pthread_create

pthread_join

pthread_exit

#include <pthread.h>

int pthread_create(pthread_t *thread**,**
pthread_attr_t *attr**,**
void *(start_routine**) (void *),**
void *arg**);**

int pthread_join(pthread_t thread**, void** **retval**);**

void pthread_exit(void *retval**);**

Compile and link with -pthread.

#1 My program calls pthread_create twice. How many stacks does my process have?

#2 What is the difference between a process and a thread?

#3 What does pthread_cancel do?
and are there alternatives?

#4 Differences between exit() and pthread_exit()?

..so why would you call pthread_exit in your main method?
#6 Give four ways that a thread can be terminated

#9 Why are some functions e.g. asctime,getenv, strtok, strerror not thread-safe?

```
char* to_message(int num) {
  char static result [256];
  if(num < 1000)
    sprintf(result, "%d : blah blah" , num);
  else strcpy(result, "Unknown");
  return result;
}
```

#10. What are condition variables, semaphores, mutexes?

#11. Advantages of threads over forking processes?

#12. Can you fork a process with multiple threads?

#13. Examples of why you might fork processes instead of using threads