Sono degli interi mantenoti dal Kernel con valore > 0. Dei semafori si pob · impostarne il valore · aggiungere on numero al valore del semaforo. · sottrarre un numero al valore del semaforo. · aspettare che il valore sia O. I semafori vengono allocati in groppi di semafori, il cui numero deve essere specificato in sempetti; CREAZIQUE SETAFORO Il semaforo viene creato can il liabus espripazioni i tra sempetti in coi si pob specuficate · Key: la chiave univora del semaforo. · nsems: · creazione: Il numero dei semafori da creare per - cannessione: Il # clese essere \(\substantia \) dia dimensione del si - semfla: flag aggiuntivi per permessi - cannessione: Il momero dei semafori da creare per - IPC creazione - IPC		MAFO RI >				
impostarne il Valore impostarne il Valore impostarne il Valore impostarne il Valore impostarne in numero al valore del semaforo. sottrarre un numero al valore del semaforo. T semafori vengono allocati in groppi chi semafori, il cui numuro deve essere specificato in semgetti; CREAZIONE SETAFORO TI semaforo viene creato can il "Issua e varyrege. Er / rec venalulity / Semget in coi si puó specificare insems: - creazione: Ti numero dei semafori da creare per quell'insiene - connessione: Ti numero dei semafori da creare per e di medifica dal comportamento insems: - creazione: Ti numero dei semafori da creare per odella creazione - TPC CREAT: se non esiste crea insemi in modifica dal comportamento comportamento insemi semi con param - TPC CREAT: se non esiste crea	Sor	ro degli	interi manter	roti dal Kernel	con valo	se ≥0.
impostarne il Valore oggivagere on numero al valore del semaforo. sottrarre un numero al valore del semaforo. aspettare che il valore sia 0. I semafori vengono allocati in groppi di semafori, il cui nonevo deve essere sperificato in semget(); Creazione Setraforo Il semaforo viene creato can il "essue egy/spenio" " su prindi" i y " semafori vengo creato can il "essue egy/spenio" " su prindi" i y " semaforo viene creato can il "essue egy/spenio" " su prindi" i y " semaforo viene creato can il "essue egy/spenio" " su prindi" i y " semaforo viene creato can il "essue egy/spenio" " su prindi" i y " semaforo viene creato can il "essue egy/spenio" o " su prindi" i y " semaforo viene creato can il "essue egy/spenio" o " su prindi" i y " semaforo viene creato can il "essue egy/spenio" o " su prindi" i y " semaforo viene creato can il "essue egy/spenio" o " su prindi" i y " semaforo viene creato can il "essue egy/spenio" o " su prindi" i y " e nsems: - creazione: Il numero dei semafori da creare per - semifig: flag aggiuntivi per permessi - semifig: flag aggiuntivi per permessi - della creazione - IPC creati: se non esiste, crea - insemio egy/spenio " se seasiale già, estore GESTIONE SETIAFORO **La seasiality / **Locatione egy/spenio " se seasiality / **Locatione egy/spenio "						Process A Process B
• configurate on numero al valore del semaforo. • zottrare on numero al valore del semaforo. • zottrare on numero al valore del semaforo. • zottrare che il valore sia O. I semafori vengono allocati in groppi di semafori, il cui numero deve essere specificato in semget(); CREAZIONE SETAFORO > Il semaforo viene creato con il tiente organizati in sungiore set identifiar co sociali. • Rey: la chiave univoca del semaforo. • nsems: • creazione: Il numero dei semafori da aleare per quell'insiene • semfla: flag aggiuntivi per permessi • cannessione: Il at deve essere C alla dimensione del set esemfla: flag aggiuntivi per permessi • Teccepent: se non esiste, crea • Ipor cepent: se non esiste, crea • Ipor cepent: se non esiste, crea • Icanida creazione • TCA-EXCL: se esiste già, errore • semnom: • semnom: • semnom: • Semnom: • Tott i semafori del groppo • cmd: Speufica l'operazione			\			Initialize semaphore to 0
