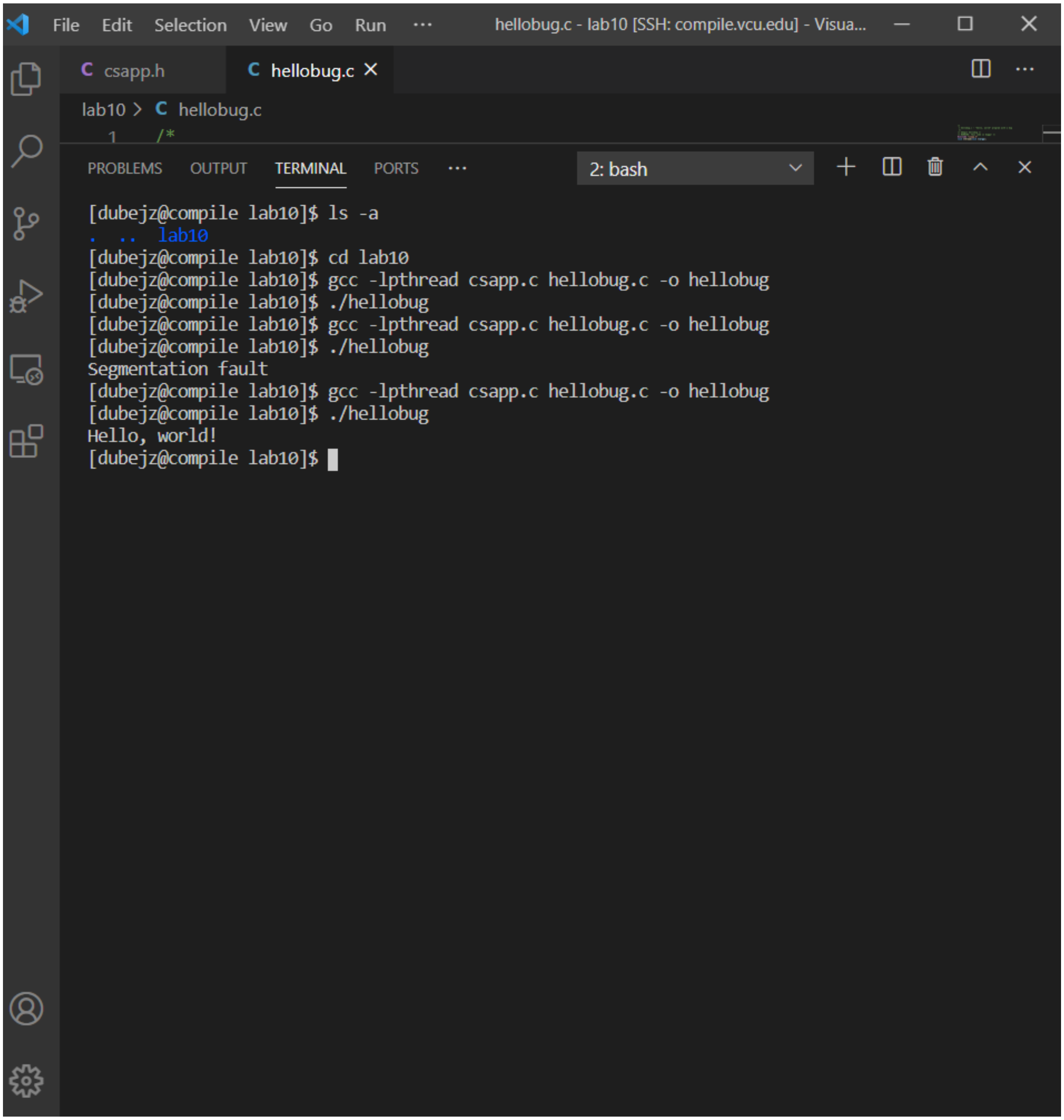
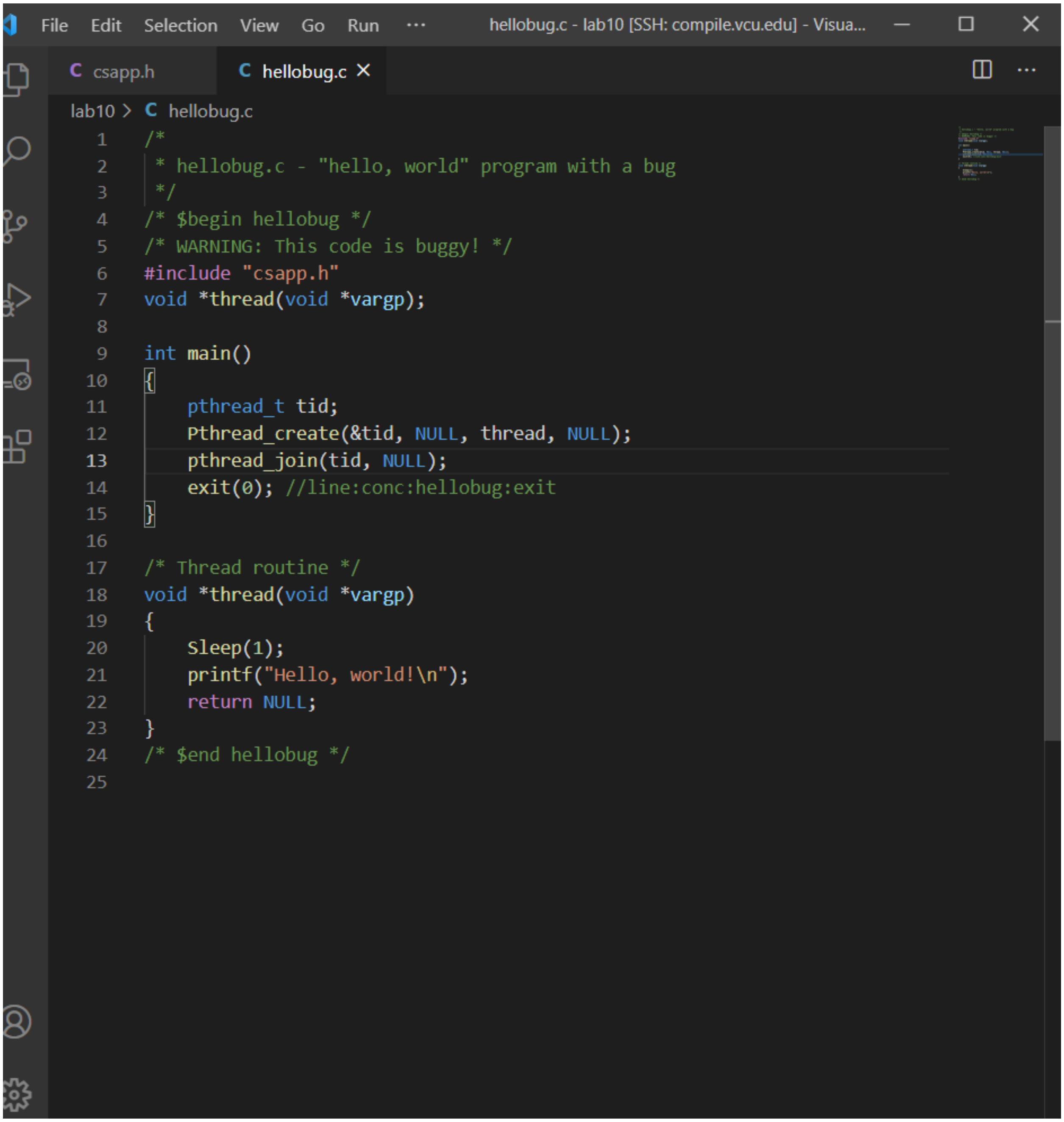
Jordan Dube
CMSC257
 Lab 10
Screenshots

1a)



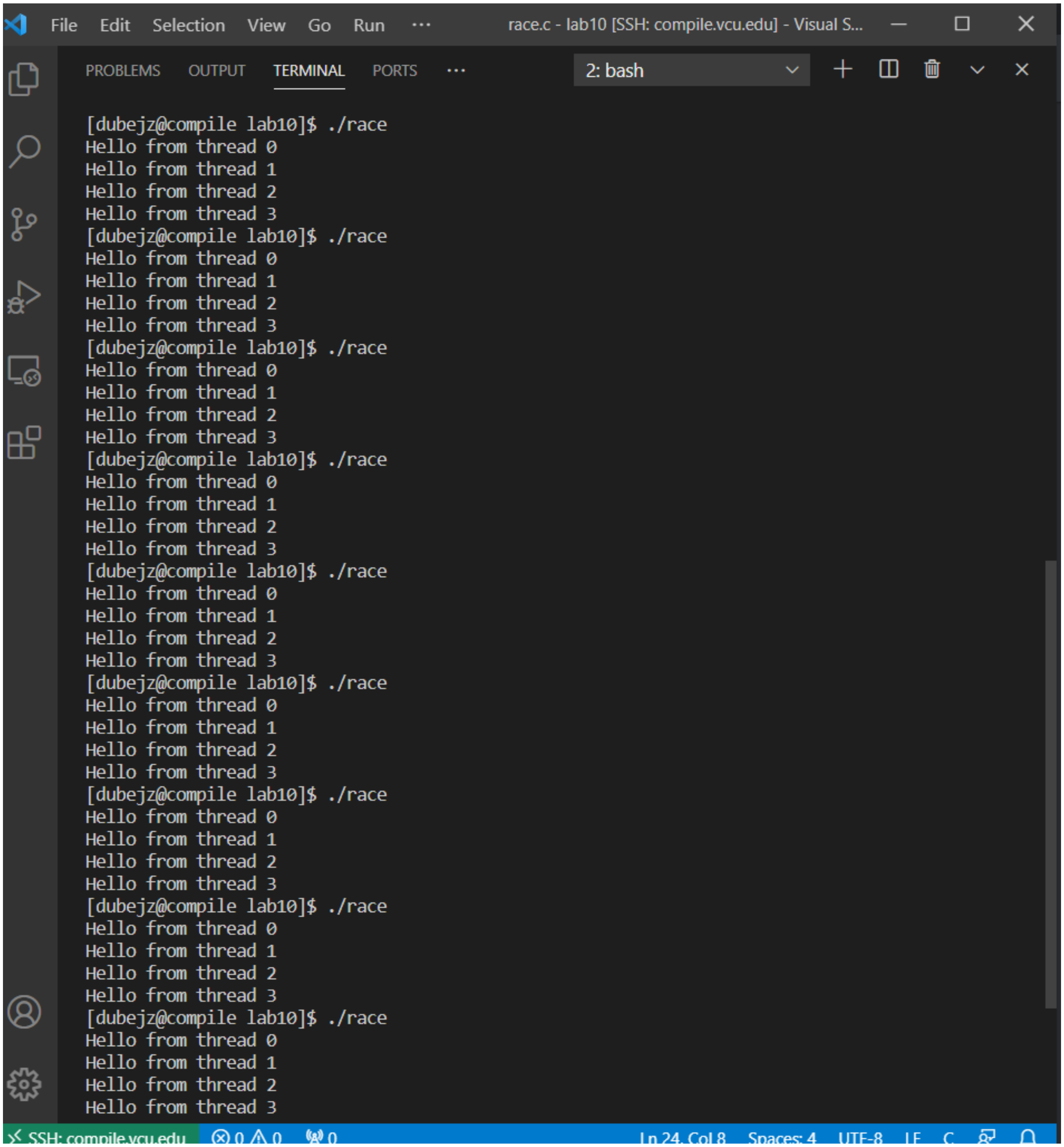
```
[dubejz@compile lab10]$ ls -a
.  ..  lab10
[dubejz@compile lab10]$ cd lab10
[dubejz@compile lab10]$ gcc -lpthread csapp.c hellobug.c -o hellobug
[dubejz@compile lab10]$ ./hellobug
[dubejz@compile lab10]$ gcc -lpthread csapp.c hellobug.c -o hellobug
[dubejz@compile lab10]$ ./hellobug
Segmentation fault
[dubejz@compile lab10]$ gcc -lpthread csapp.c hellobug.c -o hellobug
[dubejz@compile lab10]$ ./hellobug
Hello, world!
