GEBZE TECHNICAL UNIVERSITY

CSE344
SYSTEM PROGRAMMING COURSE

HOMEWORK 5
REPORT

BARAN SOLMAZ
1801042601

Problem Defination:

- Using POSIX threads to parallelize a mathematical tasks.
- First task : matrix production C = AxB,
- Second task : 2D Discrete Fourier Transform of C,
- Every thread calculates specific columns of C matrix,
- Threads must wait other threads before starting second task,
- Use only with mutexes and condition variables.

Problem Solution Approach:

Mutex and Condition Variable:

Creating:

```
pthread_cond_t cond;
pthread_mutex_t mutex;
```

Defined as global variable.

Initializing:

```
void initialize_mutex(){
    pthread_mutex_init(&mutex, NULL);
    pthread_cond_init(&cond,NULL);
}
```

Removing:

```
void remove_mutex(){
    pthread_mutex_destroy(&mutex);
    pthread_cond_destroy(&cond);
}
```

Barrier:

```
arrived++;
while (arrived<threadNumber){
    pthread_cond_wait(&cond,&mutex);
    if (sig_check_thread() == 1 )
        return NULL;
}
pthread_cond_broadcast(&cond);
```

Lock/Unlock Mutex:

```
pthread_mutex_lock(&mutex);
```

```
pthread_mutex_unlock(&mutex);
```

Threads:

```c
void create_threads(int *arr,pthread_t *threads){
    for (int i = 0; i < threadNumber; i++){
        arr[i]=i;
        void *p=&arr[i];
        if (pthread_create(&threads[i], NULL, threadX, p) != 0)
            perror_call("pthread_create");
    }
}
```

```c
void join_threads(pthread_t *threads){
    for (int i = 0; i < threadNumber; i++)
        if (pthread_join(threads[i], NULL) != 0)
            perror_call("pthread_join");
}
```

ThreadX:

```c
void* threadX(void *in){
    int id = *((int *)in);
    struct timeval mid,end;
    int e=matrix_product(result,id * matrixSize / threadNumber,(id+1)*matrixSize / threadNumber);

    if (sig_check_thread() == 1 || e==1)
        return NULL;
    gettimeofday(&mid, NULL);
    printf("%s Thread %d has reached the rendezvous point in %.4f seconds.\n",
        get_timestamp(), id + 1, getTime(mid,start));

    pthread_mutex_lock(&mutex);

    arrived++;
    while (arrived<threadNumber){
        pthread_cond_wait(&cond,&mutex);
        if (sig_check_thread() == 1 )
            return NULL;
    }
    pthread_cond_broadcast(&cond);

    if (sig_check_thread() == 1)
        return NULL;
    printf("%s Thread %d is advancing to the second part\n",get_timestamp(), id+1);

    pthread_mutex_unlock(&mutex);

    if (sig_check_thread() == 1)
        return NULL;
    e=fourierTransform(id * matrixSize / threadNumber, (id + 1) * matrixSize / threadNumber);
    if (sig_check_thread() == 1 || e == 1)
        return NULL;
    gettimeofday(&end, NULL);
    printf("%s Thread %d has has finished the second part in %.4f seconds.\n",
        get_timestamp(), id + 1, getTime(end,mid));
    return NULL;
}
```

Calculating matrix product

Waiting for other threads -- Barrier

Last thread broadcasts condition variable

2D DFT calculations

Matrix Product:

```
int matrix_product(float** result,int start ,int end){
    for (int i = 0; i < matrixSize; i++)
        for (int j = start; j < end; j++)
            for (int k = 0; k < matrixSize; k++){
                result[i][j] += first[i][k] * second[k][j];
                if (sig_check_thread() == 1)
                    return 1;
            }
    return 0;
}
```

start : thread index * column size/ thread size
end: (thread index +1)* column size/ thread size

