

List of Experiments (Programming Languages: C or C++)

4. Implement a data parallelism using multi-threading.

main.c

```
1  #include <stdio.h>
2  #include <pthread.h>
3
4  #define NUM_THREADS 4
5  #define ARRAY_SIZE 100
6  #define CHUNK_SIZE (ARRAY_SIZE / NUM_THREADS)
7
8  int data[ARRAY_SIZE];
9  int result[ARRAY_SIZE];
10
11 void* threadFunction(void* threadId) {
12     int tid = *((int*)threadId);
13
14     int start = tid * CHUNK_SIZE;
15     int end = start + CHUNK_SIZE;
16
17     for (int i = start; i < end; i++) {
18         // Process the data element
19         result[i] = data[i] * 2;
20     }
21
22     pthread_exit(NULL);
23 }
24
```

```
25 int main() {
26     pthread_t threads[NUM_THREADS];
27     int threadIds[NUM_THREADS];
28     // Initialize data array
29     for (int i = 0; i < ARRAY_SIZE; i++) {
30         data[i] = i;
31     }
32
33     // Create threads
34     for (int i = 0; i < NUM_THREADS; i++) {
35         threadIds[i] = i;
36         pthread_create(&threads[i], NULL, threadFunction, (void*)&threadIds[i]);
37     }
38
39     // Wait for threads to complete
40     for (int i = 0; i < NUM_THREADS; i++) {
41         pthread_join(threads[i], NULL);
42     }
43
44     // Display the results
45     for (int i = 0; i < ARRAY_SIZE; i++) {
46         printf("%d ", result[i]);
47     }
48     printf("\n");
49     return 0;
50 }
```

Output

[Clear](#)

/tmp/AJMd0MdATH.o

```
0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50 52 54 56 58 60 62 64
66 68 70 72 74 76 78 80 82 84 86 88 90 92 94 96 98 100 102 104 106 108 110 112 114 116 118
120 122 124 126 128 130 132 134 136 138 140 142 144 146 148 150 152 154 156 158 160 162
164 166 168 170 172 174 176 178 180 182 184 186 188 190 192 194 196 198
```

PAJAMA PADHAI