t * thread,
                  pthread_attr_t * attr,
                  void (*start_routine)(void *),
                  void * arg);
```
- thread = receives thread identifier
- attr = optional attributes, pass as NULL
- start_routine = function to call
- arg = argument to pass

© 2004-2007 Ed Lazowska, Hank Levy, Andrea and
Remzi Arpaci-Dusseau, Michael Swift

4

Example

```

void * sample_thread(void * arg)
{
    char * param = (char *) arg;
    printf("This thread reads %s\n",param);
    return(NULL);
}

int main(int argc, char * argv[])
{
    int result;
    pthread_t thd1;

    result = pthread_create(&thd1, NULL, sample_thread, "Hello");
    pthread_join(thd1, NULL);
    return(0);
}

```

© 2004-2007 Ed Lazowska, Hank Levy, Andrea and
Remzi Arpaci-Dusseau, Michael Swift

5

Waiting for a thread

- pthread_join() waits for a thread to terminate
- int pthread_join(pthread_t thread, void **value_ptr);
- The return from the thread is returned as the value

```

char test_string[] = "Test";

void * sample_thread(void * arg)
{
    char * param = (char *) arg;
    printf("This thread reads %s\n",param);
    return(test_string);
}

pthread_join(thd1, (void **) & res_string1);
printf("Thread 1 returned %s\n",res_string1);

```

© 2004-2007 Ed Lazowska, Hank Levy, Andrea and
Remzi Arpaci-Dusseau, Michael Swift

6

Compiling threads

- #include <pthread.h>
- gcc -pthread file.c

© 2004-2007 Ed Lazowska, Hank Levy, Andrea and
Remzi Arpaci-Dusseau, Michael Swift

7