[dubejz@compile lab10]$ 
```

1b)

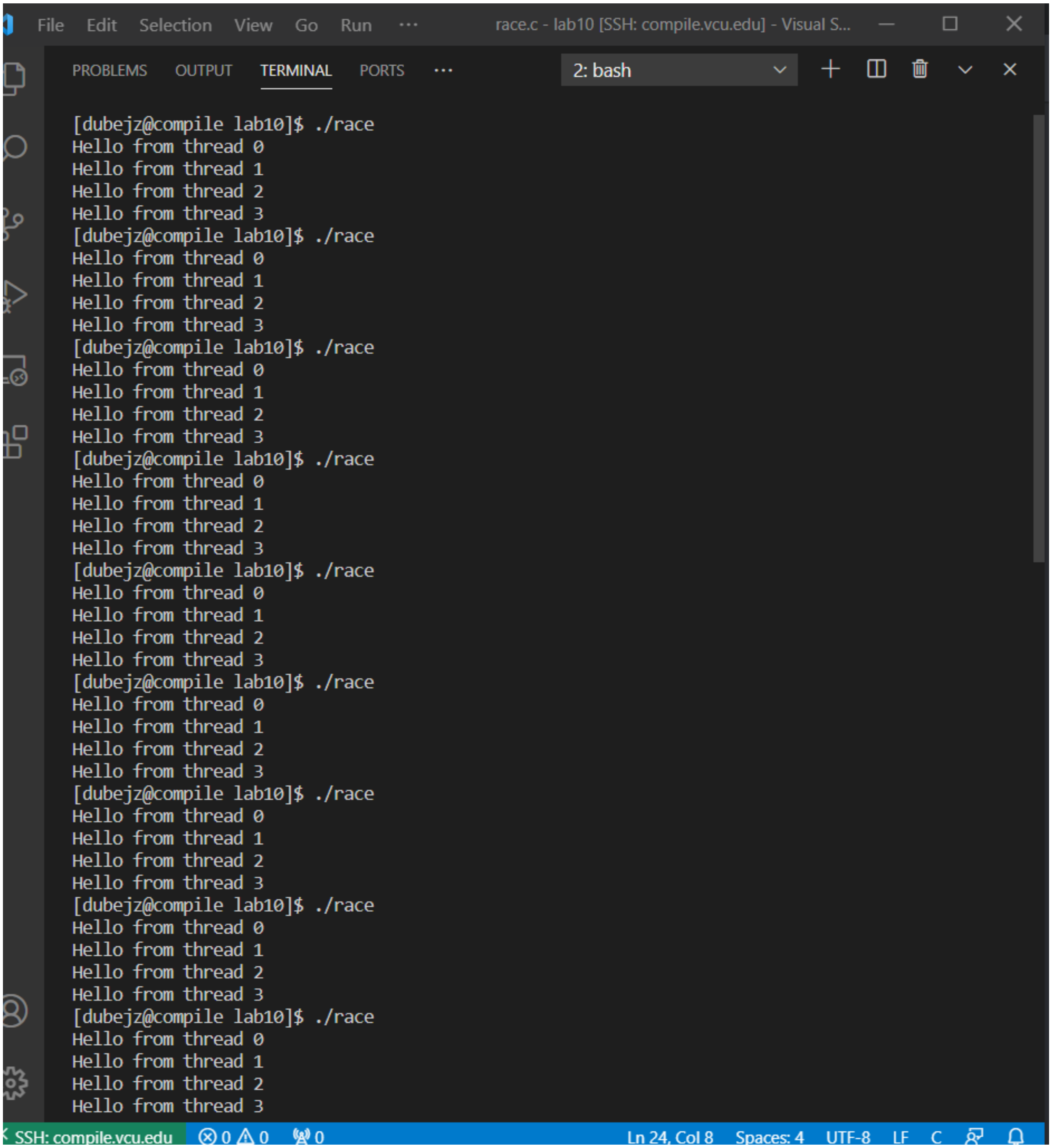


```c
/*
 * hellobug.c - "hello, world" program with a bug
 */
/* $begin hellobug */
/* WARNING: This code is buggy! */
#include "csapp.h"
void *thread(void *vargp);

int main()
{
    pthread_t tid;
    Pthread_create(&tid, NULL, thread, NULL);
    pthread_join(tid, NULL);
    exit(0); //line:conc:hellobug:exit
}

/* Thread routine */
void *thread(void *vargp)
{
    Sleep(1);
    printf("Hello, world!\n");
    return NULL;
}
/* $end hellobug */
```

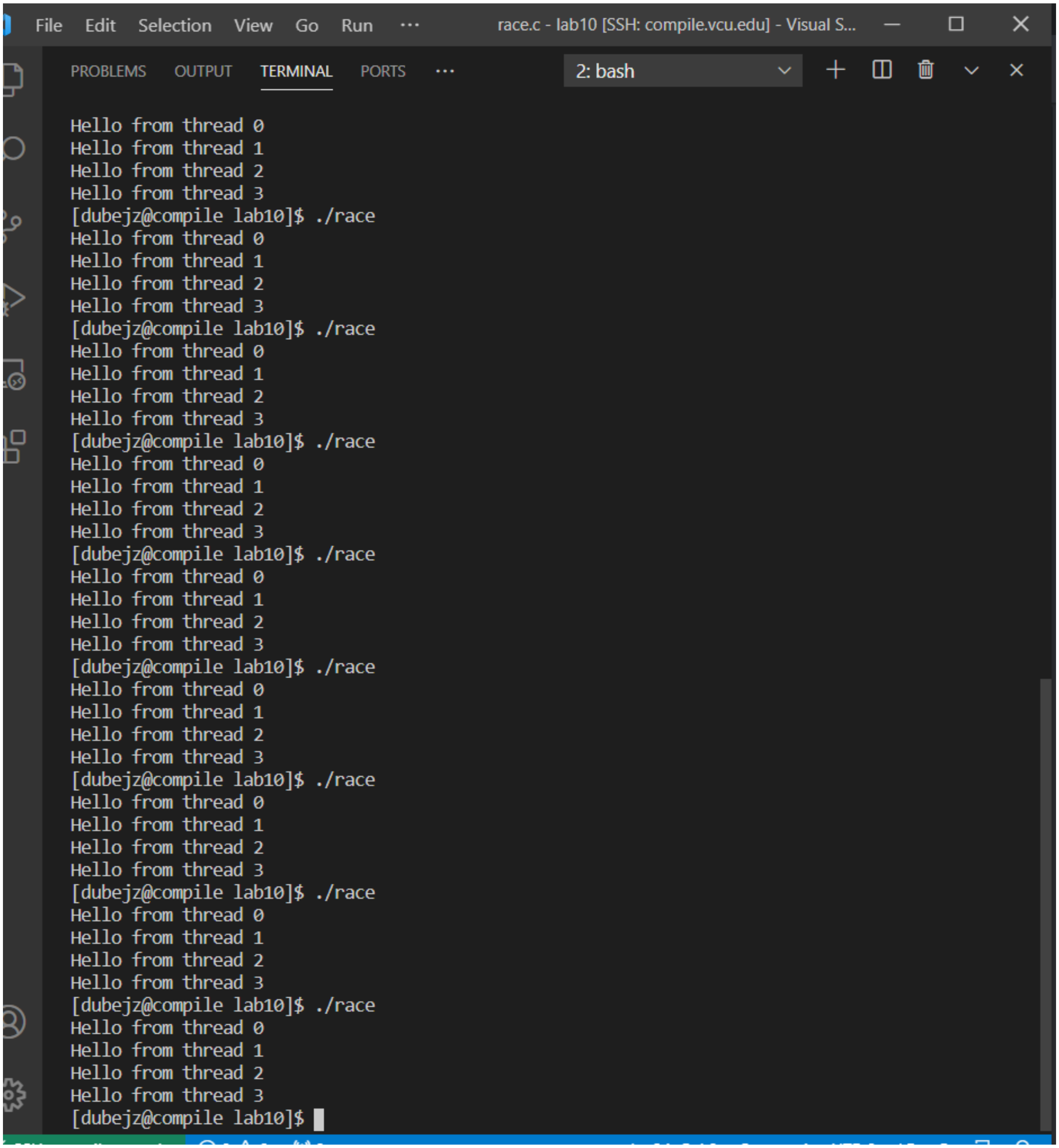
2a) (20 iterations of ./race)



```