DFT:

```
int fourierTransform(int start,int end){
    for (int yWave = 0; yWave < matrixSize; yWave++) // Two inner loops iterate on output data.
        for (int xWave = start; xWave < end; xWave++)

            for (int ySpace = 0; ySpace < matrixSize; ySpace++){ // Two inner loops iterate on input data.
                for (int xSpace = 0; xSpace < matrixSize; xSpace++){
                    realOut[yWave][xWave] += (result[ySpace][xSpace] *
                        cos(-2 * M_PI * ((1.0 * xWave * xSpace / matrixSize)
                                + (1.0 * yWave * ySpace / matrixSize)));
                    imgOut[yWave][xWave] += (result[ySpace][xSpace] *
                        sin(-2 * M_PI * ((1.0 * xWave * xSpace / matrixSize)
                                + (1.0 * yWave * ySpace / matrixSize))));

                    if (sig_check_thread() == 1)
                        return 1;
                }
            }
    return 0;
}
```

start : thread index * column size/ thread size
end: (thread index +1)* column size/ thread size

Test 1:

```
./hw5 -i InputExample/data1 -j InputExample/data2 -o output4x2.csv -n 4 -m 2
```

```
./hw5 -i InputExample/data1 -j InputExample/data2 -o output4x8.csv -n 4 -m 8
```

Outputs:

```
./hw5 -i InputExample/data1 -j InputExample/data2 -o output4x2.csv -n 4 -m 2
Thu May 19 00:17:28 2022  Two matrices of size 16x16 have been read. The number of threads is 2.
Thu May 19 00:17:28 2022  Thread 1 has reached the rendezvous point in 0.0002 seconds.
Thu May 19 00:17:28 2022  Thread 2 has reached the rendezvous point in 0.0002 seconds.
Thu May 19 00:17:28 2022  Thread 2 is advancing to the second part
Thu May 19 00:17:28 2022  Thread 1 is advancing to the second part
Thu May 19 00:17:28 2022  Thread 1 has has finished the second part in 0.0030 seconds.
Thu May 19 00:17:28 2022  Thread 2 has has finished the second part in 0.0065 seconds.
Thu May 19 00:17:28 2022  The process has written the output file. The total time spent is 0.0098 seconds.
```

```
./hw5 -i InputExample/data1 -j InputExample/data2 -o output4x8.csv -n 4 -m 8
Thu May 19 00:17:28 2022  Two matrices of size 16x16 have been read. The number of threads is 8.
Thu May 19 00:17:28 2022  Thread 1 has reached the rendezvous point in 0.0001 seconds.
Thu May 19 00:17:28 2022  Thread 2 has reached the rendezvous point in 0.0002 seconds.
Thu May 19 00:17:28 2022  Thread 3 has reached the rendezvous point in 0.0002 seconds.
Thu May 19 00:17:28 2022  Thread 4 has reached the rendezvous point in 0.0002 seconds.
Thu May 19 00:17:28 2022  Thread 5 has reached the rendezvous point in 0.0003 seconds.
Thu May 19 00:17:28 2022  Thread 6 has reached the rendezvous point in 0.0003 seconds.
Thu May 19 00:17:28 2022  Thread 7 has reached the rendezvous point in 0.0003 seconds.
Thu May 19 00:17:28 2022  Thread 8 has reached the rendezvous point in 0.0004 seconds.
Thu May 19 00:17:28 2022  Thread 8 is advancing to the second part
Thu May 19 00:17:28 2022  Thread 2 is advancing to the second part
Thu May 19 00:17:28 2022  Thread 1 is advancing to the second part
Thu May 19 00:17:28 2022  Thread 8 has has finished the second part in 0.0009 seconds.
Thu May 19 00:17:28 2022  Thread 2 has has finished the second part in 0.0012 seconds.
Thu May 19 00:17:28 2022  Thread 1 has has finished the second part in 0.0013 seconds.
Thu May 19 00:17:28 2022  Thread 4 is advancing to the second part
Thu May 19 00:17:28 2022  Thread 7 is advancing to the second part
Thu May 19 00:17:28 2022  Thread 4 has has finished the second part in 0.0020 seconds.
Thu May 19 00:17:28 2022  Thread 7 has has finished the second part in 0.0027 seconds.
Thu May 19 00:17:28 2022  Thread 6 is advancing to the second part
Thu May 19 00:17:28 2022  Thread 3 is advancing to the second part
Thu May 19 00:17:28 2022  Thread 6 has has finished the second part in 0.0045 seconds.
Thu May 19 00:17:28 2022  Thread 3 has has finished the second part in 0.0046 seconds.
Thu May 19 00:17:28 2022  Thread 5 is advancing to the second part
Thu May 19 00:17:28 2022  Thread 5 has has finished the second part in 0.0053 seconds.
Thu May 19 00:17:28 2022  The process has written the output file. The total time spent is 0.0088 seconds.
```

