CS546 “Parallel and Distributed Processing”

Homework 5
First name: Guillermo
Last Name: de la Puente
CWID: A20314328
Due date: 4/28/2014

Problem 1

(a) a(2,:)=d

D:

d1	d2	d3	d4	d5

A:

a11	a12	a13	a14	a15
d1	d2	d3	d4	d5
a31	a32	a33	a34	a35
a41	a42	a43	a44	a45
a51	a52	a53	a54	a55

(b) a(1:3,:)=b(2:4,:)

B:

b11	b12	b13	b14	b15
b21	b22	b23	b24	b25
b31	b32	b33	b34	b35
b41	b42	b43	b44	b45
b51	b52	b53	b54	b55

A:

b11	b12	b13	b14	b15
b21	b22	b23	b24	b25
b31	b32	b33	b34	b35
a41	a42	a43	a44	a45
a51	a52	a53	a54	a55

(c) where (b.eq.c) a=c

Imagine that the values x are those who are equal between b and c. They have been placed randomly.

B:

b11	b12	b13	b14	b15
b21	x1	b23	x2	b25
b31	b32	b33	b34	b35
b41	b42	x3	b44	b45
b51	b52	b53	b54	x4

C:

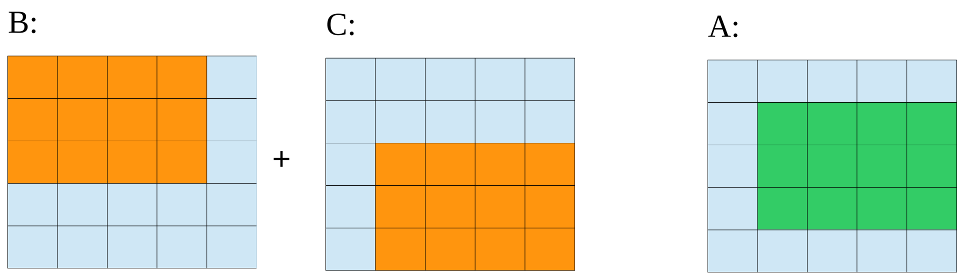
c11	c12	c13	c14	c15
c21	x1	c23	x2	c25
c31	c32	c33	c34	c35
c41	c42	x3	c44	c45
c51	c52	c53	c54	x4

A:

a11	a12	a13	a14	a15
a21	x1	a23	x2	a25
a31	a32	a33	a34	a35
a41	a42	x3	a44	a45
a51	a52	a53	a54	x4

(d) forall (i=2:4,j=2:5) a(i,j)=b(i-1,j-1)+c(i+1,j)

The result of adding the orange part of b and the orange part of c is placed in the green part of a.



(e) forall (i=1:5,j=1:5) b(i,j)=(i+j-1)

There might be a typo in this part of the assignment. However, the result of this operation is:

B:

1	2	3	4	5
2	3	4	5	6
3	4	5	6	7
4	5	6	7	8
5	6	7	8	9

(f) forall (j=1:5) d(j)=sum(c(1:4,j),dim=1)

Each of the colored columns in c is summed and assigned in d.

C:

c11	c12	c13	c14	c15
c21	c22	c23	c24	c25
c31	c32	c33	c34	c35
c41	c42	c43	c44	c45
c51	c52	c53	c54	c55

D:

c11+ c21+ c31+ c41	c12+ c22+ c32+ c42	c13+ c23+ c33+ c43	c14+ c24+ c34+ c44	c15+ c25+ c35+ c45

(g) a=spread(d,dim=2,ncopies=5)

A:

d1	d1	d1	d1	d1
d2	d2	d2	d2	d2
d3	d3	d3	d3	d3
d4	d4	d4	d4	d4
d5	d5	d5	d5	d5

(h) b=spread(d,dim=1,ncopies=5)

B:

d1	d2	d3	d4	d5
d1	d2	d3	d4	d5
d1	d2	d3	d4	d5
d1	d2	d3	d4	d5
d1	d2	d3	d4	d5

(i) a=cshift(b,dim=1,shift=3)

A:

b41	b42	b43	b44	b45
b51	b52	b53	b54	b55
b11	b12	b13	b14	b15
b21	b22	b23	b24	b25
b31	b32	b33	b34	b35

