Capitolo 3: Processi

Capitolo 3: Processi

- Concetto di processo
- Scheduling dei processi
- Operazioni sui processi
- Cooperazione tra processi
- Comunicazione tra processi
- Comunicazione nei sistemi client-server

Processo in memoria

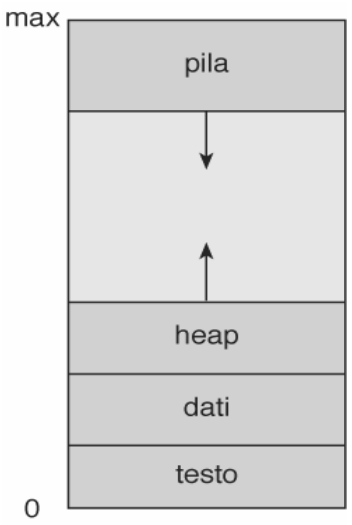
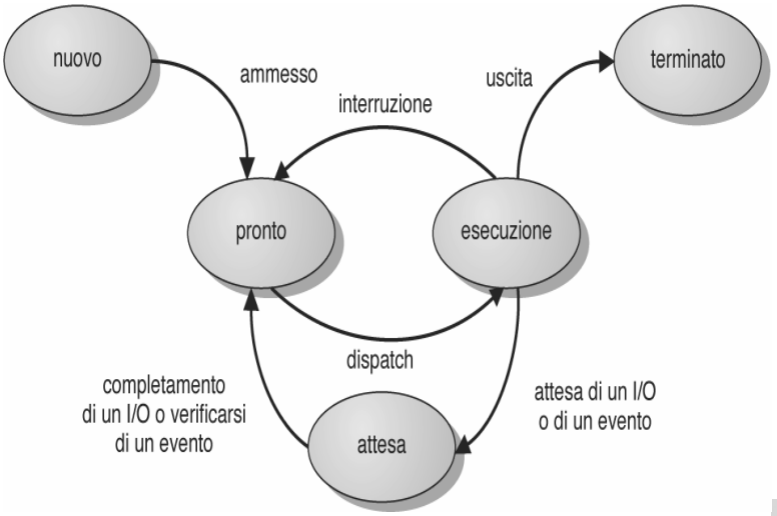
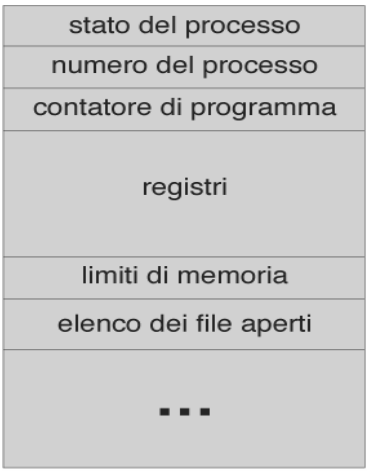


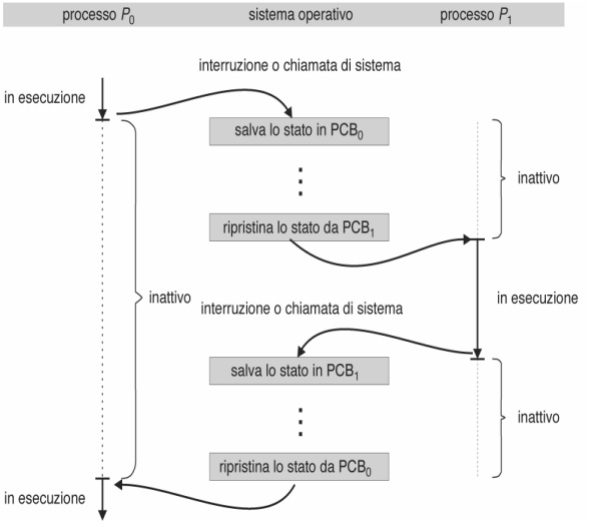
Diagramma di transizione degli stati di un processo



Blocco di controllo dei processi (PCB)