· softrare un numero al valore del semaforo. · aspettare che il valore sia 0. I semafori vengono allocalti in groppi chi semafori, il cai numero deve essere specificato in semget(1); CREAZIONE SETAFORO Il semaforo viene creato can il liciolo egglyone. 10 / Por portability 1/ semget in cui si può specificare · Rey: la chiave univoca del semaforo. · nsems: - creazione: Il numero dei semaforo. · nsems: - cannessione: Il numero dei semaforo. · semflg: fiag aggivuttivi per permessi - semflg: frag aggivuttivi per permessi - liciolo reportamento	•	impostaine	il valore			Add 1 to semaphore
aspettare che il valore sia 0. I semafori vengono allocali in groppi di semafori, il cai nunero deve essere specificato in semget(); CREAZIONE SETAFORO Il semaforo viene creato can il secule (syr/sep. h. / Fee porebility // semaforo viene creato can il secules (syr/sep. h. / Fee porebility // semaforo viene creato can il secules (syr/sep. h. / Fee porebility // semaforo viene creato can il secules (syr/sep. h. / Fee porebility // semaforo viene creato can il secules (syr/sep. h. / Fee porebility // semaforo viene creato can il secules (syr/sep. h. / Fee porebility // semaforo viene creato can il secules (syr/sep. h. / Fee porebility // semaforo viene creato can il secules (syr/sep. h. / Fee porebility // semaforo viene con	•	aggiongere	on numero	al valore del s	sematoro.	Modes
aspettare che il valore sia 0. I semafori vengono allocali in groppi di semafori, il cai nunero deve essere specificato in semget(); CREAZIONE SETAFORO Il semaforo viene creato can il secule (syr/sep. h. / Fee porebility // semaforo viene creato can il secules (syr/sep. h. / Fee porebility // semaforo viene creato can il secules (syr/sep. h. / Fee porebility // semaforo viene creato can il secules (syr/sep. h. / Fee porebility // semaforo viene creato can il secules (syr/sep. h. / Fee porebility // semaforo viene creato can il secules (syr/sep. h. / Fee porebility // semaforo viene creato can il secules (syr/sep. h. / Fee porebility // semaforo viene creato can il secules (syr/sep. h. / Fee porebility // semaforo viene con	•	sottrare	on numero	al valore del :	sematoro.	Marca Rata - Laboratario di Salani Operatri, cana 8 - sano 73
T semafori vengono allocati in groppi di semafori, il cui nunero deve essere specificato in semget(); Creazione Setaforo) Il semaforo viene creato con il Historia egy/year. 10 / Por portality / semget in cui si può specificare. Rey: La chiave univoca del semaforo. • Rey: La chiave univoca del semaforo. • nsems: - creazione: Il numero dei semafori da creare per quell'insterne - connessione: Il edere essere alla dimensione dal set • semfy: flag aggiuntivi per permessi e di maclifica dal comportamento sono semporo della creazione - IPC creati: se non esiste, crea reviento - ICP Excc: se esiste già, errore Gestrone setto con semato • semid: l'ID univoca del groppo Total i semafori del groppo						
deve essere specificato in semgelli; Creazione Setaforo Il semaforo viene creato can il fincilule (syn/type) /* For portability // fincilule (syn/type)						
CRERZIONE SETAFORO Il semafioro viene creato con il fincione syntypes. 17 / For portability / Fincione syntypes. 18 / For portability / Fincione syntypes. 18 / For portability / Fincione syntypes. 18 / For portability / Fincione syntypes. 19 / For portability / F	1	sematori vi	engono alloca	ic in groppi di	sematori, i	cui numum
Il semaforo viene creato can il include departamento. Semaget in cui si può specificare Ney: la chique univora del semaforo. Semaforo. Semaforo. Ney: la chique univora del semaforo. Semaforo. Ney: la chique univora del semaforo. Semaforo. Semaforo. Ney: la chique univora del semaforo da crease per univora del semaforo. Semaforo. Semaforo univora del semaforo del semaforo da crease per univora del semaforo. Semaforo del dimensione del semaforo del se	de	ve essere	specificato ir	semget();		
