

### Cenni storici – 1

- **Anni '50:** i S/O hanno origine con i primi elaboratori a programma memorizzato
  - Modalità di esecuzione *batch* (a lotti) gestita da un operatore umano
    - **Tutto** l'elaboratore a disposizione di **un solo** programma per tutto il tempo della sua esecuzione
    - Immissione di un programma mediante interruttori binari frontali o schede perforate
    - Emissione dei risultati mediante lampadine, testo stampato, schede perforate

Introduzione

Sistemi Operativi - T. Vardanega

Pagina 1/19

### Cenni storici – 2

- A partire dagli **anni '60**
  - Nuovi compilatori, nuovi linguaggi di programmazione, nuovi strumenti di sviluppo
- Ancora gestione a lotti
  - Immissione ed emissione di programmi e dati ancora molto laboriose (= molto costose e molto lente)
    - Un programma può essere sia ingresso che uscita di una esecuzione
  - L'operatore umano è ancora necessario per eseguire le operazioni di ingresso / uscita

Introduzione

Sistemi Operativi - T. Vardanega

Pagina 2/19

### Cenni storici – 3

- L'esecuzione di più **lavori** in modalità a lotti può essere facilmente gestita da un S/O **residente**
  - Più caricamenti seguiti da una fase ininterrotta di lavoro e dal ritrovamento dei rispettivi risultati
  - Ordine di esecuzione **predeterminato**
    - Quello di caricamento
    - Secondo il livello di privilegio del richiedente
- Operazioni di I/O fino a 1.000 volte più lente dell'elaborazione
  - Esecuzione *off-line*
    - Senza richiedere tempo di elaboratore
  - Sovrapposizione tra I/O ed elaborazione

Introduzione

Sistemi Operativi - T. Vardanega

Pagina 3/19

### Cenni storici – 4

- Sovrapposizione sempre più conveniente al crescere delle prestazioni dei dispositivi di I/O
  - Tecnica detta di *spooling* (*Simultaneous Peripheral Operation On-Line*)
  - Si effettua *spooling* quando l'emissione o la ricezione di dati avviene in parallelo all'esecuzione di altri lavori
    - Esempi
      - Invio di una richiesta di stampa
      - Caricamento di un programma
      - Invio di un messaggio di posta elettronica
  - **Senza interrompere** il lavoro in corso

Introduzione

Sistemi Operativi - T. Vardanega

Pagina 4/19

### Cenni storici – 5

- **Multiprogrammazione**
  - Desiderabile poter eseguire diversi lavori simultaneamente
    - In ambito mono-processore il parallelismo è solo simulato
  - Occorre controllare l'assegnazione dell'accesso alle risorse della macchina
    - Esempio: per **quanti di tempo** in modalità *time-sharing* sotto il controllo del S/O

Introduzione

Sistemi Operativi - T. Vardanega

Pagina 5/19

### Definizione di S/O

- Un insieme di utilità progettate per
  - Offrire all'utente un'astrazione più semplice e potente della macchina *assembler*
    - Concetto di "**macchina virtuale**"
    - Più semplice da usare (p.es., senza bisogno di conoscenze di microprogrammazione ©)
    - Più potente (p.es., usando la memoria secondaria per realizzare una più ampia memoria principale virtuale)
  - Gestire in maniera **ottimale** le risorse fisiche e logiche dell'elaboratore
    - Ottimalità è la minimizzazione dei tempi di attesa e la massimizzazione dei lavori svolti per unità di tempo

Introduzione

Sistemi Operativi - T. Vardanega

Pagina 6/19

## Nozione di processo

- Un processo è un **programma in esecuzione** e corrisponde a
  - L'insieme **ordinato** di **stati** assunti dal programma nel corso dell'esecuzione (sulla sua macchina virtuale)
    - Processo come "automa a stati"
  - L'insieme **ordinato** delle **azioni** effettuate dal programma nel corso dell'esecuzione (sulla sua macchina virtuale)
    - Processo come "attore" (operatore di azioni)

Introduzione

Sistemi Operativi - T. Vardanega

Pagina 7/19

## Realizzazione di processo

- Spazio di **indirizzamento logico**
  - La memoria della **macchina virtuale** che il processo può leggere e scrivere (*core*)
    - Memoria virtuale organizzata a pagine e/o segmenti
  - Programma eseguibile
    - Organizzazione dell'informazione in forma di *file*
  - Dati del programma
    - Organizzazione dell'informazione in forma di *file*
  - Aree di lavoro
  - Definizione del **contesto di esecuzione** ed area di salvataggio

Introduzione

Sistemi Operativi - T. Vardanega

Pagina 8/19

## Caratteristiche di processi – 1

- In un sistema coesistono processi utente e di S/O
  - Possono cooperare tra loro ma hanno privilegi diversi
- I processi avanzano **concorrentemente**
  - Il S/O assegna loro le risorse necessarie secondo diverse politiche di ordinamento
    - A divisione di tempo
    - A livello di priorità (urgenza)
- I processi possono dover comunicare e sincronizzarsi tra loro
  - Il S/O deve fornire i meccanismi e i servizi necessari

