

第 9 讲 消息传递编程

- 1、试考虑下述代码段中通信体的使用;

```
process 0:
MPI_Send(msg1,count1,MPI_INT,tag1,comm1);
parallel_fft(...);
process 1:
MPI_Recv(msg1,count1,MPI_INT,tag1,comm1);
parallel_fft(...);
```

(1)试分析上述代码段的计算功能。

(2)如果在 parallel_fft(...)中又包含了另一个发送程序:

```
if(my_rank == 0) MPI_Send(msg2,count1,MPI_INT,1,tag2,comm2);
如果没有通信体则会发生什么情况?
```

- 2、填上空白处,使下面两代码段完全等效:

```
(1)float data[1024];
MPI_Datatype floatype;
MPI_Type_vector(10,1,32,MPI_FLOAT,&floatype);
MPI_Type_commit(&floatype);
MPI_Send(data,1,floatype,dest,tag,MPI_COMM_WORLD);
MPI_Type_free(&floatype);

(2)float data[1024],buff[10];
for( _____ ; _____ ; i++) buff[i] = data [ _____ ]
MPI_Send(buff, _____ , MPI_FLOAT, _____ , _____ , _____ );
```

- 3、下面是 PVM 环境下的 hello 程序,它是一个 host/node 程序,试分析其工作过程。

```
/**PVM 主机/节点编程的 hello 代码段*/
/*host 程序 hello.c*/
#include<stdio.h>
#include "pvm3.h"
main()
{
    int cc, tid;
    char buf[100];
    printf("i'm t%x \n", pvm_myid());
    cc=pvm_spwan("hello_other", (char **)0, 0, "", 1, &tid);
    if (cc == 1)
    {
        cc=pvm_recv(-1, -1);
        pvm_buinfo(cc, (int *)0, (int *)0, &tid);
        pvm_upkstr(buf);
        printf("from t%x: %s \n", tid, buf);
    } else
        printf("can't start hello_other \n");
}
```

```

        pvm_exit();
    }

/*node 程序 hello_other.c*/
#include "pvm3.h"
#include "string.h"

main()
{
    int ptid;
    char buf[100];
    ptid=pvm_parent();
    strcpy(buf, "hello, world from");
    gethostname(buf+strlen(buf), 64);
    pvm_initsend(PvmDataDefault);
    pvm_pkstr(buf);
    pvm_send(ptid, 1);
    pvm_exit();
    exit(0);
}

```

- 4、计算 π 可以通过 MPI 程序。为了加深读者对 MPI 的学习和理解，下面给出用 FORTRAN 90 语言的实现版本。试分析其工作过程。

```

/*计算 $\pi$ 的 MPI (F90) 编程代码段*/
program main
usempi
double precision PI25DT
parameter (PI25DT=3.141592653589793238462643d0)
double precision mypi, pi, h, sum, x, f, a
integer n, myid, numprocs, i, rc
! function to integrate
f(a) = 4.d0/(1.d0+a*a)
call MPI_INIT(ierr)
call MPI_COMM_RANK(MPI_COMM_WORLD, myid, ierr)
call MPI_COMM_SIZE(MPI_COMM_WORLD, numprocs, ierr)
print *, 'process', myid, 'of', numprocs, 'is alive'
sizetype=1
sumtype=2
do
    if (myid.eq.0) then
        write(6, 98)
        format('Enter the number of intervals: (0 quit)')
        read(5, 99)n
        format(i10)

```

```

endif
call MPI_BCAST(n, 1, MPI_INTEGER, 0, MPI_COMM_WORLD, ierr)
! check for quit signal
if (n.le.0) exit
! calculate the interval size
h=1.0d0/n
sum=0.0d0
do i=myid+1, n, numprocs
    x=h*(dble(i)-0.5d0)
    sum=sum+f(x)
enddo
mypi=h*sum
! collect all the partial sums
call MPI_REDUCE(mypi, pi, 1, MPI_DOUBLE_PRECISION, MPI_SUM, 0, &
MPI_COMM_WORLD, ierr)
! process 0 prints the answer
if (myid.eq.0) then
    write (6, 97) pi, abs(pi-PI25DT)
    format ('pi is approximatetly :', F18.16, & 'Error is: ', F18.16)
endif
enddo
call MPI_FINALIZE(rc)
stop
end

```

5. 什么是 MPI 的消息、数据类型、通信域？

6. 什么是 MPI 的阻塞通信和非阻塞通信？

7. MPI 中点到点通信模式有哪些？

8. MPI 中有哪些典型的群集通信模式？

9. 计算圆周率 π 的串行代码段如下：

```

h=1.0/n;
sum =0.0;
for (i=0;i<n;i++) {
    x=h*(i+0.5);
    sum=sum+4.0/(1+x*x);
}
pi=h*sum;

```

用 MPI 编程语言，写出计算 π 的并行代码段。

10. 简述 MPI 和 OpenMP 各自的适用场合和异同。