Basics of OpenMP

Prepared By:
Sabbir Ahmed Sibli
ID: 20266027
M.Sc. in CSE, BRAC University

At a glance:

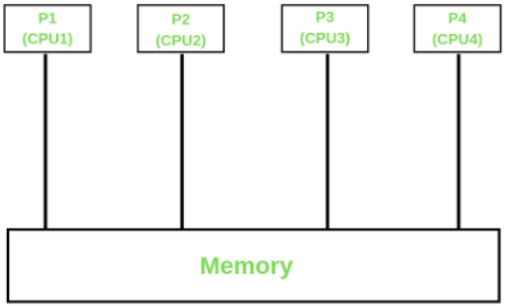
- Full form is **Open Multi-Processing**
- It is a technique of parallelizing a section(s) of C/C++/Fortran code.
- **Formal Definition:** OpenMP uses a portable, scalable model that gives programmers a simple and flexible interface for developing parallel applications for platforms that ranges from the normal desktop computer to the high-end supercomputers. (Source: geeksforgeeks)

Thread vs. Process

Process	Thread
Processes are heavyweight operations	Threads are lighter weight operations under a process.
ach process has its own memory space	Threads use the memory of the process they belong to
Processes don't share memory with other processes	Threads share memory with other threads of the same process

Parallel Memory Architecture

- **Shared memory:**



OpenMP comes under the shared memory concept

Steps to Install OpenMP

STEP 1: Check the GCC version of the compiler
Firstly, we have to check if the gcc is installed in the machine or not by following command:

gcc –version

If the system doesn't have the GCC compiler, we can use the following command

sudo apt install gcc

STEP 2: Configuring OpenMP
We can check whether the OpenMP features are configured into our compiler or not, using the command

echo |cpp -fopenmp -dM |grep -i open

If OpenMP is not featured in the compiler, we can configure it use using the command

sudo apt install libomp-dev

STEP 3: Setting the number of threads In OpenMP
Before running the code, we can initialize the number of threads to be executed using the following command. Here, we set the number of threads to be getting executed to be 8 threads.

export OMP_NUM_THREADS=8

Sample code to run with OpenMP

```c
#include <stdio.h>
#include <omp.h>
int main()
{
    #pragma omp parallel
    {
        printf("Printing from thread: %d\n", omp_get_thread_num());
    }
    return 0;
}
```

Compile the code:

gcc -o OutputFile -fopenmp CodeName.c

Execute the code:

./OutputFile