I globally initialized some variables as those are used globally.

Eg : tot\_sum2 sum2[]

initialize the seperate 2 pthreads for main two part of our program as,

pthread\_t tid[60];

pthread\_attr\_t attr[60];

pthread\_t tid3[60];

pthread\_attr\_t attr3[60];

Other than children and grandchildren threads there must be 2 separate threads to run, run\_part2 and run\_part3 methods.Those are initialized as functions.

pthread\_t maint[2];

pthread\_attr\_t mainattr[2];

Inside the main method those threads are initialized and created as follows.

pthread\_attr\_init(&mainattr[0]);

pthread\_create(&maint[0],&mainattr[0],run\_part2,NULL);

pthread\_join(maint[0],NULL);

pthread\_attr\_init(&mainattr[1]);

pthread\_create(&maint[1],&mainattr[1],run\_part3,NULL);

pthread\_join(maint[1],NULL);

For part 2:

ran a for loop which is size of argv[2] and create threads for each loop.this is a child thread.

Using runner method I sum up the values between 0-argv[1]

For write those sums and strings into a file I use append\_string method which is build by myself.

For part 3 :

Other than two main threads there are argv[2]\*argv[3]+argv[2] total threads.So I create a for loop to run all those threads.

create separate child\_runner and grand\_runner method to run child and grand threads respectively.

In the child\_runner method I again initialize the grand threads and pass the values to the grand\_runner method and according to the argv[] summation was happen.

append\_string method has used to append the relevant strings to the file.

Here child threads and grandchildren threads are indexed in following manner

input = ./main 100 10 2

total threads = 30

child indexs = 0(1,2), 3(4,5), 6(7,8), 9(10,11), 12(13,14), 15(16,17), 18(19,20), 21(22,23), 24(25,26), 27(28,29)

grand indexs are placed inside the brackets