Introduzione

Sistemi Operativi - T. Vardanega

Pagina 9/19

## Caratteristiche di processi – 2

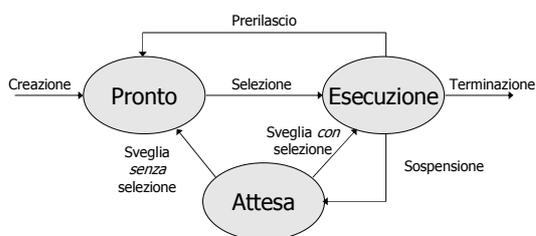
- Un processo può creare processi "figli"
  - Esempio
    - Un processo interprete di comandi (*shell*) lancia un processo figlio per eseguire un comando di utente
- I processi vengono
  - **Creati** per eseguire un lavoro
  - **Sospesi** per consentire l'esecuzione di altri processi
  - **Terminati** al compimento del lavoro assegnato
    - Un processo figlio che sopravvive alla terminazione del processo padre è detto "orfano" e può essere molto dannoso

Introduzione

Sistemi Operativi - T. Vardanega

Pagina 10/19

## Stati di avanzamento di processo



Introduzione

Sistemi Operativi - T. Vardanega

Pagina 11/19

## Gestore dei processi – 1

- Costituisce il cuore o nucleo del S/O (*kernel*)
  - Gestisce ed assicura l'avanzamento dei processi
    - Stato di avanzamento
      - In esecuzione, pronto per l'esecuzione, sospeso in attesa di un evento (una comunicazione, la disponibilità di una risorsa, ...)
    - La scelta del processo da eseguire ad un dato istante si chiama ordinamento (*scheduling*)
    - Il gestore decide il cambio di stato dei processi
      - Per divisione di tempo
      - Per trattamento di eventi (p.es., risorsa libera / occupata)

Introduzione

Sistemi Operativi - T. Vardanega

Pagina 12/19

## Gestore dei processi – 2

- Compiti del nucleo di S/O
  - Gestire l'avanzamento dei processi
    - Registrando ogni transizione nel loro stato di attivazione
  - Gestire le interruzioni esterne (all'esecuzione corrente) causate da
    - Eventi di I/O
    - Situazioni anomale rilevate da altri processi o componenti del S/O
  - Consentire ai processi di accedere a risorse di sistema e di attendere eventi

Introduzione

Sistemi Operativi - T. Vardanega

Pagina 13/19

## Gestore dei processi – 3

- La politica di ordinamento deve essere equa (*fair* → *fairness*)
  - Processi pronti per eseguire devono avere l'opportunità di farlo
  - Processi in attesa di risorse devono avere l'opportunità di accederle
- I meccanismi e servizi di comunicazione e sincronizzazione devono essere efficaci
  - Il dato (o segnale) inviato da un processo mittente deve raggiungere il destinatario in un tempo breve e in modo sicuro

Introduzione

Sistemi Operativi - T. Vardanega

Pagina 14/19

## Definizione di risorsa

- **Risorsa** è qualsiasi elemento fisico (*hardware*) o logico (realizzato a *software*) necessario alla creazione, esecuzione e avanzamento di processi
- Le risorse possono essere
  - Durevoli (p.es., CPU)
  - Consumabili (p.es., memoria fisica)
  - Ad accesso divisibile o indivisibile
    - Divisibile se tollera alternanza con accessi di altri processi
    - Indivisibili se *non* tollera alternanza durante l'uso
  - Ad accesso individuale o molteplice
    - Molteplicità fisica o logica (virtualizzata)

Introduzione

Sistemi Operativi - T. Vardanega

Pagina 15/19

## Risorsa CPU

- Risorsa indispensabile per l'avanzamento di tutti i processi
- A livello fisico (*hardware*) corrisponde alla CPU
- A livello logico (sotto gestione *software*) può essere vista come una **macchina virtuale**
  - Offerta dal S/O alle sue applicazioni

Introduzione

Sistemi Operativi - T. Vardanega

Pagina 16/19

## Risorsa memoria

- Scrittura: risorsa ad accesso individuale
- Lettura: risorsa ad accesso multiplo
- La gestione *software* la **virtualizza** (usandola insieme alla memoria secondaria) attribuendone l'accesso ai vari processi secondo particolari politiche
- Se virtualizzata, diventa riutilizzabile e preriutilizzabile
  - Altrimenti consumabile e indivisibile
- Gestione velocizzata con l'utilizzo di supporto *hardware*

Introduzione

Sistemi Operativi - T. Vardanega

Pagina 17/19

## Risorsa I/O

- Risorse generalmente riutilizzabili, non preriutilizzabili, ad accesso individuale
- La gestione *software* ne facilita l'impiego nascondendone le caratteristiche *hardware* e uniformandone il trattamento
- L'accesso fisico ha bisogno di utilizzare programmi proprietari e specifici
  - **BIOS**

Introduzione

Sistemi Operativi - T. Vardanega

Pagina 18/19

## Caricamento del S/O

Il S/O può risiedere

- Permanentemente in ROM
  - Soluzione tipica di sistemi di controllo industriale e di **sistemi dedicati**
- In memoria secondaria per essere caricato (tutto o in parte) in RAM all'attivazione di sistema (*bootstrap*)
  - Adatto a sistemi di elevata complessità oppure predisposti al controllo (alternativo) da parte di più istanze di S/O
  - In ROM risiede solo il caricatore di sistema (*bootstrap loader*)