P1.c

1 //when sem\_op is a positive intiger

2 #include<stdio.h>

3 #include<sys/sem.h>

4 main()

5 {

6 int id;

7 struct sembuf v;

8 id=semget(5,3,IPC\_CREAT|0644);

9 if(id<0)

10 {

11 perror("semget");

12 return;

13 }

14 v.sem\_num=0; //0 (or)1(or)2

15 v.sem\_op=1; //semval=semval+sem\_op

16 v.sem\_flg=0;

17 printf("before..\n");

18 semop(id,&v,1);

19 printf("after semop..\n");

20 }

P2.c

1 //when sem\_op is less then zero

2 #include<stdio.h>

3 #include<sys/sem.h>

4 main()

5 {

6 int id;

7 struct sembuf v;

8 id=semget(5,3,IPC\_CREAT|0644);

9 if(id<0)

10 {

11 perror("semget");

12 return;

13 }

14 v.sem\_num=0; //0 (or)1(or)2

15 v.sem\_op=-1; //semval=semval-|sem\_op|

16 //v.sem\_flg=0;

17 v.sem\_flg=SEM\_UNDO;

18 printf("before semop..\n");

19 semop(id,&v,1);

20 printf("after semop..\n");

21 printf("accessing resource...\n");

22 while(1);

23 }

P3.c

1 //when sem\_op is less then zero

2 #include<stdio.h>

3 #include<sys/sem.h>

4 main()

5 {

6 int id;

7 struct sembuf v;

8 id=semget(5,3,IPC\_CREAT|0644);

9

10 if(id<0)

11 {

12 perror("semget");

13 return;

14 }

15 v.sem\_num=0; //0 (or)1(or)2

16 v.sem\_op=-1; //semval=semval-|sem\_op|

17 //v.sem\_flg=0;

18 v.sem\_flg=SEM\_UNDO;

19 printf("before semop..\n");

20 semop(id,&v,1);

21 printf("after semop..\n");

22 printf("accessing resource...\n");

23 while(1);

24 }

P4.c

1 //when sem\_op is less then zero

2 #include<stdio.h>

3 #include<sys/sem.h>

4 main()

5 {

6 int id;

7 struct sembuf v;

8 id=semget(5,3,IPC\_CREAT|0644);

9

10 if(id<0)

11 {

12 perror("semget");

13 return;

14 }

15 v.sem\_num=0; //0 (or)1(or)2

16 v.sem\_op=-1; //semval=semval-|sem\_op|

17 //v.sem\_flg=0;

18 v.sem\_flg=SEM\_UNDO;

19 printf("before semop..\n");

20 semop(id,&v,1);

21 printf("after semop..\n");

22 printf("accessing resource...\n");

23 while(1);

24 }

P5.c

1 //when sem\_op is less then zero

2 #include<stdio.h>

3 #include<sys/sem.h>

4 main()

5 {

6 int id;

7 struct sembuf v;

8 id=semget(5,3,IPC\_CREAT|0644);

9

10 if(id<0)

11 {

12 perror("semget");

13 return;

14 }

15 v.sem\_num=0; //0 (or)1(or)2

16 v.sem\_op=-1; //semval=semval-|sem\_op|

17 v.sem\_flg=0;

18 printf("before semop..\n");

19 semop(id,&v,1);

20 printf("after semop..\n");

21 printf("accessing resource...\n");

22 while(1);

23 }