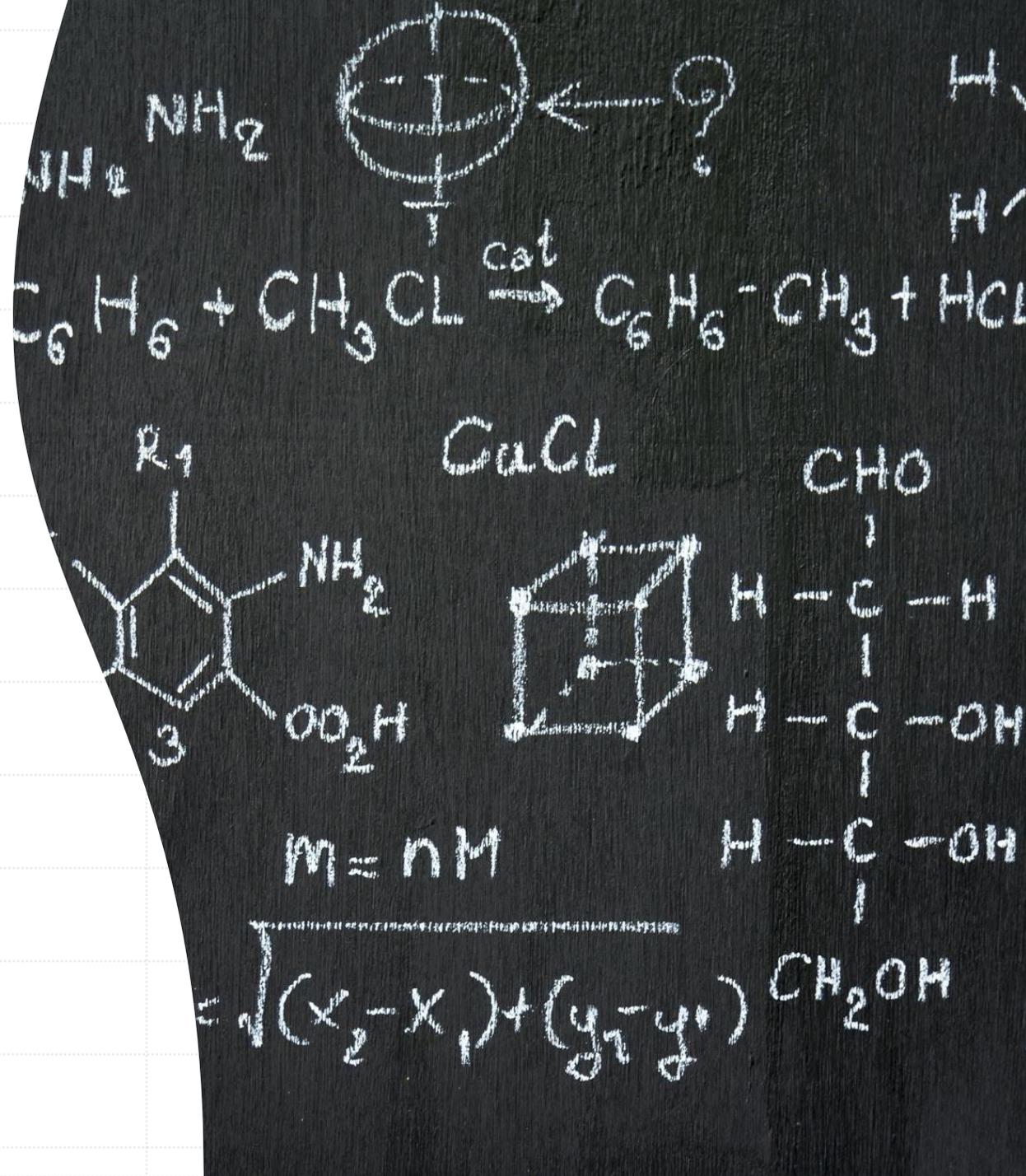


Storia della programmazione



Le origini: Antiche forme di programmazione

- Nel corso della storia, l'umanità ha sviluppato diverse forme di programmazione molto prima dell'avvento dei computer moderni
- Queste prime forme di programmazione coinvolgevano l'utilizzo di dispositivi meccanici per eseguire calcoli e automatizzare compiti specifici
- Uno dei dispositivi più antichi utilizzati per la programmazione era l'abaco, un semplice strumento di calcolo composto da perline o pietre mobili disposte su fili o aste



