PERL SEMOP FUNCTION

Description

This function performs the semaphore operations defined by OPSTRING on the semaphore ID associated with KEY. OPSTRING should be a packed array of semop structures, and each structure can be generated with.

Syntax

Following is the simple syntax for this function −

```
semop KEY, OPSTRING
```

Return Value

This function returns 0 on failure and 1 on success.

Example

Following is the example code showing its basic usage, creating a semaphore and incrementing its value −

```
#!/usr/bin/perl -w

# Assume this file name is left.pl

use IPC::SysV;

#use these next two lines if the previous use fails.
eval 'sub IPC_CREAT {0001000}' unless defined &IPC_CREAT;
eval 'sub IPC_EXCL {0002000}'  unless defined &IPC_EXCL;
eval 'sub IPC_RMID {0}'        unless defined &IPC_RMID;

$key = 1066;

$| = 1;
$num = 0;
$flag = 0;

# Create the semaphor
$id = semget ( $key, 1, &IPC_EXCL|&IPC_CREAT|0777 ) or
 die "Can't semget: $!";
foreach( 1..5) {
 $op  = 0;
 $operation = pack( "s*", $num, $op, $flags );
 semop( $id, $operation ) or die "Can't semop: $! ";
 print "Left....\n";
 sleep 1;
 $op = 2;
 $operation = pack( "s*", $num, $op, $flags );
 # add 2 to the semaphore ( now 2 )
 semop( $id, $operation ) or die "Can't semop $! ";
}
semctl (  $id, 0, &IPC_RMID, 0 );
```

Run the above program in background using $left.pl& and write following another program. Here Left sets the semaphore to 2 and Right prints Right and resets the semaphore to 0. This continues until Left finishes its loop after which it destroys the semaphore with semctl()

```
#!/usr/bin/perl -w
```

```
# Assume this file name is right.pl

$key = 1066;

$| = 1;
$num = 0;
$flags = 0;

# Identify the semaphore created by left.
$id = semget( $key, 1, 0 ) or die ("Can't semgt : $!" );

foreach( 1..5){
 $op = -1;
 $operation =  pack( "s*", $num, $op, $flags );
 # Add -1 to the semaphore (now 1)
 semop( $id, $operation ) or die " Can't semop $!";
 print "Right....\n";
 sleep 1;
 $operation = pack( "s*", $num, $op, $flags );
  # Add -1 to the semaphore (now  0)
 semop( $id, $operation ) or die "Can't semop $! ";
}
```

Now run right.pl and it will produce the following results −

```
Right....
Left....
Right....
Left....
Right....
Left....
Right....
Left....
Right....
Left....
```