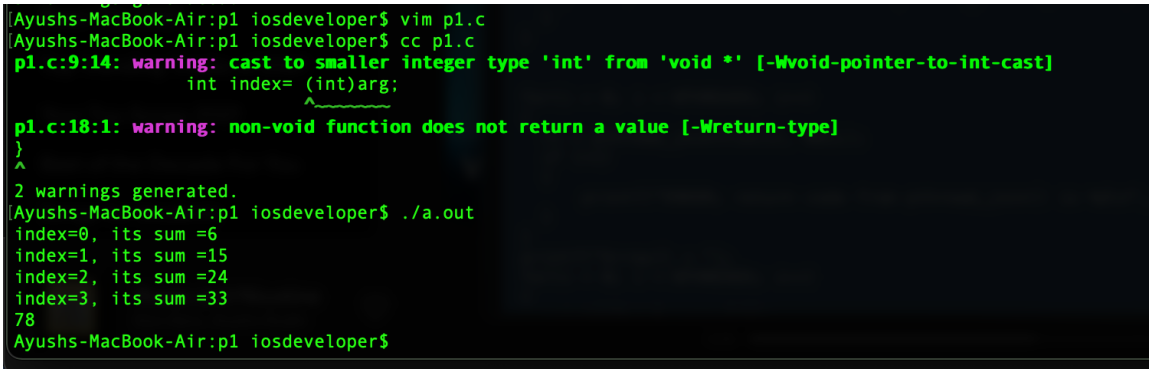
OS Lab 7

1. **Write Multithreaded program using 4 threads, for adding 12 array elements each thread is adding 3 elements of the array.**

```
#include<stdio.h>
#include<pthread.h>

  int sum[4];
  int i[]={1,2,3,4,5,6,7,8,9,10,11,12};

  void *function (void *arg){

    int index= (int)arg;
    int start = index * 3;
    int cnt=0;

  while (cnt<3){
    sum[index] += *(i+cnt+start);
    cnt++;
  }
  printf("index=%d, its sum =%d \n",index,sum[index]);
}

void main(){

  pthread_t t1,t2,t3,t4;

  pthread_create (&t1, NULL, function, (void *)0);
  pthread_create (&t2, NULL, function, (void *)1);
  pthread_create (&t3, NULL, function, (void *)2);
  pthread_create (&t4, NULL, function, (void *)3);
  pthread_join(t1,NULL);
  pthread_join(t2,NULL);
  pthread_join(t3,NULL);
  pthread_join(t4,NULL);

  printf("%d\n", sum[0] + sum[1] + sum[2] + sum[3]);
}
```

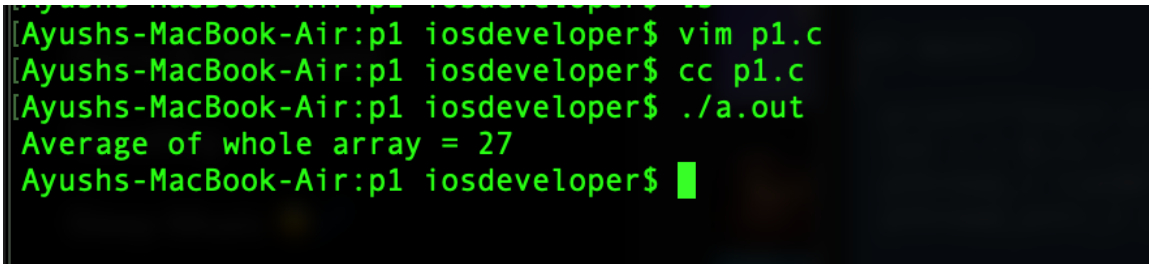


2. **Multithreaded program for calculating average of array elements by one thread and calculating factorial of each array element by creating one thread for each element.**

    1. **Calculating average of array elements**

```
#include <pthread.h>
#include <stdlib.h>
#include <stdio.h>

typedef struct data{
    int* arr;
    int thread_num;
} data;

int arrSize = 10;

void* halfSum(void* p){
    data* ptr = (data*)p;
    int n = ptr->thread_num;


    int* thread_sum = (int*) calloc(1, sizeof(int));

    if(n == 0){
        for(int i = 0; i < arrSize/2; i++)
            thread_sum[0] = thread_sum[0] + ptr->arr[i];
    }
    else{
        for(int i = arrSize/2; i < arrSize; i++)
            thread_sum[0] = thread_sum[0] + ptr->arr[i];
    }

    pthread_exit(thread_sum);
}

int main(void){

    int* int_arr = (int*) calloc(arrSize, sizeof(int));
    for(int i = 0; i < arrSize; i++)
```