Il semaforo viene creato can il include departamento. Semaget in cui si può specificare Ney: la chique univora del semaforo. Semaforo. Semaforo. Ney: la chique univora del semaforo. Semaforo. Ney: la chique univora del semaforo. Semaforo. Semaforo. Ney: la chique univora del semaforo da crease per univora del semaforo. Semaforo. Semaforo univora del semaforo del semaforo da crease per univora del semaforo. Semaforo del dimensione del semaforo del se	CRI	EAZIONE ?	Setaforo >			
Semget in coi si pub specificare Rey: la chiave univora del semaforo • nsems: - creazione: Il numero dei semafori da creare per quell'insiene - cannessione: Il # cleve essere C alla dimensione dal set • semfly: flag aggiuntivi per permessi e di maclifica del comportamento 1,500 000 1,500				#include <sys t<="" th=""><th></th><th>ility */</th></sys>		ility */
• Rey: La chiave univoca del semaforo • nsems: - creazione: Il numero dei semafori da creare per - cannessione: Il # deve essere \(\) alla dimensione dal set • semflg: flag aggivutivi per permessi \(\) \\ \) \(\) \\ \) \(\) \\ \(\) \(\) \(\) \(\) \(\) \(\) \(\) \(\) \(\) \(\) \(\) \\ \(\) \(\) \(\) \(\) \(\) \(\) \(\) \\\ \(\) \\\ \(\) \\\ \(\) \\\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\ \\				int semget (key		semflg);
• nsems: - creazione: Il numero dei semafori da creare per - carinessione: Il # cleve essere ⊆ alla dimensione dal set • semfly: flag aggiuntivi per permessi - semfly: flag aggiuntivi per permessi - similio devo sementi - sementi: se non esiste crea - similio devo sementi - semilio de caya/syma. b> /* hor portability */ - semid: l'ID onivoco del gruppo - seminori: - > - Il nom del semafori del gruppo - totti seminori del gruppo - cond: Specifica l'oparazione - cond: Specifica l'oparazione	sem	iget in co	i si può spe	cificate Return	rns semaphore set ident	
- creazione: Il numero dei semafori da creare per quell'insterne - connessione: Il # cleve essere \(\) alla dimensione dal set • semfla: flag aggiuntivi per permessi \(\) \$1,500 0000 \(\text{Semafori} \) \$1,500 000 \	•	Key: La	chique univoc	a del semaforc) .	
- creazione: Il numero dei semafori da creare per quell'insterne - connessione: Il # cleve essere \(\) alla dimensione dal set • semfla: flag aggiuntivi per permessi \(\) \$1,500 0000 \(\text{Semafori} \) \$1,500 000 \	•	nsems:				
- comnessione: Il # deve essere C alla dimensione del set • semfly: flag aggiuntivi per permessi • semfly: flag aggiuntivi per permessi • della comportamento • simble e di machifica del comportamento • sinso della creazione • IPC CREAT: se non esiste crea • ILINE e con semctl con param • ICCP EXCL: se esiste già, errore • semid: l'ID univa del grappo • semnom: • semnom: • semnom: • semnom: • semnom: • cmd: semafori del grappo • cmd: specifica l'operazione • cmd: specifica l'operazione			a Ti num oco	dei sama fori		05
- cannessione: It # clave essere \(\) alla dimensione del set\ • semfla: flag aggiuntivi per permessi \(\) Constant Octol value Permission bit 1,500 6000 Sequent D 1,500 6000		Cleater	ie . Il nomero) del semiliar à	ta creare	
• semflg: flag aggiuntivi per permessi • semflg: flag aggiuntivi per permessi • di modifica dal comportamento • signo eso segroupito • della creazione • le di modifica dal comportamento • signo eso segroupito • le di modifica dal comportamento • signo eso segroupito • le di modifica dal comportamento • signo eso segroupito • le di modifica dal comportamento • signo eso segroupito • signo eso so s			quell'insiem	۷ ا		
e di malifica del comportamento 5.1678 0200 Setgroup-10 0200 Sidis/s 5.1678 0400 Sidis/s 5.1678 040 Sid		- connessi	ione: Il # d	eve essere C	alla dimen	sione del set
e di malifica del comportamento 5.1678 0200 Setgroup-10 0200 Sidis/s 5.1678 0400 Sidis/s 5.1678 040 Sid	•	semfla: F	lag aggiuntivi	ber bermezzi		
della creazione S. JANSR 0000 Userwice		Δ -			_	
