

CSE4001 – Lab Assessment 3

17BCI0113 – Namit Nathwani

Prof. Jaisakthi S M – L55+L56

Question: Write an OpenMP program to specify that the schedule(dynamic, chunk-size) clause of the loop construct specifies that the for loop has the dynamic scheduling type.

Code:

```
#include <omp.h>
#include <stdio.h>
#include <stdlib.h>
#define CHUNKSIZE 10
#define N 100
int main (int argc, char *argv[])
{
    int nthreads, tid, i, chunk;
    float a[N], b[N], c[N];

    for (i=0; i < N; i++)
        a[i] = b[i] = i * 1.0;

    chunk = CHUNKSIZE;

    #pragma omp parallel shared(a,b,c,nthreads,chunk)
    private(i,tid)
    {
        tid = omp_get_thread_num();

        if (tid == 0) {
            nthreads = omp_get_num_threads();
            printf("Number of threads = %d\n", nthreads);
       }
```

```

printf("Thread %d starting...\n",tid);

#pragma omp for schedule(dynamic,chunk)

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

    c[i] = a[i] + b[i];

    printf("Thread %d: c[%d]= %f\n",tid,i,c[i]);

}

}

}

```

Output:

```

17bci0113@sjt418scs045:~/Documents$ gcc Chunks.c -fopenmp
17bci0113@sjt418scs045:~/Documents$ ./a.out
Thread 3 starting...
Thread 3: c[0]= 0.000000
Thread 3: c[1]= 2.000000
Thread 3: c[2]= 4.000000
Thread 3: c[3]= 6.000000
Thread 3: c[4]= 8.000000
Thread 3: c[5]= 10.000000
Thread 3: c[6]= 12.000000
Thread 3: c[7]= 14.000000
Thread 3: c[8]= 16.000000
Thread 3: c[9]= 18.000000
Thread 3: c[10]= 20.000000
Thread 3: c[11]= 22.000000
Thread 3: c[12]= 24.000000
Thread 3: c[13]= 26.000000
Thread 3: c[14]= 28.000000
Thread 3: c[15]= 30.000000
Thread 3: c[16]= 32.000000
Thread 3: c[17]= 34.000000
Thread 3: c[18]= 36.000000
Thread 3: c[19]= 38.000000
Thread 3: c[20]= 40.000000
Thread 3: c[21]= 42.000000
Thread 3: c[22]= 44.000000
Thread 3: c[23]= 46.000000
Thread 3: c[24]= 48.000000
Thread 3: c[25]= 50.000000
Thread 3: c[26]= 52.000000
Thread 3: c[27]= 54.000000
Thread 3: c[28]= 56.000000
Thread 3: c[29]= 58.000000
Thread 3: c[30]= 60.000000
Thread 3: c[31]= 62.000000
Thread 3: c[32]= 64.000000
Thread 3: c[33]= 66.000000
Thread 3: c[34]= 68.000000
Thread 3: c[35]= 70.000000
Thread 3: c[36]= 72.000000
Thread 3: c[37]= 74.000000

```

```
Thread 3: c[38]= 76.000000
Thread 3: c[39]= 78.000000
Thread 3: c[40]= 80.000000
Thread 3: c[41]= 82.000000
Thread 3: c[42]= 84.000000
Thread 3: c[43]= 86.000000
Thread 3: c[44]= 88.000000
Thread 3: c[45]= 90.000000
Thread 3: c[46]= 92.000000
Thread 3: c[47]= 94.000000
Thread 3: c[48]= 96.000000
Thread 3: c[49]= 98.000000
Thread 3: c[50]= 100.000000
Thread 3: c[51]= 102.000000
Thread 3: c[52]= 104.000000
Thread 3: c[53]= 106.000000
Thread 3: c[54]= 108.000000
Thread 3: c[55]= 110.000000
Thread 3: c[56]= 112.000000
Thread 3: c[57]= 114.000000
Thread 3: c[58]= 116.000000
Thread 3: c[59]= 118.000000
Thread 3: c[60]= 120.000000
Thread 3: c[61]= 122.000000
Thread 3: c[62]= 124.000000
Thread 3: c[63]= 126.000000
Thread 3: c[64]= 128.000000
Thread 3: c[65]= 130.000000
Thread 3: c[66]= 132.000000
Thread 3: c[67]= 134.000000
Thread 3: c[68]= 136.000000
Thread 3: c[69]= 138.000000
Thread 3: c[70]= 140.000000
Thread 3: c[71]= 142.000000
Thread 3: c[72]= 144.000000
Thread 3: c[73]= 146.000000
Thread 3: c[74]= 148.000000
Thread 3: c[75]= 150.000000
Thread 3: c[76]= 152.000000
Thread 3: c[77]= 154.000000
Thread 3: c[78]= 156.000000
```

```
Thread 3: c[64]= 128.000000
Thread 3: c[65]= 130.000000
Thread 3: c[66]= 132.000000
Thread 3: c[67]= 134.000000
Thread 3: c[68]= 136.000000
Thread 3: c[69]= 138.000000
Thread 3: c[70]= 140.000000
Thread 3: c[71]= 142.000000
Thread 3: c[72]= 144.000000
Thread 3: c[73]= 146.000000
Thread 3: c[74]= 148.000000
Thread 3: c[75]= 150.000000
Thread 3: c[76]= 152.000000
Thread 3: c[77]= 154.000000
Thread 3: c[78]= 156.000000
Thread 3: c[79]= 158.000000
Thread 3: c[80]= 160.000000
Thread 3: c[81]= 162.000000
Thread 3: c[82]= 164.000000
Thread 3: c[83]= 166.000000
Thread 3: c[84]= 168.000000
Thread 3: c[85]= 170.000000
Thread 3: c[86]= 172.000000
Thread 3: c[87]= 174.000000
Thread 3: c[88]= 176.000000
Thread 3: c[89]= 178.000000
Thread 3: c[90]= 180.000000
Thread 3: c[91]= 182.000000
Thread 3: c[92]= 184.000000
Thread 3: c[93]= 186.000000
Thread 3: c[94]= 188.000000
Thread 3: c[95]= 190.000000
Thread 3: c[96]= 192.000000
Thread 3: c[97]= 194.000000
Thread 3: c[98]= 196.000000
Thread 3: c[99]= 198.000000
Thread 2 starting...
Number of threads = 4
Thread 0 starting...
Thread 1 starting...
17bci0113@sjt418scs045: ~/Documents$
```