

```
for (expr 1 ; expr 2 ; expr 3 ) { };
```

或

```
for (expr 1 ; expr 2 ; expr 3 ):  
    statement ;  
endfor ;
```

for 循环有 { } 和 : 两种写法

expr 1 是在所有 statement 之前只执行一次的初始化条件 ; expr 2 是每次 statement 前执行一次的 boolean 求值, 当 expr 2 为 true 时 statement 才执行 ;

1

expr 3 时每次 statement 结束后附加执行的代码, 通常为递增、递减表达式

e.g.

```
for ($i=0 ; $i<5 ; $i++) { echo $i };
```

```
 $\Leftrightarrow$  $i=0 ; for( ; $i<5 ; $i++) { echo $i };
```

```
 $\Leftrightarrow$  $i=0 ; for( ; ; $i++) { for($i>4){break;}  
echo $i };
```

```
 $\Leftrightarrow$  $i=0 ; for( ; ; )
```

```

{ for ($i > 4) break ;
  echo $i ;

  i++ ;
}

```

\Leftrightarrow `for ($i=0 ; $i<5 ; print $i++) ;`

此处只能用 `print` 而不能
用 `echo` 语句，否则语法错误，

原因未知；`print $i++` \Leftrightarrow `print $i ; $i++`；

注意对于 `$i < count(array)` 这样的 `expr2`，
每次判断都调用一次 `count()` 函数，很
慢

消耗资源,可在初始时将 count_i 结果存于一个变量,再用该变量来和 σ_i 比较判定.