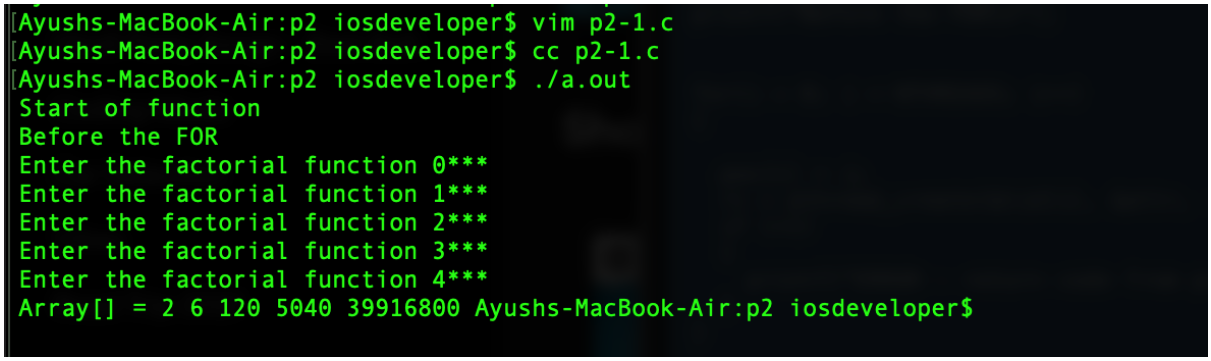
```
        int_arr[i] = i + 1;


    data thread_data[2];
    thread_data[0].thread_num = 0;
    thread_data[0].arr = int_arr;
    thread_data[1].thread_num = 1;
    thread_data[1].arr = int_arr;


    pthread_t tid[2];


    pthread_create(&tid[0], NULL, halfSum, &thread_data[0]);
    pthread_create(&tid[1], NULL, halfSum, &thread_data[1]);


    int* sum0;
    int* sum1;
    //int average=(*sum0 + *sum1)/2;


    pthread_join(tid[0], (void**)&sum0);
    pthread_join(tid[1], (void**)&sum1);

    printf("Average of whole array = %i\n", (*sum0+*sum1)/2);

    return 0;
}
```



2. **Calculating factorial of each array element by creating one thread for each element**

```
#include <pthread.h>
#include <stdio.h>
#include <unistd.h>

#define NTHREADS 5


int array[5] = { 2, 3, 5, 7, 11 };
```

```
void *Fatorial(void *pos)
{
    int res = 1, fat;
    int *postion = (int *)pos;
    printf("Enter the factorial function %d***\n", *postion);

    for(fat = array[*postion]; fat > 1; fat--)
    {
         res *= fat;
    }

    array[*postion] = res;
    pthread_exit(NULL);
}

int main()
{
  printf("Start of function\n");
  int i = 0,rc = 0, pos[NTHREADS];
  pthread_t tid[NTHREADS];
  pthread_attr_t attr;

  pthread_attr_init(&attr);
  pthread_attr_setdetachstate(&attr, PTHREAD_CREATE_JOINABLE);
  printf("Before the FOR\n");


  for(i = 0; i < NTHREADS; i++)
  {

    pos[i] = i;
    rc = pthread_create(&tid[i], &attr, &Fatorial, (void *)&pos[i]);
    if (rc)
    {
      printf("ERROR - return code from pthread_create() is %d\n", rc);
    }
  }


  for(i = 0; i < NTHREADS; i++)
  {
    rc = pthread_join(tid[i], NULL);
    if (rc)
    {
        printf("ERROR; return code from pthread_join() is %d\n", rc);
    }
  }
  printf("Array[] = ");
  for(i = 0; i < NTHREADS; i++)
  {
    printf("%d ",array[i]);
  }
  pthread_attr_destroy(&attr);
  pthread_exit(NULL);
  return 0;
}
```


Ayushs-MacBook-Air:p2 iosdeveloper$ vim p2-1.c
Ayushs-MacBook-Air:p2 iosdeveloper$ cc p2-1.c
Ayushs-MacBook-Air:p2 iosdeveloper$ ./a.out
Start of function
Before the FOR
Enter the factorial function 0***
Enter the factorial function 1***
Enter the factorial function 2***
Enter the factorial function 3***
Enter the factorial function 4***
Array[] = 2 6 120 5040 39916800 Ayushs-MacBook-Air:p2 iosdeveloper$