Ada Lovelace e il suo contributo pionieristico alla programmazione



Nel XIX secolo, Ada lavorò con Charles Babbage, un matematico e inventore britannico, sul suo progetto di macchina analitica, un precursore del computer moderno



L'opera di Ada Lovelace ha avuto un impatto significativo sullo sviluppo della programmazione moderna

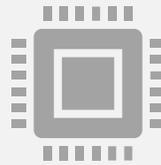


Il suo lavoro pionieristico ha contribuito a gettare le basi per l'idea di utilizzare le macchine per svolgere compiti diversi dalla semplice elaborazione numerica, aprendo la strada alla programmazione come la conosciamo oggi

I computer primitivi: ENIAC, UNIVAC e l'avvento dell'elettronica digitale



Uno dei primi computer di grande rilevanza fu l'ENIAC , sviluppato negli Stati Uniti durante la Seconda Guerra Mondiale



L'ENIAC fu il primo computer completamente elettronico e fu utilizzato per calcoli complessi e per scopi militari

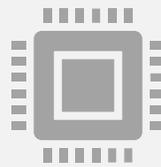


L'UNIVAC introduceva l'uso di memorie magnetiche, consentendo una maggiore capacità di archiviazione dei dati

Linguaggi di programmazione dei primi computer: Assembly, linguaggi a basso livello



I primi computer venivano programmati utilizzando linguaggi di basso livello, come l'Assembly



L'Assembly era un linguaggio di programmazione molto vicino al linguaggio macchina, basato su istruzioni specifiche per l'hardware del computer



I programmatori dovevano scrivere il codice Assembly direttamente in base alle specifiche del computer e delle sue istruzioni

L'era dei computer mainframe

- Questi enormi sistemi informatici, spesso custoditi all'interno di grandi sale dedicate, erano utilizzati da aziende, istituzioni governative e organizzazioni scientifiche per elaborare grandi quantità di dati e risolvere problemi complessi
- I computer mainframe erano dotati di potenti processori, capacità di archiviazione notevole e supportavano un gran numero di utenti simultanei
- Essi fornivano servizi di calcolo, archiviazione e condivisione dei dati su larga scala, rappresentando un elemento chiave nell'evoluzione della tecnologia informatica



Il linguaggio di programmazione FORTRAN

- Nel contesto dell'era dei computer mainframe, il linguaggio di programmazione FORTRAN ha svolto un ruolo fondamentale
- FORTRAN semplificava la scrittura di programmi complessi, offrendo una sintassi più vicina all'inglese e introducendo funzionalità di calcolo avanzate
- Questo linguaggio ha consentito ai ricercatori di scrivere programmi più facilmente comprensibili e di sfruttare al meglio le potenzialità dei computer mainframe per risolvere problemi scientifici e di ingegneria

