Range: Ring **Z**^2L (L is a integer) or Finite Field **F**q (q is a prime)

Note: x to denote a scalar; **x** to denote a vector; **X** to denote a matrix

#case1 : scalar

ADD: z=x+y

INPUT

S0：<x>0, <y>0; S1：<x>1, <y>1

COMPUTE

S0: Compute <z>0=<x>0+ <y>0, locally

S1: Compute <z>1=<x>1+ <y>1, locally

OUTPUT

S0: <z>0; S1: <z>1

MUL: z=x.y

INPUT

S0：<x>0, <y>0, (<a>0, <b>0, <c>0)

S1：<x>1, <y>1, (<a>1, <b>1, <c>1)

COMPUTE

Si: Compute <e>i=<x>i - <a>i; <f>i=<y>i - <b>i; , send <e>i , <f>i to S1-i;

and set e= <e>0+<e>1, f=<f>0+<f>1;

Compute <z>i=i.e.f+f.<a>i+e.<b>i+<c>i

i.e.

S0: compute <e>0=<x>0 - <a>0; <f>0=<y>0 - <b>0; , send <e>0 , <f>0 to S1;

S1: compute <e>1=<x>1 - <a>1; <f>1=<y>1 - <b>1; , send <e>1 , <f>1 to S0;

S0: reconstruct e= <e>0+<e>1, f=<f>0+<f>1;

S1: reconstruct e= <e>0+<e>1, f=<f>0+<f>1;

S0: compute <z>0=f.<a>0+e.<b>0+<c>0

S1: compute <z>1=e.f+f.<a>1+e.<b>1+<c>1

OUTPUT

S0: <z>0; S1: <z>1

CORRECTNESS

z=<z>0+ <z>1=e.f+f.a+e.b+c=(x-a)(y-b)+(y-b)a+(x-a)b+ab=x.y

#case2 : matrix

ADD: Z=X+Y

INPUT

S0**:** <**X**>0, <**Y**>0; S1**:** <**X**>1, <**Y**>1

COMPUTE

S0: Compute <**Z**>0=<**X**>0+ <**Y**>0, locally

S1: Compute <**Z**>1=<**X**>1+ <**Y**>1, locally

OUTPUT

S0: <**Z**>0; S1: <**Z**>1

MUL: Z=X.Y (For example: **Z**mx1=**X**mxd.**Y**dx1)

INPUT

S0**:** <**X**>0 mxd, <**Y**>0 dxn, (<**A**>0 mxd, <**B**>0 dxn, <**C**>0 mxn)

S1**:** <**X**>1 mxd, <**Y**>1 dxn, (<**A**>1 mxd, <**B**>1 dxn, <**C**>1 mxn)

COMPUTE

Si: Compute <**E**>i=<**X**>i - <**A**>i; <**F**>i=<**Y**>i - <**B**>i; , send <**E**>i , <**F**>i to S1-i;

and set **E**= <**E**>0+<**E**>1, **F**=<**F**>0+<**F**>1;

Compute <**Z**>i=i.**E**.**F**+<**A**>i.**F**+**E**.<**B**>i+<**C**>i

i.e.

S0: compute <**E**>0 mxd=<**X**>0mxd - <**A**>0mxd; <**F**>0dxn=<y>0dxn - <**B**>0dxn;

send <**E**>0 mxd, <**F**>0 dxnto S1;

S1: compute <**E**>1mxd=<**X**>1mxd - <**A**>1mxd; <**F**>1dxn=<y>1dxn - <**B**>1dxn;

send <**E**>1 mxd, <**F**>1 dx1 to S0;

S0: reconstruct **E** mxd= <**E**>0mxd+<**E**>1mxd, **F**dxn=<**F**>0dxn+<**F**>1dxn;

S1: reconstruct **E** mxd= <**E**>0mxd+<**E**>1mxd, **F**dxn=<**F**>0dxn+<**F**>1dxn;

S0: compute <**Z**>0mxn=<**A**>0mxd.**F**dxn+**E**mxd.<**B**>0dxn+<**C**>0mxn

S1: compute <**Z**>1mxn=**E**mxd.**F**dxn+<**A**>1mxd.**F**dx1+**E**mxd.<**B**>1dxn+<**C**>1mxn

OUTPUT

S0: <**Z**>0mxn; S1: <**Z**>1mxn

CORRECTNESS

Z=<**Z**>0+ <**Z**>1=**E**.**F**+**F**.**A**+**E**.**B**+**C**=**(X-A)(Y-B)+(Y-B)A+(X-A)B+AB=X.Y**