İşletim Sistemleri Uygulama 4 İplikler

Bilgisayar Mühendisliği

İstanbul Teknik Üniversitesi 34469 Maslak, İstanbul

March 4, 2011





Bugün

İşletim Sistemleri Uygulama 4 İplik Yaratma İplik Yaratma, Birleştirme, Sonlandırma İpliklerle Global Değişkenlerin Kullanımı





İplik Yaratma

```
#include <pthread.h>
int pthread_create(pthread_t *thread, const pthread_attr_t *attr, void
*(*start_routine)(void*), void *arg);
```

```
pthread_t *thread: Yaratılacak ipliğe gösterge
const pthread_attr_t *attr: Yaratılacak ipliğin özelliklerine gösterge
void *(*start_routine)(void*): İpliği başlatacak yordama gösterge
void *arg: Başlangıç yordamının parametrelerine gösterge
```





Örnek bir program

```
void print_message_function( void *ptr );
int main()
        pthread_t thread1, thread2;
        char *message1 = "Hello";
        char *message2 = "World":
        pthread_create( &thread1, pthread_attr_default,
            (void*)&print_message_function , (void*) message1);
                                                                              11
        pthread_create(&thread2, pthread_attr_default,
            (void*)&print_message_function , (void*) message2);
        exit (0);
                                                                              16
void print_message_function ( void *ptr )
        char *message;
        message = (char *) ptr;
                                                                              21
        printf("%s ", message);
```





BLG 312

İplikli program derleme

- Kaynak dosyasi: kaynak.c
- ► Çalıştırılabilir dosya: cikti
- ▶ Bu uygulamaların çalıştırılabilmesi için pthread kitaplığına bağlanmaları gereklidir. Bu nedenele şu şekilde derlenmeliler:

gcc -D_POSIX_C_SOURCE -lpthread kaynak.c -o cikti





```
#define _GNU_SOURCE
#define _REENTRATNT
                                                                                 2
#include <stdio.h>
#include < stdlib .h>
#include <pthread.h>
#include <sys/types.h>
#include < sys/time.h>
                                                                                 7
#include < unistd.h>
#define MAX 5
int my_rand(int,int);
                                                                                 12
void *say_it(void *);
int main(int argc, char* argv[]){
        pthread_t thread_id [MAX];
        int status, * p_status=&status;
                                                                                 17
        int i;
        setvbuf(stdout,(char*)NULL,_IONBF,0);
        if (argc>MAX+1){
                                                                                 22
                 fprintf(stderr, "%s arg1, arg2, ... arg%d\n", argv[0], MAX);
                 return 1;
        printf("Displaying\n");
                                                                                27
        for (i=0; i < argc-1; ++i){
                 if ( pthread_create(&thread_id[i], NULL, say_it , (void *)argv[i+1])>0){
                          fprintf(stderr, "pthread_create failure\n");
                         return 2;
                                                                                 32
                 }
```









```
void* say_it(void* word){
                                                                                1
        int i:
        int numb=my_rand(2,6);
        printf("%s\t to be printed %d times\n",(char*)word,numb);
        sleep(1);
        for (i = 0; i < numb; i++){
                sleep (1);
                                                                                11
                 printf("%s ",(char*)word);
        sleep (4);
        return (void*)NULL;
                                                                                16
int my_rand(int start, int range){
        struct timeval t;
        gettimeofday(&t,(struct timezone *)NULL);
        return (int) (start+(float)range *
                                                                                21
                         rand_r((unsigned *)&t.tv_usec))/(RAND_MAX+1.0));
```





Örnek Çıktısı

```
ovatman@susa:/$ ./cikti 1 2 3 4 5
Displaying
         to be printed 7 times
         to be printed 2 times
         to be printed 6 times
         to be printed 3 times
         to be printed 6 times
1 3 2 5 4 5 1 4 2 3 1 3 4 5 5 3 1 3 1 5 1 3 5 1
Thread 3077884784 returns 0
Thread 3069492080 returns 0
Thread 3061099376 returns 0
Thread 3052706672 returns 0
Thread 3044313968 returns 0
Done
```

5

10





```
#include <pthread.h>
#include < stdlib .h>
#include < unistd.h>
#include <stdio.h>
                                                                                  5
int myglobal;
void* thread_function(void *arg){
         int i.j:
         for (i=0; i<20; i++){
                                                                                  10
                 j=myglobal;
                 i = i + 1;
                 //myglobal++;
                  printf(".");
                  fflush (stdout);
                                                                                  15
                  sleep (1);
                  myglobal=i:
         return NULL:
                                                                                  20
int main(void){
         pthread_t mythread;
         int i:
                                                                                  25
         if ( pthread_create(&mythread, NULL, thread_function, NULL)) {
                  printf("error creating thread");
                  abort();
```





BLG 312

3

```
for( i=0;i<20;i++){
    myglobal=myglobal+1;
    //myglobal++;
    printf("o");
    fflush (stdout);
    sleep(1);
}

if( pthread_join(mythread ,NULL)){
    printf("error joining thread");
    abort();
}

printf("\nmyglobal equals %d\n",myglobal);
exit(0);</pre>
15
```

```
ovatman@susa:/$ ./cikti
o.o..o.o.o.o.o.o.o.o.o.o.o.o.o.o.o
myglobal equals 21
```



