

# Programarea algoritmilor

Marinescu–Ghemeci Ruxandra

[verman@fmi.unibuc.ro](mailto:verman@fmi.unibuc.ro)

[ruxandra.marinescu@fmi.unibuc.ro](mailto:ruxandra.marinescu@fmi.unibuc.ro)

[ruxandra.marinescu@unibuc.ro](mailto:ruxandra.marinescu@unibuc.ro)

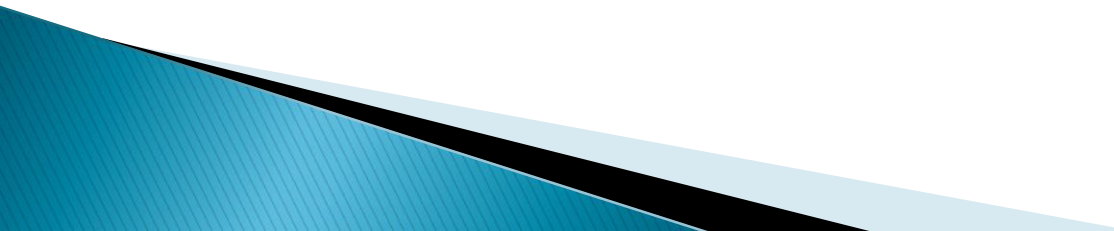


# Programa



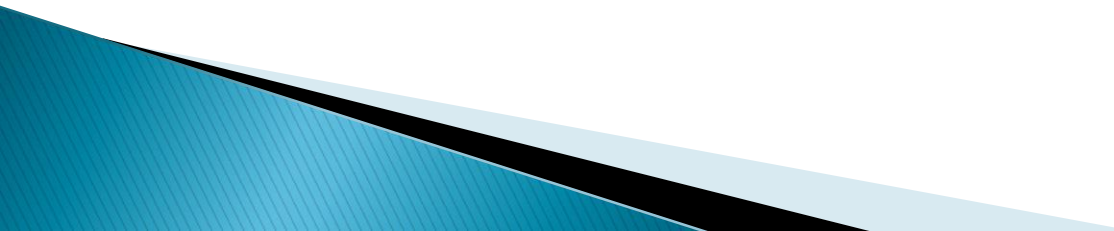
# Programa

## ► Introducere în limbajul Python

- Elemente de bază
  - Colecții
  - Șiruri de caractere
  - Funcții
  - Fișiere
  - Excepții
- 

# Programa

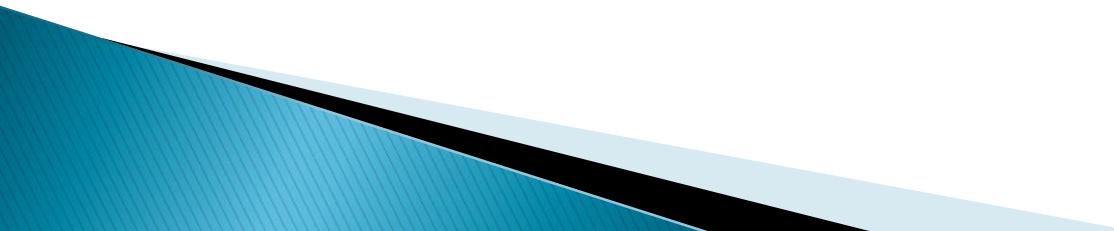
## ► Tehnici de programare:

- Algoritmi. Complexitate
  - Greedy
  - Divide et Impera
  - Programare dinamica
  - Backtracking
- 

# Obiective generale

- ▶ Însușirea principalelor tehnici de elaborare a algoritmilor și a tipurilor de probleme la care se pretează acestea
- ▶ Însușirea elementelor de bază ale limbajului Python, utilizarea corectă a structurilor de date și algoritmilor puși la dispoziție de acest limbaj pentru implementarea algoritmilor elaborați

# Obiective specifice

- ▶ cunoașterea principalelor tehnici de programare
  - ▶ abilități de utilizare a **structurilor de date** și tehnicilor potrivite în rezolvarea unei probleme
  - ▶ dezvoltarea unei gândiri algoritmice
  - ▶ abilități de justificare a **corectitudinii** algoritmilor propuși și de determinare a **complexității** acestora
  - ▶ abilități de implementare a algoritmilor în limbajul Python, de testare
- 

# Obiective. Motivații

## ▶ Python

- elemente de bază
- lucrul cu structuri de date

# Obiective. Motivații

## ► Python – avantaje

- sintaxa simplă, sugestivă
- dinamic
- de actualitate
- numeroase facilități (incluse automat): dezvoltare software, web, GUI, module pentru IA, ML (Google – motoare de căutare)
- portabil
- open-source: [www.python.org](http://www.python.org)
- garbage collection



# Objective. Motivații

## ► Tehnici de programare

- algoritmi eficienți

*"Perhaps the most important principle for the good algorithm designer is to refuse to be content" –*