TPC CREAT: Se non esiste crea TPC CREAT: Se non esiste crea TCP-Excc: Se esiste già, esiste TCP-Excc: Se esiste già, esiste TCP-Excc: Se esiste già, esiste Tichiede; TCP-Excc: Se esiste già, esiste Tichiede (sys/types.h) /* For portability */ **include (sys/types.h) /* For portability */ **include (sys/sem.h) Int semoti, int semonum, int emd, /* union semun arg */); Returns nonnegative integer on success; returns -1 on error **Semnom: Totti semosori del groppo Totti semosori del groppo Totti semosori del groppo **Cmcl: Specifica l'oparazione **Cmcl: Specifica l'oparazione Totti semosori del groppo **Cmcl: Specifica l'oparazione			di handi fina		S_ISV	ID 02000 Set-group-ID IX 01000 Sticky
- TPC CREAT: Se non esiste crea TCP-CREAT: Se non esiste crea TCP-EXCL: Se esiste già, esiste TCP-EXCL: Se esiste già, esiste GESTIONE SETIAFORO #include <sys types.h=""> /* For portability */ #inclu</sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys>				del comportamen	S_IRU	D
TCP-EXCL: Se esiste già, enlose GESTIONE SETIAFORO #include (sys/types.h) /* For portability */ #include (sys/sem.h) Viene gestito con semct! con param int semcl!(int semid, int semnum, int cmd, /* union semun arg */); Returns nonnegative integer on success; returns -1 on error Semnom: Semnom: Totti i semafori del groppo Totti i semafori del groppo ""In union semu i singualo """In union semu i singualo """ """ """ """ """ """ """				del comportamen	S IRU S IRU S IXU S IXU S IXU S IXU S IXG S	D
GESTIONE SETIATORO #include <sys types.h=""> /* For portability */ #include <sys types.h=""> /* For portability */ #include <sys sem.h=""> Uiene gestito con sematt con param int semati, int semanum, int and, /* union semun arg */); Returns nonnegative integer on success; returns -1 on error • Semnom: ->O II nom del semaforo por gestie il singolo -=O Itali i semafori del groppo interiori servi integer con success; returns -1 on error • Cmd: Specifica l'operazione #include <sys types.h=""> /* For portability */ #include <sys types.h=""> /* #include <sys #include="" *="" *<="" <sys="" th="" types.ho=""><th></th><th>- IPC CR</th><th>ella creazione LEAT: Se non</th><th></th><th> SIRU SINU SIXU SIRG SIXG SIXG SIXG SIXG SIXG</th><th> Description Color Color</th></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys></sys>		- IPC CR	ella creazione LEAT: Se non		SIRU SINU SIXU SIRG SIXG SIXG SIXG SIXG SIXG	Description Color
Viene gestito con sematt con param int semati, int semanum, int and, /* union seman arg */); semid: L'TD univara del grappo semnom: semnom: ->o I nom del semaforo per cestire il singua		- IPC CR	ella creazione LEAT: Se non iedet	esiste, crea	SIRU SINU SIXU SIRG SIXG SIXG SIXG SIXG SIXG	Description Color
semid: l'TD univoco del gropo semnom: Semnom: TI nom del semaforo per gestire il singula		- IPC CR	ella creazione LEAT: Se non iedes CL: Se esist	esiste, crea	SIRU SINU SINU SIXU SIXU SIXU SIXU SIXG	10
• semid: L'TD univore del gruppo semnum: ->0 - Il num del semaforo per gestire il singdo -=0 - Tutti i semafori del gruppo union semun (// value for SETVAL int val) butter for IPC STAT, IPC SET struct semid de' butt; // array for SETALL, STATAL unsigned short: array; // Linux specific pat if defined _linux) // butter for IPC STALL butter for IPC STALL unsigned short: array; // Linux specific pat if defined _linux) // butter for IPC NOR		TCP-EX	ella creazione LEAT: Se non Medet CL: Se esist	esiste crea e già, essor #incl	S RU S IWU S IWU S IWU S IWU S IWU S IWG S I	10
->> TI nom del semaforo per gestire il singolo >> Totti i semafori del groppo int val: // value for SETVAL int val. // buffer for IFC STAT, IFC SET struct semid del but; // array for GETALL unsigned short array; // Linux specific part fit defined linux) // buffer for IFC STAT, IFC SET struct semid del but; // array for GETALL unsigned short array; // Linux specific part fit defined linux) // buffer for IFC INFO		TCP-EX	ella creazione LEAT: Se non Medet CL: Se esist	esiste crea e già, essor #incl	SIRU SINU Mude <sys types.h=""> /* Foude <sys sem.h=""> emctl (int semid, int semid, int semid, /*</sys></sys>	Description of the control of the co