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
```

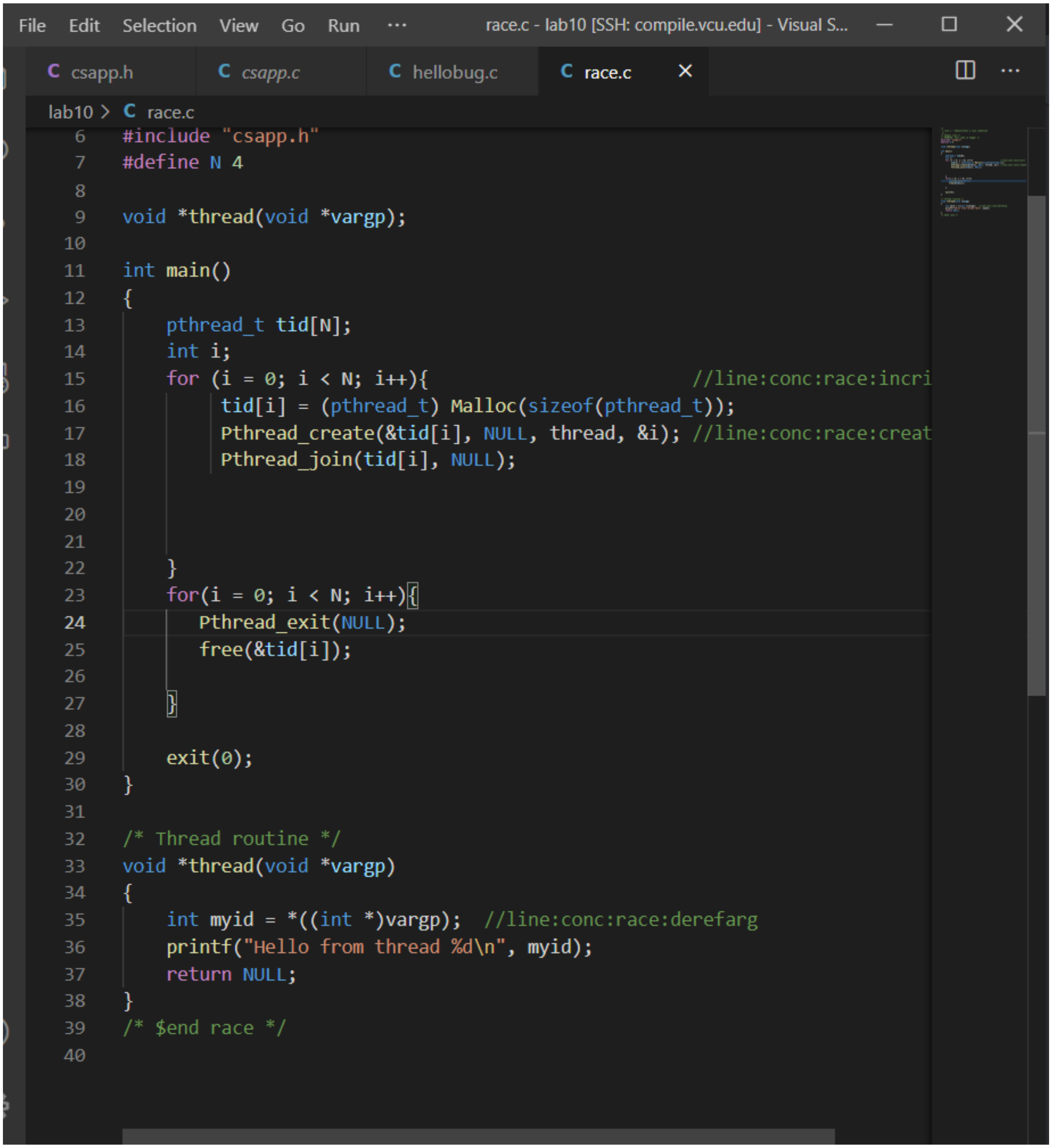


```
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
```



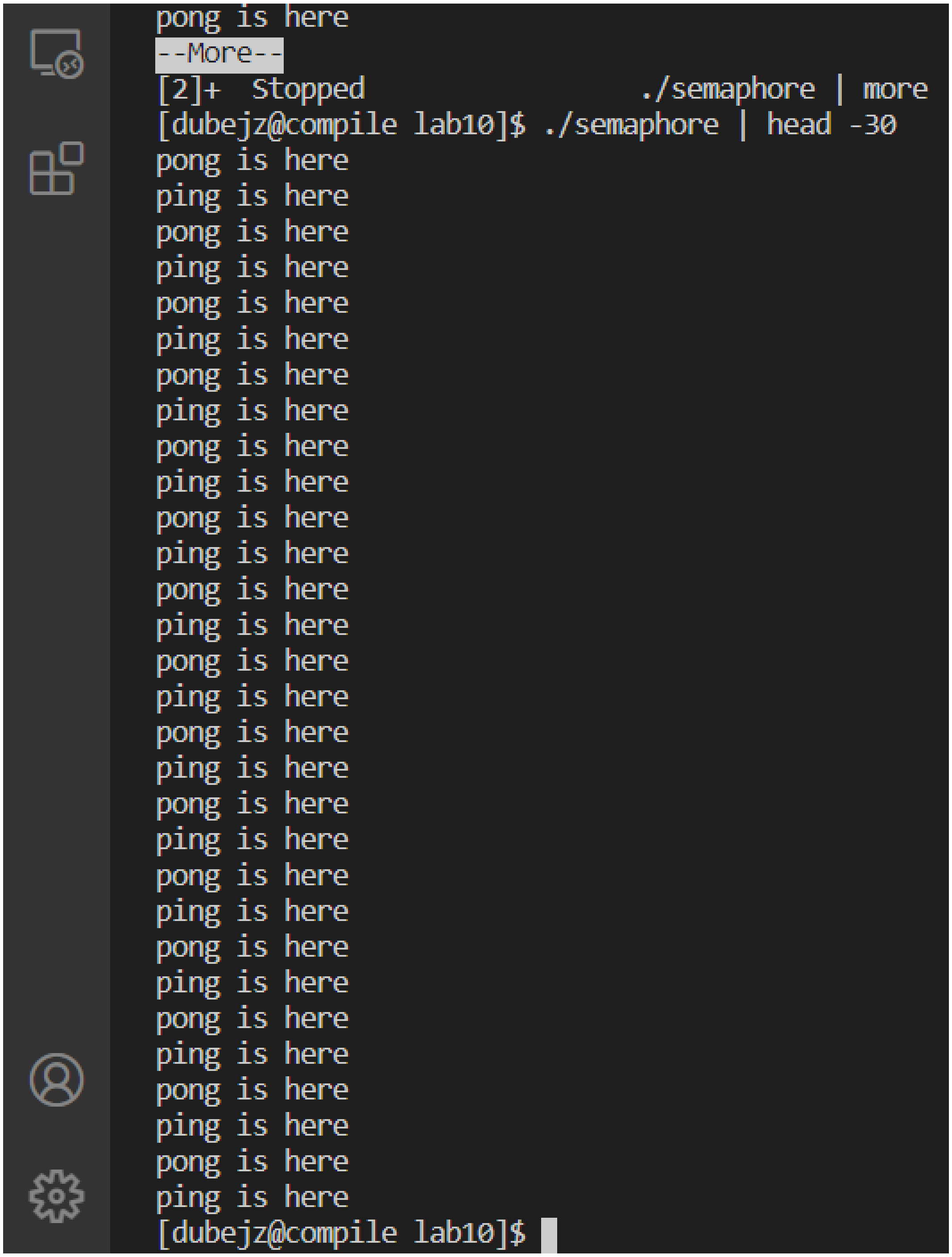
```
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ ./race
Hello from thread 0
Hello from thread 1
Hello from thread 2
Hello from thread 3
[dubejz@compile lab10]$ 
```

2b)



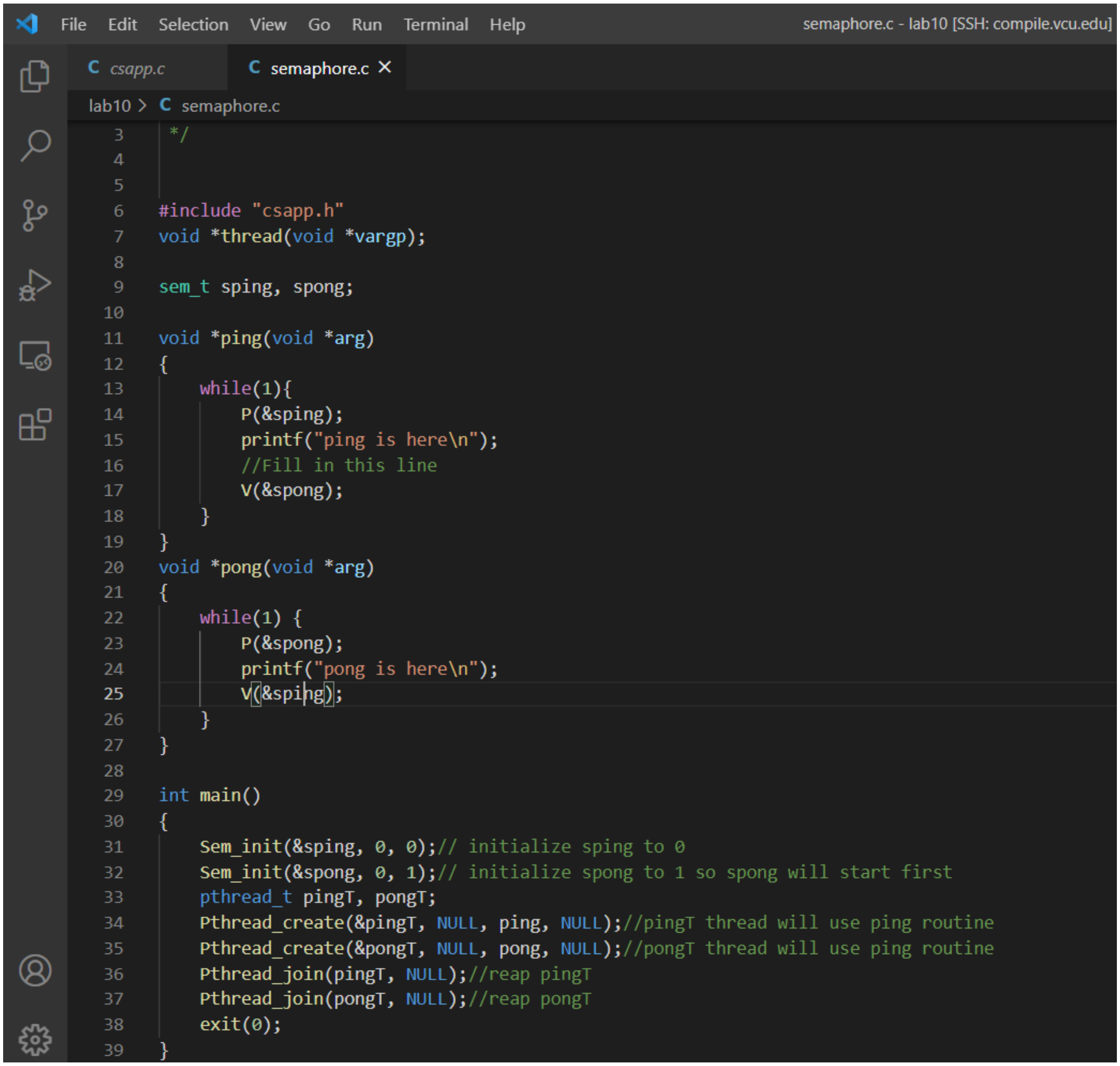
```c
#include "csapp.h"
#define N 4

void *thread(void *vargp);

int main()
{
    pthread_t tid[N];
    int i;
    for (i = 0; i < N; i++){                         //line:conc:race:incri