La CPU può essere commutata tra i processi



Coda dei processi pronti e diverse code dei dispositivi di I/O

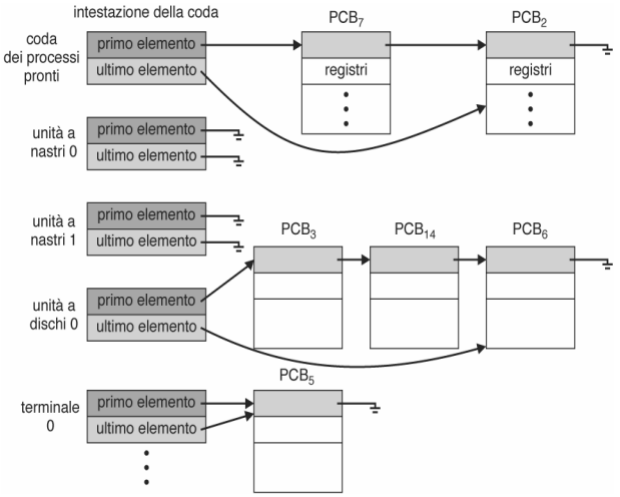
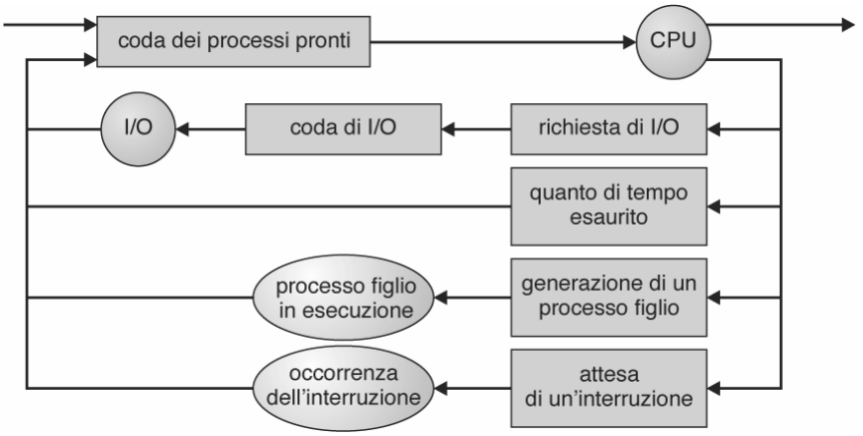
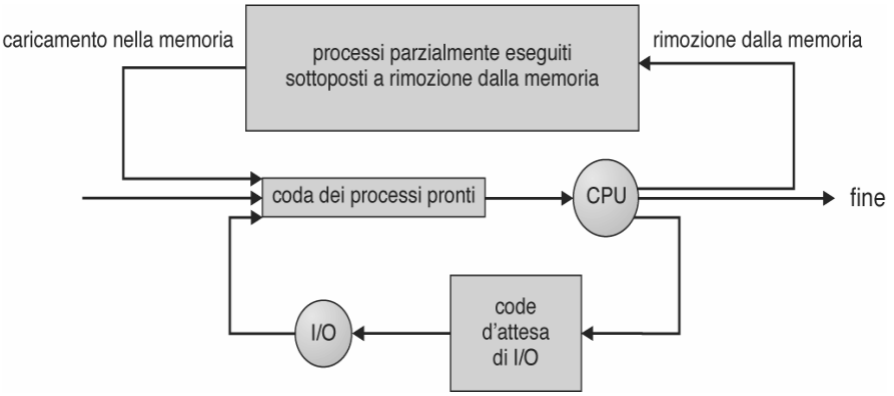


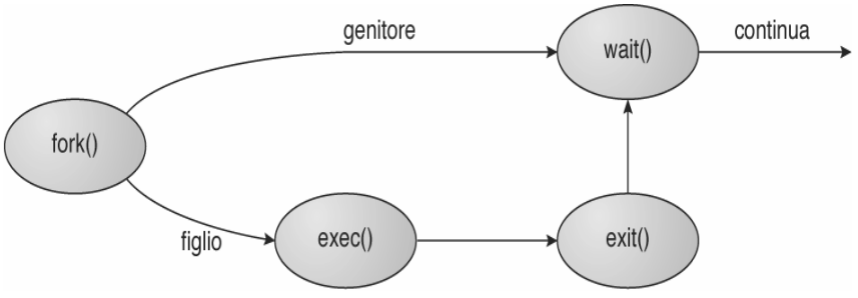
Diagramma di accodamento per lo scheduling dei processi



