QUESTION:

Plan a Guessing Game on using pthreads; you compete against pthreads to guess a number between 1 and 1000; the pthreads are delayed between their guesses using a suitable delay of approximately 0.1 to 2 seconds. Moreover, while you can use an efficient binary search, implement sub-optimal search methods for the pthreads. Who wins?

PROGRAM

/\*  
 ============================================================================  
 Name        : game3025.c  
 Author      : stephy baby  
 Version     :  
 Copyright   : Your copyright notice  
 Description : Hello World in C, Ansi-style  
 ============================================================================  
 \*/  
  
#include <stdio.h>  
#include <stdlib.h>  
#include <pthread.h>  
#include <unistd.h>  
#include <time.h>  
  
int a = 0;  
int b = 0;  
int x;  
void \*randomrand(void \*number)  
  
{  
    srand(time(0));  
    while((rand() % 1000) != x)  
        a++;  
}  
  
void \*binfun(void \*number)  
{  
    int num1 = 1;  
    int num2 = 1000;  
    int num;  
    num = ((num1 + num2)/ 2);  
  
  
    while(x != num)  
    {  
        if((x > num) && (x < num2))  
        {  
            num1 = num;  
        }  
        else if((x > num1) && (x < num))  
        {  
            num2 = num;  
        }  
        num = ((num1 + num2)/ 2);  
        b++;  
        usleep(1000);  
  
    }  
}  
  
int main()  
{  
    printf("Enter one number to guess \n");  
    scanf("%d", &x);  
  
    pthread\_t player1, player2;  
    if(pthread\_create(&player1, NULL, &randomrand, NULL)!=0)  
    {  
        printf("Failed to create the thread\n");  
        return 1;  
    }  
    if(pthread\_create(&player2, NULL, &binfun, NULL)!=0)  
    {  
        printf("Failed to create the thread\n");  
        return 1;  
    }  
    pthread\_join(player1, NULL);  
    pthread\_join(player2, NULL);  
    while (1)  
    {  
        printf("plr1 = %d",a);  
        printf("\n");  
        printf("plr2= %d",b);  
        printf("\n");  
        if (a > b)  
        {  
            printf("Game over : player 2 wins");  
            break;  
        }  
        if (b > a)  
        {  
            printf("Game over : player 1 wins");  
            break;  
        }  
        usleep(10000);  
  
    }  
    return 0;  
}

**OUTPUT**