         tid[i] = (pthread_t) Malloc(sizeof(pthread_t));
         Pthread_create(&tid[i], NULL, thread, &i); //line:conc:race:creat
         Pthread_join(tid[i], NULL);



    }
    for(i = 0; i < N; i++){
       Pthread_exit(NULL);
       free(&tid[i]);

    }

    exit(0);
}

/* Thread routine */
void *thread(void *vargp)
{
    int myid = *((int *)vargp);  //line:conc:race:derefarg
    printf("Hello from thread %d\n", myid);
    return NULL;
}
/* $end race */

```

3a)



```
pong is here
--More--
[2]+  Stopped                 ./semaphore | more
[dubejz@compile lab10]$ ./semaphore | head -30
pong is here
ping is here
pong is here
ping is here
pong is here
ping is here
pong is here
ping is here
pong is here
ping is here
pong is here
ping is here
pong is here
ping is here
pong is here
ping is here
pong is here
ping is here
pong is here
ping is here
pong is here
ping is here
pong is here
ping is here
pong is here
ping is here
pong is here
ping is here
pong is here
ping is here
[dubejz@compile lab10]$
```

3b)



```c
 */


#include "csapp.h"
void *thread(void *vargp);

sem_t sping, spong;

void *ping(void *arg)
{
    while(1){
        P(&sping);
        printf("ping is here\n");
        //Fill in this line
        V(&spong);
    }
}
void *pong(void *arg)
{
    while(1) {
        P(&spong);
        printf("pong is here\n");
        V(&sping);
    }
}

int main()
{
    Sem_init(&sping, 0, 0);// initialize sping to 0
    Sem_init(&spong, 0, 1);// initialize spong to 1 so spong will start first
    pthread_t pingT, pongT;
    Pthread_create(&pingT, NULL, ping, NULL);//pingT thread will use ping routine
    Pthread_create(&pongT, NULL, pong, NULL);//pongT thread will use ping routine
    Pthread_join(pingT, NULL);//reap pingT
    Pthread_join(pongT, NULL);//reap pongT
    exit(0);
}
```