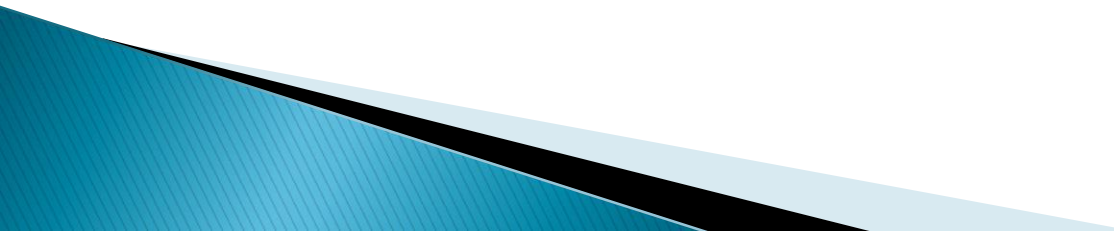
Aho, Hopcroft, and Ullman, The Design and Analysis of Computer Algorithms

# Obiective. Motivații

## ► Tehnici de programare

- algoritmi eficienți

### Exemple de probleme

- Aflarea minimului și maximului dintr-un vector
  - Cele mai apropiate două puncte dintr-o mulțime de puncte din plan dată
  - Numărul de inversiuni dintr-un vector
  - Înmulțirea a două numere / matrice
- 

# Obiective. Motivații

## ► Tehnici de programare

- algoritmi corecți

### Exemple de probleme

- Dată o mulțime de intervale, să se determine o submulțime de cardinal maxim de intervale care nu se suprapun
- Dată o mulțime de intervale, fiecare interval având asociată o pondere, să se determine o submulțime de intervale care nu se suprapun având ponderea totală maximă

# Obiective. Motivații

## ► Tehnici de programare

- algoritmi eficienți (chiar dacă există soluții evidente polinomiale – se poate mai bine?)
- corectitudinea algoritmilor – demonstrații
- probleme dificile  $\rightarrow$  NP-completitudine
- pentru ce tipuri de probleme se aplica metodele
- Complexitate – structuri de date

# Objective. Motivații

- Numeroase aplicații
  - Bioinformatică, procesare texte, imagini
  - Geometrie computațională
  - Căutare web, similitudini, aliniere
  - Probleme de planificare
  - Proiectare, jocuri, strategii
  - Baze de date – arbori de căutare optimi
- Probleme interviuri

# Structura


## ▶ Curs

- 2 ore pe săptămâna
- finalizat cu examen

## ▶ Laborator

- 2 ore la două săptămâni
- limbaj Python
- finalizat cu test de laborator

## ▶ Seminar

- 2 ore la două săptămâni
  - discuții probleme curs/laborator
  - nu este notat separat, subiecte legate de seminar se vor regăsi la examen
- 

# Evaluare



# Evaluate

- ▶ Test de laborator – în prima sâmbătă din ianuarie după vacanța de iarnă – **8 ianuarie 2022**
- ▶ Examen scris în sesiune
- ▶ **Nota finală** = media celor două note

Condiții necesare:

**Nota test laborator (nerotunjită)  $\geq 5$  puncte**

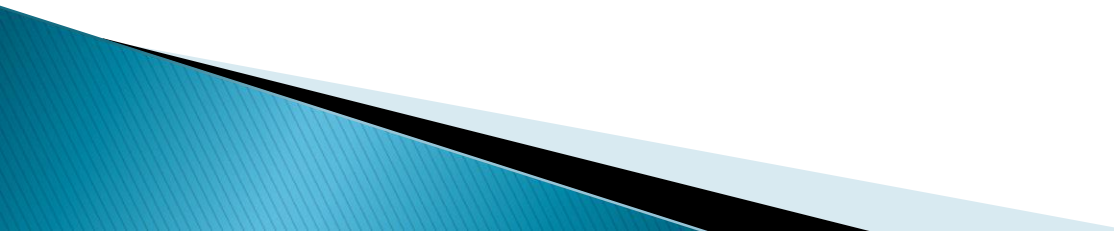
**Nota examen scris (nerotunjită)  $\geq 5$  puncte**



# BIBLIOGRAFIE

- ❖ Jon Kleinberg, Éva Tardos, **Algorithm Design**, Addison–Wesley 2005  
<https://www.cs.princeton.edu/~wayne/kleinberg-tardos/>
- ❖ T.H. Cormen, C.E. Leiserson, R.R. Rivest – **Introdúcere in algoritmi**, Mit Press, trad. Computer Libris Agora
- ❖ S. Dasgupta, C.H. Papadimitriou, U.V. Vazirani, **Algorithms**, McGraw–Hill, 2008

# BIBLIOGRAFIE

- ❖ Horia Georgescu. **Tehnici de programare.** Editura Universității din București 2005
  - ❖ Leon Livovschi, Horia Georgescu. **Sinteza și analiza algoritmilor.** 1986
  - ❖ Dana Lica, Mircea Pașoi, **Fundamentele programării,** L&S Infomat
- 

# BIBLIOGRAFIE

- ❖ **coursera.org**

Algorithms, Part II – Princeton University

Algorithms: Design and Analysis – Stanford University

- ❖ **MIT** <https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-006-introduction-to-algorithms-fall-2011/>

- ❖ **infoarena.ro**

# BIBLIOGRAFIE – Python

- ❖ <https://docs.python.org/3/>
- ❖ Magnus Lie Hetland– **Beginning Python From Novice to Professional** – Apress (2017)
- ❖ Naomi Ceder – **The Quick Python Book** –Manning Publications, 3rd ed (2018)

# Materialle curs, laborator, seminar

- ❖ MoodleUB
- ❖ MS Teams – secundar, materialele vor fi structurate pe Moodle
- ❖ Consultații