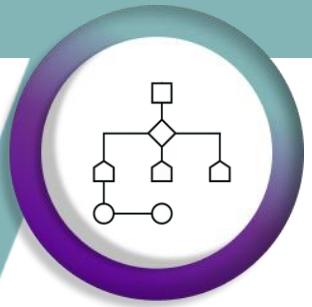
ALGORITAM

Boj projekta : 2023-2-IT03-KA220-YOU-000179130

Co-funded by
the European Union

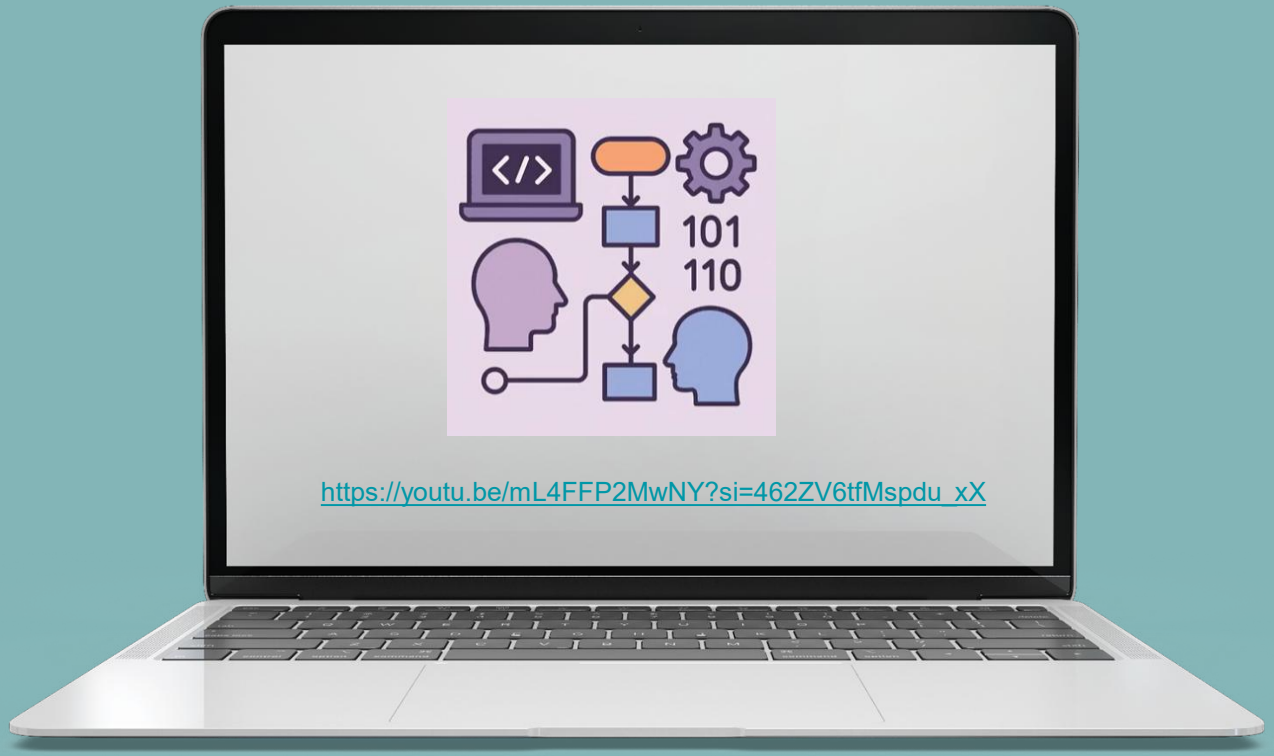
Napredni plan časa



- Definišite algoritam u tehničkom smislu i objasnite ga jednostavno.
- Razumeti efikasnost algoritma: znati šta znače „vremenska složenost“ i „prostorna složenost“.
- Identifikovati uobičajene tipove algoritama: sortiranje, pretraživanje, rekurzivni, iterativni, pohlepni itd.
- Analizirati algoritam korak po korak da bi se proverila ispravnost i efikasnost.
- Napisati jednostavne algoritme u pseudokodu ili dijagramima toka.

Co-funded by
the European Union

Video



https://youtu.be/mL4FFP2MwNY?si=462ZV6tfMspdu_xX

DEAF YOUNG CODE

Co-funded by the European Union

Napredni plan časa

Tehnička definicija:

Algoritam je konačan niz dobro definisanih instrukcija za rešavanje problema.

Co-funded by
the European Union

Ključne karakteristike

DEAF YOUNG CODE

1. Input: Podaci sa kojima počinjete.
2. Output: Rezultat ili rešenje.
3. Finiteness: Mora se završiti nakon ograničenog broja koraka.
4. Definitivnost: Svaki korak je jasno definisan.
5. Efikasnost: Svaki korak se može obaviti u razumnom vremenskom periodu.

