|  |  |
| --- | --- |
| #include <stdio.h>  #include <pthread.h>  #include <semaphore.h>  #include <stdlib.h>  #include <fcntl.h>  #include <unistd.h>  #include <sys/types.h>  #include <sys/mman.h>  #include <string.h>  #include <errno.h>  sem\_t mutex;  int c=0;  char\* data;  void\* thread1(char\* arg[])  {  int i,j,k=0;  for(i=0;i<=strlen(data)/2;i++)  {  if(data[i]==arg[2][0])  {  for(j=0;j<strlen(arg[2]);j++)  {  if(data[i+j]==arg[2][j])  k++;  }  if(k==strlen(arg[2]))  {  sem\_wait(&mutex);  c++;  sem\_post(&mutex);  }  k=0;  }  }  }  void\* thread2(char\* arg[])  {  int i,j,k=0;  for(i=1+strlen(data)/2;i<=strlen(data);i++) | {  if(data[i]==arg[2][0])  {  for(j=0;j<strlen(arg[2]);j++)  {  if(data[i+j]==arg[2][j])  k++;  }  if(k==strlen(arg[2]))  {  sem\_wait(&mutex);  c++;  sem\_post(&mutex);  }  k=0;  }  }  }  int main(int argc, char\*argv[])  {  sem\_init(&mutex, 0, 1);  pthread\_t t1,t2;  int fd;  if((fd = open(argv[1],O\_RDONLY)) == -1)  {  perror("open");  exit(1);  }  data = mmap(NULL, getpagesize(), PROT\_READ,MAP\_SHARED, fd,0);  pthread\_create(&t1,NULL,thread1,argv);  sleep(2);  pthread\_create(&t2,NULL,thread2,argv);  pthread\_join(t1,NULL);  pthread\_join(t2,NULL);  sem\_destroy(&mutex);  printf("Count= %d\n",c);  return 0;  } |



