Homework 11

Problem 1

Using the progress graph in Figure 12-21 of file "**badcnt.c**", draw the following trajectories out and point out the value of **cnt** after the execution (assume the value of **cnt** is **0** initially for each trajectory).

A) H1, L1, H2, L2, U2, U1, S2, T2, S1, T1
B) H2, L2, U2, H1, S2, L1, T2, U1, S1, T1
C) H1, L1, U1, H2, L2, S1, U2, S2, T1, T2
D) H1, H2, L1, U1, S1, L2, U2, T1, S2, T2

Problem 2

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

void *thread(void *vargp) {
    int myid = *((int *)vargp);
    printf("in thread %d\n", myid);
    return NULL;
}

int main() {
    pthread_t tid[N];
    int i, *ptr;

    for (int i = 0; i < N; i++) {
        ptr = Malloc(sizeof(int));
        *ptr = i;
        // Create a thread running thread with argument ptr
        // Your code here

        Free(ptr);
    }
    // Join all threads
```

1. Complete the previous code according to the comment.
2. Is there any race condition in the previous code? Why or why not?

Problem 3

Consider the following three functions, please point out whether they are thread safe or not and whether they are reentrant or not.

```
int t;
void swap1(int *x, int *y)
{ t = *x; *x = *y; *y = t; }

void swap2(int *x, int *y)
{ P(&mutex); t = *x; *x = *y; *y = t; V(&mutex); }

void swap3(int *x, int *y)
{ int t; t = *x; *x = *y; *y = t; }
```

Problem 4

Can the following program have dead lock? Try to explain your answer using progress graph.

T1	T2
P(a)	P(c)
P(b)	P(b)
V(a)	V(b)
P(c)	P(a)
V(b)	V(c)
V(c)	V(a)