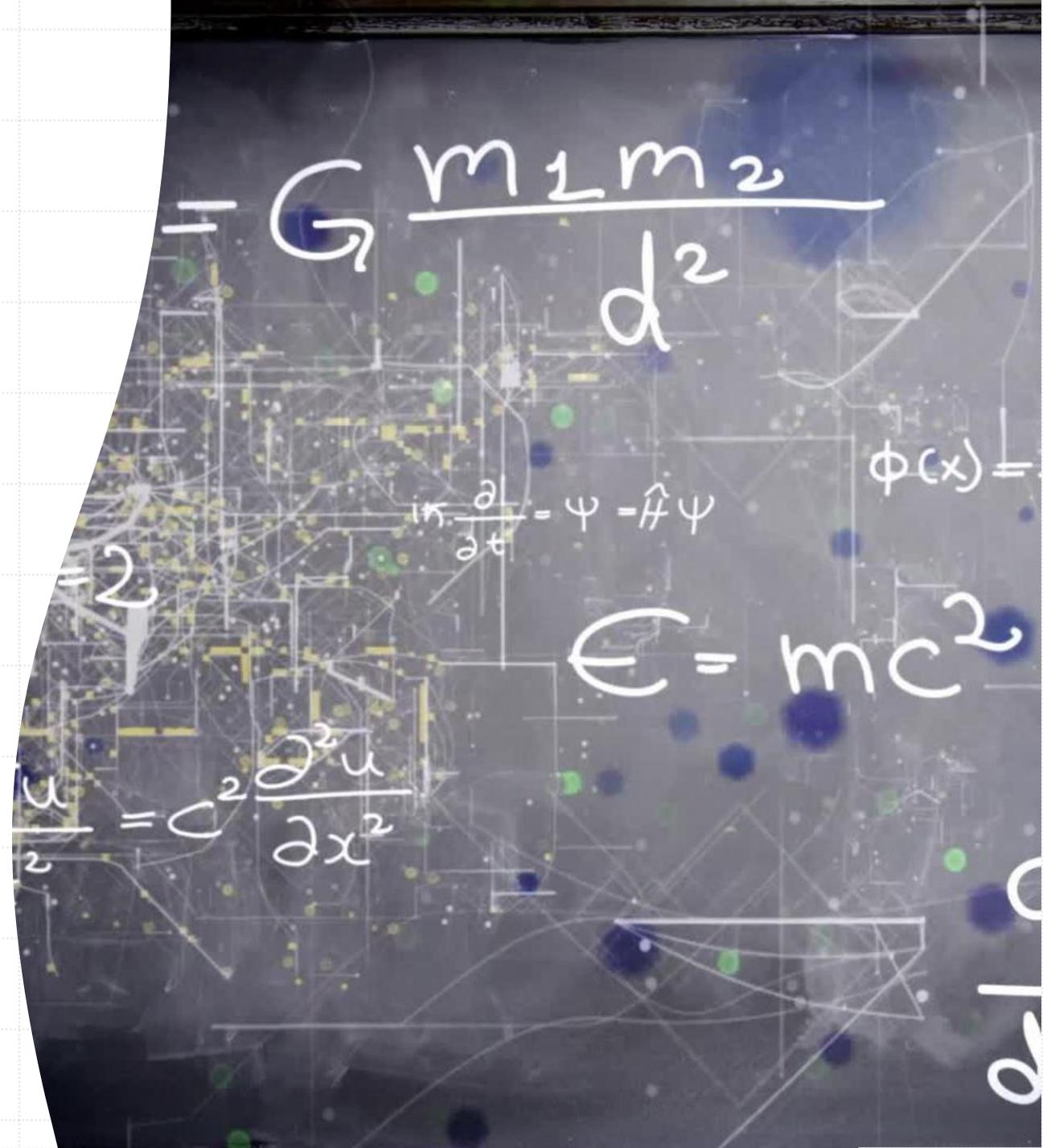


L'evoluzione dei linguaggi di programmazione: COBOL, ALGOL, Lisp

- COBOL : COBOL è stato sviluppato negli anni '60 ed è stato progettato per l'elaborazione di dati aziendali e finanziari
- ALGOL : ALGOL è stato sviluppato negli anni '60 ed è stato uno dei primi linguaggi di programmazione ad alto livello a introdurre concetti fondamentali della programmazione strutturata
- Lisp : Lisp è stato sviluppato negli anni '50 ed è noto per essere il secondo linguaggio di programmazione ad alto livello mai creato



L'evoluzione dei linguaggi di programmazione: COBOL, ALGOL, Lisp



Rivoluzione informatica: l'avvento dei personal computer

- Negli anni '70, l'avvento dei personal computer ha segnato una vera e propria rivoluzione informatica
- L'era dei personal computer ha aperto nuove possibilità e ha contribuito a diffondere l'informatica a livello globale
- L'importanza dei personal computer risiede nella democratizzazione dell'informatica, offrendo alle persone la possibilità di eseguire calcoli, creare documenti e persino scrivere programmi direttamente dal proprio computer domestico



Il linguaggio di programmazione BASIC

- Con l'avvento dei personal computer, è emersa la necessità di un linguaggio di programmazione accessibile e intuitivo
- Il linguaggio BASIC è stato sviluppato per soddisfare questa esigenza
- Questo linguaggio ha giocato un ruolo significativo nell'educazione informatica e nella diffusione della programmazione tra il grande pubblico



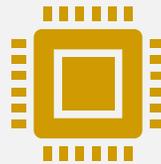
Linguaggio di programmazione C



Negli anni '70, il linguaggio di programmazione C è stato introdotto da Dennis Ritchie presso i laboratori Bell