Aggiunta di scheduling a medio termine al diagramma di accodamento



Generazione di un processo



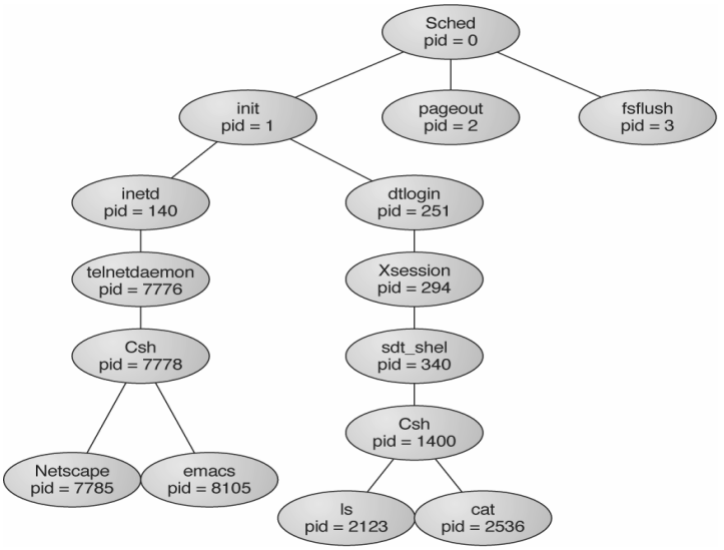
Programma C che genera un nuovo processo

```
int main()
{
pid_t pid;

    /* genera un nuovo processo */
    pid = fork();

    if (pid < 0) { /* errore */
fprintf(stderr, "generazione del nuovo processo fallita");
exit(-1);
    }
    else if (pid == 0) { /* processo figlio */
      execlp("/bin/ls", "ls",NULL);
    }
    else {/* processo genitore */
      /* il genitore attende il completamento del figlio */
wait(NULL);
printf("il processo figlio ha terminato");
exit(0);
    }
 }
```


Esempio di albero dei processi di Solaris



Buffer limitato – Condivisione di memoria tra processi

- Dati condivisi

```
#define DIM_BUFFER 10

          typedef struct {
. . .
          } elemento;

          elemento buffer[DIM_BUFFER]
          int inserisci = 0;
                  int preleva = 0;
```

- La soluzione è corretta, ma può utilizzare solo DIM_BUFFER_-1 elementi

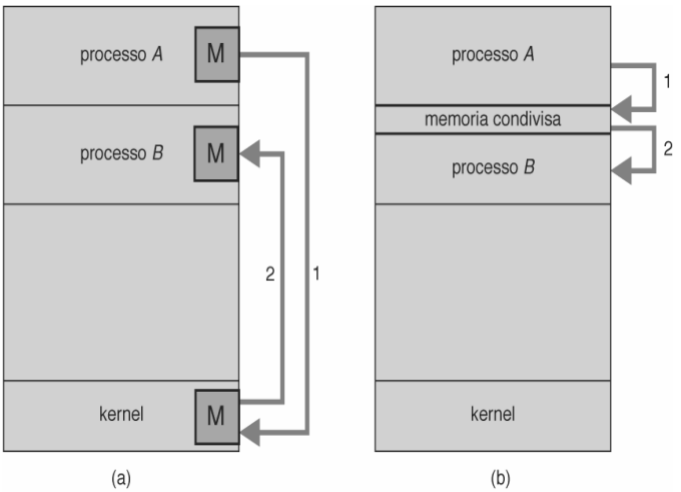
Buffer limitato – Metodo Insert()

```
while (true) {
   /* produce un elemento in appena_Prodotto */
   while (((inserisci + 1) % DIM_BUFFER) == preleva)
     ; /* non fa niente */
   buffer[inserisci] = appena_Prodotto;
   inserisci = (inserisci + 1) % DIM_BUFFER;
}
```

Buffer limitato – Metodo Remove()

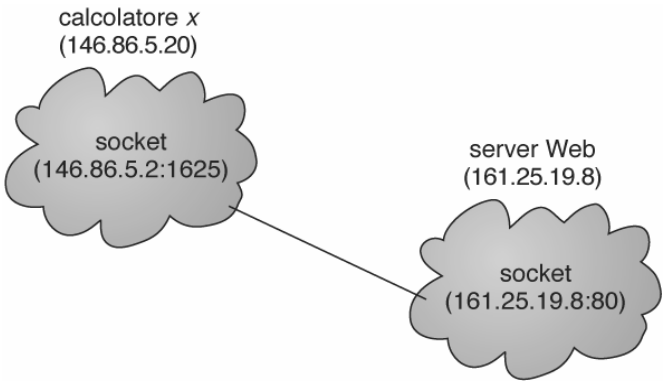
```
while (true) {
   while (inserisci == preleva)
        ; /* non fa niente */
    da_Consumare = buffer[preleva];
    preleva = (preleva + 1) % DIM_BUFFER;
    /* consuma l'elemento in da_Consumare */
}
```