->> TI nom del semaforo per gestire il singolo >> Totti i semafori del groppo int val: // value for SETVAL int val. // buffer for IFC STAT, IFC SET struct semid del but; // array for GETALL unsigned short array; // Linux specific part fit defined linux) // buffer for IFC STAT, IFC SET struct semid del but; // array for GETALL unsigned short array; // Linux specific part fit defined linux) // buffer for IFC INFO	Vier	- IPC CR CIED - ICP EX STIONE SI	ella creazione CEAT: Se non iedej CC: Se esiste ETIAFORO > con semct	esiste crea esiste crea #incl #incl #incl int s	SIRU SINU Mude <sys types.h=""> /* Foude <sys sem.h=""> emctl (int semid, int semid, int semid, /*</sys></sys>	Description of the control of the co
Tutti Esemafori del gruppo "Int val: int val: int val: int val: int val: // buffer for IPC. STAT. IPC. SET struct senid de l' array for GETALL, SETALL unsigned short array; // Linux specific part fit defined linux // buffer for IPC. INDO	Vier	- IPC CR - ICP EX STIONE SI ne gestito semid: l'	ella creazione CEAT: Se non iedej CC: Se esiste ETIAFORO > con semct	esiste crea esiste crea #incl #incl #incl int s	SIRU SINU Mude <sys types.h=""> /* Foude <sys sem.h=""> emctl (int semid, int semid, int semid, /*</sys></sys>	Description of the control of the co
* struct semid ds* buf; // array for GETALL, SETALL unsigned short* array; // Linux specific part #if defined linux_ // buffer for IFC_INFO	Vier	- IPC CR - IPC CR (Ic) - ICP EX STIONE SI ne gestito semid: l' semnom:	ella creazione LEAT: Se non iele) CC: Se esist ETIAFORO Con semct	esiste, crea e già, eclose #incl #incl int s del gruppo	SIRU SINU Mude <sys types.h=""> /* Fo ude <sys sem.h=""> emctl(int semid, int sem int cmd, /* Returns nonneg</sys></sys>	Set-group-ID Sticky SR 0400 User-read User-write SR 0100 User-execute SR 0100 User-execute SR 0100 User-execute SR 0100 Group-read SR 010 Group-write SR 010 Group-write SR 010 Group-execute SR 010 Other-write Other-write Other-write Other-execute Th 01 Other-execute Th 02 other-write Other-write Th 01 other-write Other-write Other-write Other-write Other-write
• cmcl: Specifica l'operazione unsigned short* array; // Linux specific part #if defined (_linux_) // buffer for IPC_INFO	Vier	- IPC CR - ICP EX SI (ONE SI ne gestito semid: l' semnom: ->0 -> I	ella creazione PEAT: Se non Piede) PCL: Se esist PETIAFORO PCON Semct TD UNIVOCO TD UNIVOCO	esiste, crea e già, estose #incl #incl int s del gruppo	SIRU SINU Mude <sys sem.h=""> /* Fo ude <sys sem.h=""> Returns nonneg</sys></sys>	Set-group-ID Sticky SR 0400 User-read User-write SR 0100 User-execute SR 0100 User-execute SR 0100 User-execute SR 0100 Group-read SR 010 Group-write SR 010 Group-write SR 010 Group-execute SR 010 Other-write Other-write Other-write Other-execute Th 01 Other-execute Th 02 other-write Other-write Th 01 other-write Other-write Other-write Other-write Other-write
struct seminfo* buf;	Vier	- IPC CR - ICP EX SI (ONE SI ne gestito semid: l' semnom: ->0 -> I	ella creazione PEAT: Se non Piede) PCL: Se esist PETIAFORO PCON Semct TD UNIVOCO TD UNIVOCO	esiste, crea e già, eclose #incl #incl int s del gruppo semaforo per ges del gruppo	SIRU SINU SINU SINU SIRU SINU SIRU SINU SINU	Set-group-ID Sticky SR 0400 User-read User-write SR 0100 User-execute SR 0100 User-execute SR 0100 User-execute SR 0100 Group-read SR 010 Group-write SR 010 Group-write SR 010 Group-execute SR 010 Other-write Other-write Other-write Other-execute Th 01 Other-execute Th 02 other-write Other-write Th 01 other-write Other-write Other-write Other-write Other-write