Il linguaggio C ha avuto un impatto significativo nell'industria informatica ed è diventato uno dei linguaggi di programmazione più popolari di tutti i tempi



È stato utilizzato per lo sviluppo di sistemi operativi come Unix e ha influenzato il design di molti altri linguaggi successivi, come C++, Java e Python

Linguaggio di programmazione Pascal

- Nel 1970, il linguaggio di programmazione Pascal è stato sviluppato da Niklaus Wirth
- Pascal era un linguaggio di programmazione strutturato progettato per favorire la chiarezza e la comprensibilità del codice
- Pascal ha introdotto molti concetti importanti della programmazione strutturata, come le procedure e le funzioni, che hanno contribuito a migliorare l'organizzazione e la manutenibilità del codice



L'evoluzione della programmazione orientata agli oggetti

- La programmazione orientata agli oggetti è emersa negli anni '60 e '70 come approccio innovativo alla progettazione del software
- Si basa sul concetto di "oggetti" che racchiudono dati e comportamenti correlati
- La POO offre un modo modulare e organizzato per creare programmi complessi, consentendo una maggiore riusabilità del codice e una migliore gestione delle interazioni tra gli oggetti
- L'introduzione di linguaggi di programmazione specifici per la POO ha contribuito alla sua adozione diffusa
- Java ha trovato ampia applicazione nello sviluppo di applicazioni web, software di sistema e dispositivi mobili
- Negli ultimi anni, Python ha guadagnato popolarità come linguaggio di programmazione orientato agli oggetti

```
mirror_mod = modifier_ob.  
    # Add mirror object to mirror_mod  
    mirror_mod.mirror_object =  
    operation == "MIRROR_X":  
    mirror_mod.use_x = True  
    mirror_mod.use_y = False  
    mirror_mod.use_z = False  
    operation == "MIRROR_Y":  
    mirror_mod.use_x = False  
    mirror_mod.use_y = True  
    mirror_mod.use_z = False  
    operation == "MIRROR_Z":  
    mirror_mod.use_x = False  
    mirror_mod.use_y = False  
    mirror_mod.use_z = True
```

```
#selection at the end -add  
mirror_ob.select= 1  
modifier_ob.select=1  
context.scene.objects.active  
    ("Selected" + str(modifier_ob.name))  
    mirror_ob.select = 0  
    bpy.context.selected_objects  
    data.objects[one.name].select  
print("please select exactly
```

--- OPERATOR CLASSES ---

```
types.Operator):  
    # Add X mirror to the selected  
    object.mirror_mirror_x"  
    mirror X"
```

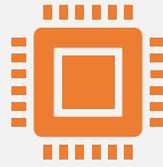
```
context):  
    context.active_object is not
```

Programmazione funzionale: concetti e linguaggi come Lisp, Haskell e Scala

- Nel campo della programmazione, sono emersi diversi paradigmi che offrono approcci innovativi per lo sviluppo del software
- Questa lezione esplorerà tre paradigmi di programmazione moderni: la programmazione funzionale, la programmazione logica e i paradigmi emergenti come la programmazione reattiva e la programmazione parallela
- La programmazione funzionale è un paradigma che si basa sul concetto di funzioni pure e immutabilità dei dati



Programmazione logica: Prolog e il ragionamento automatizzato



La programmazione logica è un paradigma che si basa sulla logica formale e sul ragionamento automatizzato



Questo paradigma si presta particolarmente bene alla risoluzione di problemi basati su regole e relazioni



Prolog ha trovato applicazioni in diversi campi, come l'intelligenza artificiale, la programmazione di agenti intelligenti e l'elaborazione del linguaggio naturale

Paradigmi emergenti: programmazione reattiva e programmazione parallela

- Oltre ai paradigmi tradizionali, sono emersi nuovi paradigmi che affrontano sfide specifiche nell'ambito dello sviluppo del software
- La programmazione parallela si occupa della suddivisione dei compiti in sottocompiti che possono essere eseguiti contemporaneamente su più processori o core di elaborazione
- Questo paradigma è diventato cruciale per sfruttare al massimo le prestazioni dei moderni sistemi di elaborazione parallela e distribuita