Modelli di comunicazione tra processi

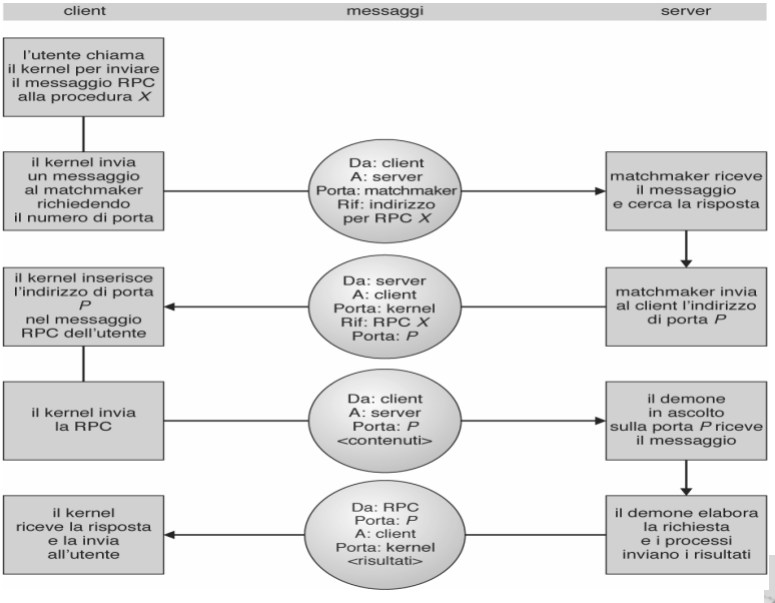


Modelli di comunicazione tra processi basati su (a) scambio di messaggi, e (b) condivisione della memoria.

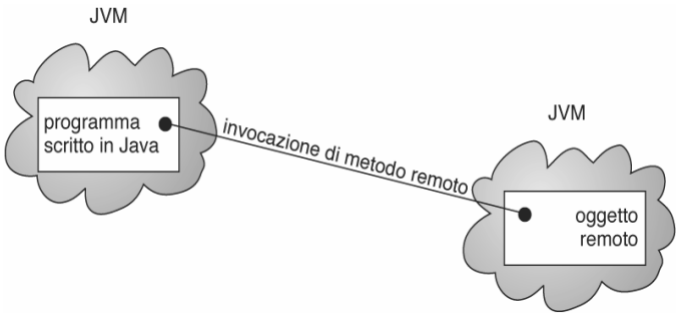
Comunicazione tramite socket



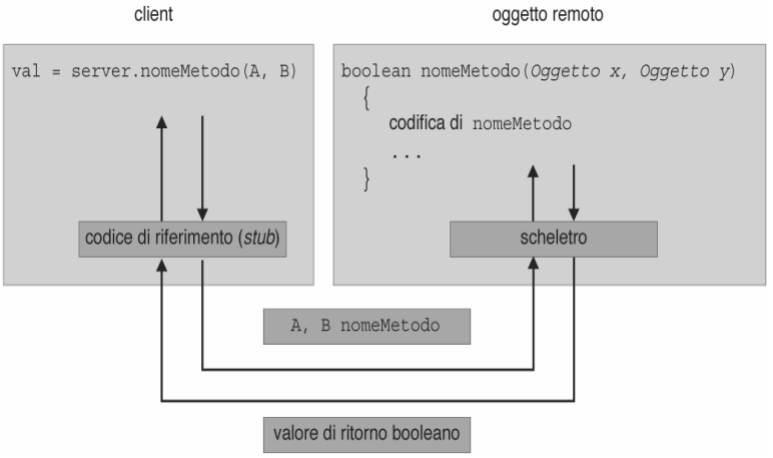
Esecuzione di una RPC



Invocazione di metodi remoti



Strutturazione dei parametri



Fine del Capitolo 3