(j) d=sum(spread(d,dim=1,ncopies=5),dim=2)

spread(d,dim=1,ncopies=5):

d1	d2	d3	d4	d5
d1	d2	d3	d4	d5
d1	d2	d3	d4	d5
d1	d2	d3	d4	d5
d1	d2	d3	d4	d5

D:

d1+d2+ d3+d4+ d5	d1+d2+ d3+d4+ d5	d1+d2+ d3+d4+ d5	d1+d2+ d3+d4+ d5	d1+d2+ d3+d4+ d5

Problem 2

(a) given a 2-dimensional matrix a(100,100), zero out the upper diagonal elements

```
forall(i=1:100,j=1:100, i.gt.j) a(i,j)=0
```

(b) given a 2-dimensional matrix a(100,100), transpose it and assign to matrix b(100,100)

```
real b(100,100)
b = transpose (a)
```

(c) given a 1-dimensional array a(100), assign it to a two dimensional array b(100,5) by replicating a() column-wise 5 times

```
real b(100,5)
b = spread(a, dim=2, ncopies=5)
```

(d) given a 2-dimensional array a(100,100), assign it to an array b(100,100) such that each element is circularly shifted left by two columns and down by 1 row in the result matrix

```
real b(100,100)
b = cshift( cshift(a, dim=1, shift=1) , dim=2, shift=-2)
```

(e) perform the following data transfer between arrays a(8) and b(4)

```
forall(i=1:4) b(i)=a(2*i)
```

Problem 3

(a) array a(18) distributed across four processors P1,...,P4 …

```
!HPF$ PROCESSORS PR(4)
!HPF$ DISTRIBUTE a(BLOCK) ONTO (PR)
```

(b) partition array a(12) across four processors in a blocked distribution, and...

```
      !HPF$ PROCESSORS PR(4)
      !HPF$ ALIGN a(:) WITH b(*,:)
      !HPF$ DISTRIBUTE a(BLOCK), b(*,BLOCK)
```

Problem 4

Just like in the previous assignments, the loop to be parallelized is the second one. The row distribution is made in a cyclic way.

What we do is simply parallelize the loop where every processor will calculate the multiplying factor and affect the whole row with it.

There could be a faster way using temporary arrays that are aligned with A, but that might bring new delays of data transfer.

```
!   Gauss Elimination without pivoting

    program gauss
    integer n,row,col,norm
    parameter (n=256)
    real  X(n),B(n),A(n,n),multiplier
    real*8 elapsed1, elapsed2,rtc,elapsedp1,elapsedp2

! We are going to define 4 processors for example
!HPF$ PROCESSORS PR(4)
!HPF$ DISTRIBUTE A(CICLYC,*)

    elapsed1 = rtc()

! --------- Initialize all elements to Random Values.
    do row = 1,n
       do col = 1,n
          A(row,col) = (1.0 * irand())/32768.0
       enddo
       B(row) = (1.0 * irand())/32768.0
    enddo
```

```
! --------------------------------------------------
! This loop is parallelized

    elapsedp1 = rtc()
    do norm = 1 , n-1
! The forall is the loop that we parallelized
       forall (row= norm+1:n)
          multiplier=A(row,norm)/A(norm,norm)
          do col = norm,n
             A(row,col) = A(row,col) - A(norm,col)* multiplier
          enddo
          B(row) = B(row) - B(norm) * multiplier
       end forall
    enddo
    elapsedp2 = rtc()

! -------------------------------- backsubstitute
    do row=n-1,1,-1
       X(row) = B(row)
       do col = n-1,row+1,-1
          X(row) = X(row) - A(row,col) * X(col)
       enddo
       X(row) = X(row)/ A(row,row)
    enddo
    elapsed2 = rtc()

! -----------------------Check correctness of code
    do row=1,n
       do col=1,row -1
          if ( A(row,col) .GT. 1e-3 ) print *,"Error in",row,col,A(row,col)
       enddo
    enddo
    print *,"Elapsed Time", elapsed2 - elapsed1
    print *,"Elapsed Time in elimination phase", elapsedp2 - elapsedp1
    stop
    end
```