

Name	Manish Shashikant Jadhav
UID no.	2023301005

Experiment 4A

AIM :

Write a multithreaded program that calculates various statistical values for a list of numbers. This program will be passed a series of numbers on the command line and will then create three separate worker threads. One thread will determine the average of the numbers, the second will determine the maximum value, and the third will determine the median value.

Discussion & Output:

Program:

```
#include <stdio.h>
#include <pthread.h>
#define MAX_SIZE 50
int arr[MAX_SIZE], size, index;
void *calculateAvg()
{
    float sum = 0;
    float average;
    printf("What will be the size of an array? ");
    scanf("%d", &size);
    printf("Please enter %d elements:\n", size);
    for (index = 0; index < size; index++)
    {
        scanf("%d", &arr[index]);
    }
    for (index = 0; index < size; index++)
    {
        sum = sum + arr[index];
    }
    average = sum / size;
    printf("The average of given elements in an array is: %.2f\n", average);
    return NULL;
}
void swap(int *x, int *y)
{
```

```

int temp = *x;
*x = *y;
*y = temp;
}
void *calculateMid()
{
    for (index = 0; index < size - 1; index++)
    {
        int min = index;
        for (int j = index + 1; j < size; j++)
        {
            if (arr[j] < arr[min])
                min = j;
        }
        swap(&arr[min], &arr[index]);
    }
    float median;
    if (size % 2 == 0)
    {
        median = (arr[(size - 1) / 2] + arr[(size - 1) / 2 + 1]) / 2.0;
    }
    else
    {
        median = arr[size / 2];
    }
    printf("The median of given elements in an array is: %.2f\n", median);
    return NULL;
}
void *calculateMaxi()
{
    int maxVal = arr[0];
    for (int i = 1; i < size; i++)
    {
        if (maxVal < arr[i])
        {
            maxVal = arr[i];
        }
    }
}

```

```

    }
}

printf("The maximum of given elements in an array is:: %d\n", maxVal);
return NULL;
}

int main()
{
    pthread_t thread1, thread2, thread3;
    pthread_create(&thread1, NULL, calculateAvg, NULL);
    pthread_join(thread1, NULL);
    pthread_create(&thread2, NULL, calculateMid, NULL);
    pthread_join(thread2, NULL);
    pthread_create(&thread3, NULL, calculateMaxi, NULL);
    pthread_join(thread3, NULL);
    return 0;
}

```

Output:

```

Windows PowerShell
PS D:\Manish\SPIT\4th SEM\OS\Exp4> gcc 4a.c
4a.c:4:26: warning: built-in function 'index' declared as non-function [-Wbuiltin-declaration-mismatch]
    int arr[MAX_SIZE], size, index;
                           ^
PS D:\Manish\SPIT\4th SEM\OS\Exp4> ./a.exe
What will be the size of an array? 9
Please enter 9 elements:
28
9
13
2
19
10
4
6
24
The average of given elements in an array is: 12.78
The median of given elements in an array is: 10.00
The maximum of given elements in an array is:: 28
PS D:\Manish\SPIT\4th SEM\OS\Exp4> |

```

CONCLUSION: Hence, by completing this experiment I understood how to write a multithreaded program that calculates various statistical values for a list of numbers. Program will be passed a series of numbers on the command line and will then create three separate worker threads. One thread will determine the average of the numbers, second will determine the maximum value, and the third will determine the median value.

Experiment 4B

AIM : Write a multithreaded program that creates two functions one for adding the element from array and other function to delete last two elements in array and print it. The code runs 5 times, addition of elements happens first and then the deletion. Check necessary conditions like minimum 2 elements remain in array.

Discussion & Output:

Program:

```
#include <stdio.h>
#include <pthread.h>
#define MAX_SIZE 8
int array[MAX_SIZE];
int size = 0;
pthread_mutex_t mutex = PTHREAD_MUTEX_INITIALIZER;
void *addValue(void *arg)
{
    int num = *(int *)arg;
    pthread_mutex_lock(&mutex);
    if (size < MAX_SIZE)
    {
        array[size++] = num;
        printf("New Value %d is added to the array. Hence the New array is: ", num);
        for (int i = 0; i < size; i++)
        {
            printf("%d ", array[i]);
        }
        printf("\n");
    }
    else
    {
        printf("Cannot add %d. Array is full.\n", num);
    }
    pthread_mutex_unlock(&mutex);
    return NULL;
}
void *deleteValue(void *arg)
{
    pthread_mutex_lock(&mutex);
```

```

if (size >= 2)
{
    int lastElement1 = array[size - 1];
    int lastElement2 = array[size - 2];
    size -= 2;
    printf("Deleted %d and %d from the array. New array is: ", lastElement1,
        lastElement2);
    for (int i = 0; i < size; i++)
    {
        printf("%d ", array[i]);
    }
    printf("\n");
}
else
{
    printf("As there are only 2 elements left in array. Therefore cannot perform delete operation.\n");
}
pthread_mutex_unlock(&mutex);
return NULL;
}

int main()
{
    pthread_t addThread, deleteThread;
    int numbers[MAX_SIZE];
    printf("Enter up to %d integers: \n", MAX_SIZE);
    int number, index = 0;
    while (index < MAX_SIZE && scanf("%d", &number) == 1 && number != -1)
    {
        numbers[index++] = number;
    }
    for (int i = 0; i < index; i++)
    {
        pthread_create(&addThread, NULL, addValue, &numbers[i]);
        pthread_join(addThread, NULL);
        pthread_create(&deleteThread, NULL, deleteValue, NULL);
        pthread_join(deleteThread, NULL);
    }
}

```

```

}

pthread_mutex_destroy(&mutex);

return 0;

}

```

Output:

```

Windows PowerShell
PS D:\Manish\SPIT\4th SEM\OS\Exp4> gcc 4b.c
PS D:\Manish\SPIT\4th SEM\OS\Exp4> ./a.exe
Enter up to 8 integers:
28
9
13
2
19
10
24
8
New Value 28 is added to the array. Hence the New array is: 28
As there are only 2 elements left in array. Therefore cannot perform delete operation.
New Value 9 is added to the array. Hence the New array is: 28 9
Deleted 9 and 28 from the array. New array is:
New Value 13 is added to the array. Hence the New array is: 13
As there are only 2 elements left in array. Therefore cannot perform delete operation.
New Value 2 is added to the array. Hence the New array is: 13 2
Deleted 2 and 13 from the array. New array is:
New Value 19 is added to the array. Hence the New array is: 19
As there are only 2 elements left in array. Therefore cannot perform delete operation.
New Value 10 is added to the array. Hence the New array is: 19 10
Deleted 10 and 19 from the array. New array is:
New Value 24 is added to the array. Hence the New array is: 24
As there are only 2 elements left in array. Therefore cannot perform delete operation.
New Value 8 is added to the array. Hence the New array is: 24 8
Deleted 8 and 24 from the array. New array is:
PS D:\Manish\SPIT\4th SEM\OS\Exp4> |

```

CONCLUSION: Hence, by completing this experiment I came to know about how to write a multithreaded program that creates two functions one for adding the element from array and other function to delete last two elements in array and print it. The code runs 5 times, addition of elements happens first and then the deletion. Check necessary conditions like minimum 2 elements remain in array.