Test 2:

Inputs:

```
./hw5 -i InputExample/data1 -j InputExample/data2 -o output8x2.csv -n 8 -m 2
```

```
./hw5 -i InputExample/data1 -j InputExample/data2 -o output8x8.csv -n 8 -m 8
```

Outputs:

```
./hw5 -i InputExample/data1 -j InputExample/data2 -o output8x2.csv -n 8 -m 2
Thu May 19 00:22:02 2022  Two matrices of size 256x256 have been read. The number of threads is 2.
Thu May 19 00:22:02 2022  Thread 2 has reached the rendezvous point in 0.1803 seconds.
Thu May 19 00:22:02 2022  Thread 1 has reached the rendezvous point in 0.1855 seconds.
Thu May 19 00:22:02 2022  Thread 1 is advancing to the second part
Thu May 19 00:22:02 2022  Thread 2 is advancing to the second part
Thu May 19 00:24:40 2022  Thread 2 has has finished the second part in 158.1696 seconds.
Thu May 19 00:24:40 2022  Thread 1 has has finished the second part in 158.2417 seconds.
Thu May 19 00:24:41 2022  The process has written the output file. The total time spent is 159.0007 seconds.
```

```
./hw5 -i InputExample/data1 -j InputExample/data2 -o output8x8.csv -n 8 -m 8
Thu May 19 00:26:47 2022  Two matrices of size 256x256 have been read. The number of threads is 8.
Thu May 19 00:26:47 2022  Thread 8 has reached the rendezvous point in 0.0541 seconds.
Thu May 19 00:26:47 2022  Thread 3 has reached the rendezvous point in 0.0641 seconds.
Thu May 19 00:26:47 2022  Thread 7 has reached the rendezvous point in 0.0694 seconds.
Thu May 19 00:26:47 2022  Thread 2 has reached the rendezvous point in 0.0775 seconds.
Thu May 19 00:26:47 2022  Thread 1 has reached the rendezvous point in 0.0802 seconds.
Thu May 19 00:26:47 2022  Thread 5 has reached the rendezvous point in 0.0869 seconds.
Thu May 19 00:26:47 2022  Thread 6 has reached the rendezvous point in 0.0904 seconds.
Thu May 19 00:26:47 2022  Thread 4 has reached the rendezvous point in 0.0923 seconds.
Thu May 19 00:26:47 2022  Thread 4 is advancing to the second part
Thu May 19 00:26:47 2022  Thread 8 is advancing to the second part
Thu May 19 00:26:47 2022  Thread 2 is advancing to the second part
Thu May 19 00:26:47 2022  Thread 3 is advancing to the second part
Thu May 19 00:26:47 2022  Thread 7 is advancing to the second part
Thu May 19 00:26:47 2022  Thread 1 is advancing to the second part
Thu May 19 00:26:47 2022  Thread 6 is advancing to the second part
Thu May 19 00:26:47 2022  Thread 5 is advancing to the second part
Thu May 19 00:28:47 2022  Thread 5 has has finished the second part in 119.6484 seconds.
Thu May 19 00:28:47 2022  Thread 1 has has finished the second part in 120.0548 seconds.
Thu May 19 00:28:48 2022  Thread 6 has has finished the second part in 121.0724 seconds.
Thu May 19 00:28:48 2022  Thread 4 has has finished the second part in 121.3671 seconds.
Thu May 19 00:28:49 2022  Thread 3 has has finished the second part in 121.7741 seconds.
Thu May 19 00:28:49 2022  Thread 2 has has finished the second part in 121.8519 seconds.
Thu May 19 00:28:49 2022  Thread 8 has has finished the second part in 122.3688 seconds.
Thu May 19 00:28:50 2022  Thread 7 has has finished the second part in 122.8207 seconds.
Thu May 19 00:28:50 2022  The process has written the output file. The total time spent is 123.4659 seconds.
```