|  |  |
| --- | --- |
| ФИО Студента | Витомсков Алексей Вадимович |
| Группа | КЭ-220 |
| Курс | Технологии параллельного программирования |
| Задание | Задание 31. Программа «I am» |
| Отчет по выполнению практического задания | |

*1.* ***Напишите программу, в которой в каждом процессе создается n нитей. Каждая нить должна выводить на экран свой номер, номер процесса родителя и общее количество нитей во всех процессах в следующем формате:***

*I am <Номер нити> thread from <Номер родительского процесса> process.*

*Number of hybrid threads = <Количество нитей \* Количество процессов>.****Входные данные:*** *целое число n – количество нитей, которые должны быть  
запущены.****Выходные данные:*** *строки вида «I am <Номер нити> thread from <Номер  
родительского процесса> process. Number of hybrid threads = <Количество нитей \* Количество процессов>».*

Код программы:

|  |
| --- |
| #include <mpi.h>  #include <omp.h>  #include <stdio.h>  #include <windows.h>  int main(int argc, char\* argv[])  {  int x;  x = atoi(argv[1]);  printf("num\_theards: %d\n", x);  MPI\_Init(&argc, &argv);    int rank\_mpi; //номер текущего процесса  MPI\_Comm\_rank(MPI\_COMM\_WORLD, &rank\_mpi);  int size\_mpi; //Количество процессов в приложении  MPI\_Comm\_size(MPI\_COMM\_WORLD, &size\_mpi);  omp\_set\_num\_threads(x);  #pragma omp parallel  {  int rank\_omp, size\_omp;  rank\_omp = omp\_get\_thread\_num();  size\_omp = omp\_get\_num\_threads();  printf("I am %d thread from %d> process. Number of hybrid threads = % d.\n", rank\_omp, rank\_mpi, size\_omp \* size\_mpi);  }  MPI\_Finalize();  return 0;  } |

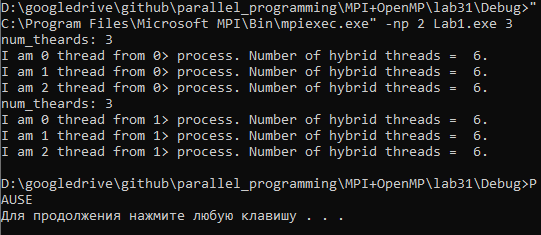
*Bat файл для запуска программы:*

|  |
| --- |
| CMD /c  CLS &  "C:\Program Files\Microsoft MPI\Bin\mpiexec.exe" -np 3 Lab1.exe 4 &  PAUSE |

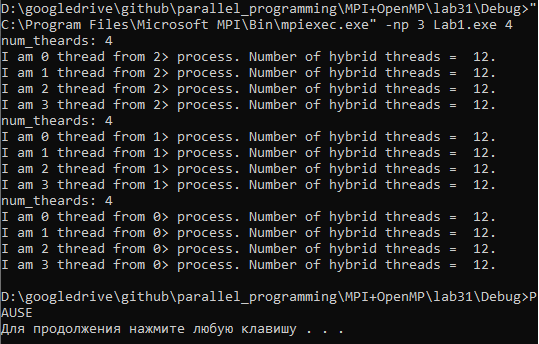
В программу mpiexec.exe передается число процессов (3), с которым запускается программа Lab1.exe. В свою очередь, в программу Lab1.exe передается число нитей в каждом процессе. Программа Lab1.exe выводит сообщение о заданном количестве нитей. Затем в параллельном регионе выводится сообщение “I am…” для каждой нити.

*Результат работы программы:*

***число потоков = 2, число нитей = 3***



***Число потоков = 3, число нитей = 4***



Интересно то, что строка num\_threads = 4 вывелась в каждом процессе, хотя в программе функция print расположена перед инструкцией MPI\_Init(). Таким образом, программа (функция main) выполняется в каждом процессе полностью.

***5. Ответить на вопросы***

1) Можно ли в коде данной программы поменять порядок вложения блоков MPI и OpenMP (MPI\_Init-MPI\_Finalize внутри параллельного региона OpenMP)? Почему?

Поменять нельзя. Программа изначально запускается на n процессах MPI. Соответственно блок MPI\_Init должен принять параметры, переданные в консоли. Запустить сначала n нитей, а затем в каждой нити n процессов не получится. Если посмотреть с другой стороны, гибридное программирование предназначено для систем с раcпределенной оперативной памятью, где каждый кластер представлен системой с общей памятью. То есть нити OpenMP запускаются внутри MPI процессов. Для систем другой архитектуры должны быть использованы другие технологии.

2) Почему так отличаются интерфейсы функций получения rank и size в MPI и OpenMP?

В результате выполнения функции MPI\_Init() создается группа процессов, в которую помещаются все процессы приложения, и создается область связи, описываемая предопределенным коммуникатором MPI\_COMM\_WORLD. Эта область связи объединяет все процессы-приложения. Число процессов и номер процесса определяется в конкретной области связи. Поэтому функции rank и size в MPI требуют передачу дополнительного параметра.