左右两个叉子一起申请

Int a[5]={0},I

Semaphore　b[5]={0},s;

procedure test(I)

{if(a[i]==1&&a[(i-1)%5]!=2&& a[(i+1)%5]!=2)

{a[i]=2;v(b[i]);}}

Procedure philosopher(I)

{p(s);

a[i]=1;test[i];

V[s];

P(b[i]);

pick\_up\_fork(I);pick\_up\_fork((I+1)mod 5);

{Eating};

put\_down\_fork(I);put\_down\_fork((I+1)mod 5);

P(s);

A[i]=0;test((i-1)%5); 0;test((i+1)%5);

V(s);

相邻哲学家申请不同方向的叉子(奇数号哲学家先申请左边的叉子，再申请右边的叉子；偶数号哲学家先申请右边的叉子，再申请左边的叉子）

Semaphore　S[5]={1};

if(i%2==1)

{p(s[i])

pick\_up\_fork(I);

p(s[(i+1)%5)；

pick\_up\_fork((I+1)mod 5);

{Eating};

put\_down\_fork(I);put\_down\_fork((I+1)mod 5);

V(s[i])；

V(s[(i+1)%5)；

}

else {p(s[(i+1)%5)；

pick\_up\_fork((I+1)mod 5);

p(s[i])

pick\_up\_fork(I);

p(s[(i+1)%5);

{Eating};

put\_down\_fork(I);put\_down\_fork((I+1)mod 5);

V(s[i])；

V(s[(i+1)%5)；

}