# Project I :

1 ) Created independent functions to :

column\_check  
bool column\_check(int array[][9]); //This function checks for duplicate values in each column.

row\_check()  
bool row\_check(int array[][9]); // This function checks for duplicate values in each row.

subgrid\_check()  
bool subgrid\_check(int array[][9]); //This function checks for duplicate values in each 3x3 subgrid.  
  
2) Created a Independent thread for each function that checks their respective scope for the validation.

A screen shot of a computer

Description automatically generated3) Output to prove its multithreading:

Figure showing Thread 1,Thread 2 and Thread 3 are running Parallelly .

# Project II:

1. Wrote function for sorting.

* Given a array of elements this functions basically sorts the element in that array.

void sorting\_elements(int start,int end){

    for (int i = start ; i <= end-1; ++i ){

        int temp\_min = vect\_list[i];

        for (int j = i +1 ; j <= end;++j){

            if (temp\_min > vect\_list[j]){

                int temp = temp\_min;

                temp\_min = vect\_list[j];

                vect\_list[j]= temp;

            }

        }

        vect\_list[i] = temp\_min;

    }

}

1. Given two sorted arrays this function returns the overall merged array in a sorted form.
2. void merge\_elements(int start1,int end1){
3. int start2 = end1+1;
4. int end2 = vect\_list.size()-1;
5. int i = start1;
6. int j = start2;
7. while(i <= end1 && j <=end2){
8. if(vect\_list[i] < vect\_list[j]){
9. sorted\_list.push\_back(vect\_list[i]);
10. ++i;
11. }
12. else{
13. sorted\_list.push\_back(vect\_list[j]);
14. ++j;
15. }
16. }
17. while (i <= end1){
18. sorted\_list.push\_back(vect\_list[i]);
19. ++i;
20. }
22. while (j <= end2){
23. sorted\_list.push\_back(vect\_list[j]);
24. ++j;
25. }
27. }

3) We spiltted the given array into two halves and two different threads of sorting function was run concurrently to obtain the sorted array separately.

4) The Separately obtained sorted array was passed to merging function using third thread.

A screen shot of a computer

Description automatically generated