

Специализированные языки и  
библиотеки для параллельных  
вычислений. MPI. Разработка  
параллельного алгоритма с  
использованием  
коллективных операций

Презентацию выполнил:  
студент группы 06-208  
Тонков Артем

# Функции коллективного взаимодействия

Набор операций типа точка-точка является достаточным для программирования любых алгоритмов, однако MPI вряд ли бы завоевал такую популярность, если бы ограничивался только этим набором коммуникационных операций. Одной из наиболее привлекательных сторон MPI является наличие широкого набора коллективных операций, которые берут на себя выполнение наиболее часто встречающихся при программировании действий.

Главное отличие коллективных операций от операций типа точка-точка состоит в том, что в них всегда участвуют все процессы, связанные с некоторым коммутатором. Несоблюдение этого правила приводит либо к аварийному завершению задачи, либо к еще более неприятному зависанию задачи.

# Отличительные особенности коллективных операций:

- Коллективные коммуникации не взаимодействуют с коммуникациями типа точка–точка.
- Коллективные коммуникации выполняются в режиме с блокировкой. Возврат из подпрограммы в каждом процессе происходит тогда, когда его участие в коллективной операции завершилось, однако это не означает, что другие процессы завершили операцию.
- Количество получаемых данных должно быть равно количеству посланных данных.
- Типы элементов посылаемых и получаемых сообщений должны совпадать.
- Сообщения не имеют идентификаторов.

# Программа

student@Alpha: /home/student/parallel\_programming/06-208/Vital\_k

File: tema.c

GNU nano 2.2.2

```
#include "mpi.h"
#include <math.h>
#include <stdio.h>
#include <stdlib.h>

void main(int argc, char *argv[])
{
double x[100], y[100];
double res, p_res=0.0;
MPI_Status status;
int n, myid, numprocs, i, N;
N=100;

MPI_Init(&argc, &argv);
MPI_Comm_size(MPI_COMM_WORLD, &numprocs);
MPI_Comm_rank(MPI_COMM_WORLD, &myid);
if(0==myid)
{
for (i=0; i<100; i++)
{x[i] = (i+1);
y[i] = (i+3);
}

MPI_Bcast(x,N,MPI_DOUBLE,0,MPI_COMM_WORLD);
MPI_Bcast(y,N,MPI_DOUBLE,0,MPI_COMM_WORLD);

for (i=myid*N/numprocs; i<(myid + 1)*N/numprocs; i++)
p_res = p_res + x[i] * y[i];
if (0 ==myid)
{
for (i=1; i<numprocs; i++)
MPI_Recv(&res,1,MPI_DOUBLE,i,MPI_ANY_TAG,MPI_COMM_WORLD,&status);
p_res = p_res+res;
}
else
MPI_Send(&p_res,1,MPI_DOUBLE,0,0,MPI_COMM_WORLD);
if (0==myid)
printf("\nInner product = %10.2f\n",p_res);
MPI_Finalize();
}
```

Get Help  
Exit

WriteOut  
Justify

Read File  
where Is

Read 46 lines

Prev Page  
Next Page

Cut Text  
UnCut Text

Cur Pos  
To Spell



A fluffy, light-colored kitten is sitting upright in a dark environment. To its right is a lit lamp with a yellow, conical shade and a glass base. The lamp's light illuminates the kitten's face and the surrounding area. The kitten has large, dark eyes and a slightly open mouth, looking directly at the viewer.

**СПАСИБО ЗА ВНИМАНИЕ**