• arg: Union di campi di supporto.	Vier	- IPC CR - ICP - EX SI (ONE SI ne gestito semid: l' semnom: ->0 -> I	ella creazione CEAT: Se non iede) CC: Se esist ETIAFORO Con semct TD univoca I num del semafori	esiste crea egià, eclose #incl #incl int s del gruppo semaforo per ges del gruppo	SIRU SINU SINU	Set-group-ID Sticky SR 0400 User-read User-write SR 0100 User-execute SR 0100 User-execute SR 0100 User-execute SR 0100 Group-read SR 010 Group-write SR 010 Group-write SR 010 Group-execute SR 010 Other-write Other-write Other-write Other-execute Th 01 Other-execute Th 02 other-write Other-write Th 01 other-write Other-write Other-write Other-write Other-write

```
di cma possono essere
Le operazioni
  · IPC_RMID:
                         Rimuoue il set di semafori con semid
                         copia la struttura coi dati attuali in arg. buf
   · IPC_STATT
  · IPC SET
                         Aggiorna la struttura dei dati in asy. but
   · GETUAL
                         Dato semnom, restituisce il valore dei semaforo.
                         Dato semnom, imposto il valore da aug. val
   · SETUAL
                         Restituisce i valori di tutti i semafori in argiarrag
  · GETALL
                         Imposta i valori di totti i semafori in arg. amag
  · SETALL
                         Ritorna il PID dell'ultimo processo che ha eseguito
  · GETPLD
                         semop() altrimenti 0
    Csement1
                         Ritorna il # dei processi in attesa di increnento
   · GETCUT
     (sem zont)
                         Riborna il # dei processi che attendano lo O.
   · CETZCUI
                                                                    IPC_SET.

sem_otime. Questo membro è settato a 0 alla creazione del set di semafori, ed aggiornato all'ora corrente ad agni semopi) che va a buon fine, a quando il valore del semaforo è madificato i nesguito a uni-perazione SEM_UNDO.

sem_ctime. questo membro è impostato all'ara corrente al momento della creazione del semaforo, è na aguito a agni aperazione IPC_SET, SETALL, a SETNALI.

sem_nsems. membro inizializzato al momento della creazione del set di semafori, in contien il numero di semafori nel set.
                                                       #include <sys/types.h>/* For portability */
OPERAZIONI SUI SEMAFORI >
                                                      #include <sys/sem.h>
                                                      int semop(int semid, struct sembuf *sops, unsigned int nsops);
            una o pus operazioni sui
Esegue
                                                                        Returns 0 on success, or -1 on error
semafori definiti da semidi
  · sops: Army contenente le
                                                  operazioni da eseguire
  · nsops: Dimensione array sops.
                                                     struct sembuf {
                                                       unsigned short sem_num; // numero semaforo
                                                       short sem_op;
                                                                     // operazione da eseguire
  STRUTTURA SERBOF
                                                       short sem flg;
                                                                    // flags operazione
                                                                     // (IPC NOWAIT and SEM UNDO)
  Contiene
     · sem_nom: II # del semaforo
     sem-op;
                       se >0 aggionge, se =0 aspetta fin quando è 0
                             <0 decrementa se il nuovo val
                       altrineuti asperta
     · sem fly: flag aggiontivi
         - IPC NOWAT: Al posto di blacate
                                                                 da errore GAGAW
Operazioni moltiple vengono fatte nei semafori in maniera
```