Osnove algoritama

DEAF YOUNG CODE

Kako algoritmi funkcionišu

- Jasno definišite problem → Šta želimo da rešimo?
- Osmislite korak-po-korak proceduru → Koja su uputstva za dolazak do rešenja?
- Testirajte sa primerima → Proverite da li algoritam radi sa uzorcima ulaza.
- Analizirajte efikasnost → Koliko je brz? Koliko memorije koristi?
- Usavršite i optimizujte → Može li biti brži ili koristiti manje memorije?

Dijagram algoritma (Flowchart)

DEAF YOUNG CODE

Dijagram toka je vizuelni način da se algoritam prikaže korak po korak.
Pomaže vam da vidite logiku, ne samo da je pročitate.

Element	Simbol	Funkcija	Primer
Start / End	Oval 	Pokazuje gde algoritam počinje ili se završava	“Start” / “End”
Process / Step	Pravougaonik 	Radnja ili instrukcija	“Dodajte dva broja”
Decision / Condition	Dijamant 	Pitanje koje menja tok (Da/Ne)	“Ako je a > b?”
Input / Output	Paralelogram 	Podaci koji ulaze ili izlaze	“Unesite brojeve” / „Izbacite rezultat”
Arrow	→ 	Pokazuje smer toka	
Loop / Connector	Zakrivljena strelica 	Korak koji se ponavlja dok se ne ispuni uslov	“Ponavljajte dok se ne sortira”

Dijagram algoritma (Flowchart)

DEAF YOUNG CODE

Kako čitati dijagram toka

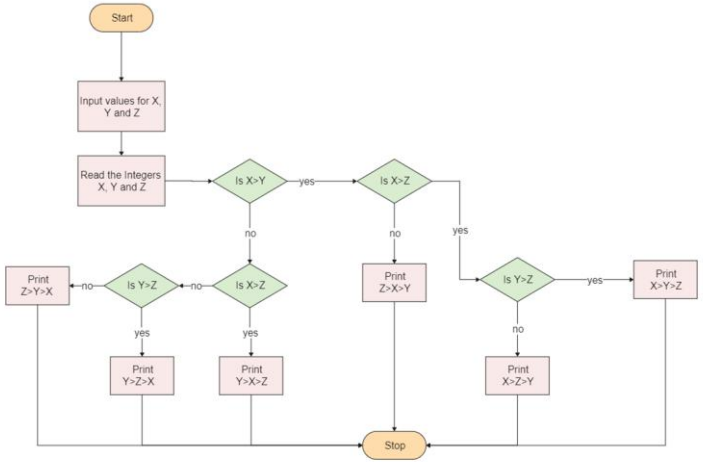
- Start → Pronađite oval „Start“..
- Input → Pogledajte koje informacije ulaze u sistem (na primer: brojevi, tekst).
- Process → Pratite svaki pravougaonik. To su radnje ili koraci.
- Decision → Kada dođete do dijamanta, postavlja se pitanje sa Da/Ne.
  Ako je odgovor „Da“, pratite jednu strelicu.
  Ako je odgovor „Ne“, pratite drugu.
- Loop → Kada se strelica vraća na prethodni korak, to znači „ponavljanje”
- Output → Na kraju, pogledajte rezultat ili odgovor.
- End → Proces se zaustavlja.

Primeri: Sortiranje brojeva

Zadatak: Sortiraj brojeve [4, 2, 7] od najmanjeg do najvećeg

1. Flowchart (Step-by-Step Visual)

```
Start
  |
  v
Input numbers: 4, 2, 7
  |
  v
Compare first two numbers:
Is 4 > 2? Yes → Swap
  |
  v
Numbers now: 2, 4, 7
  |
  v
Compare next numbers: 4 and 7
Is 4 > 7? No → Keep order
  |
  v
Output sorted numbers: 2, 4, 7
  |
  v
End
```



Savet za gluve učenike:

- **Koristite strelice i okvire za svaki korak.**
- **Označite radnju zamene drugom bojom.**
- **Jasno prikažite input → process → output.**

Co-funded by the European Union

Primeri: Sortiranje brojeva

Zadatak: Sortiraj brojeve [4, 2, 7] od najmanjeg do najvećeg

2. Pseudokod (Jednostavna tekstualna verzija)

```
Algorithm SortThreeNumbers:
  Input: a, b, c
  If a > b then
      Swap a and b
  If b > c then
      Swap b and c
  If a > b then
      Swap a and b
  Output: a, b, c
End Algorithm
```

Napomene:

- **Pseudokod je kao engleska uputstva, a ne programski jezik.**
- **Jasnoća korak po korak je ključna.**

Co-funded by the European Union

Primeri: Sortiranje brojeva

Zadatak: Sortiraj brojeve [4, 2, 7] od najmanjeg do najvećeg

3. Dijagram (Vizuelni pregled)

```
[Input: 4,2,7]
      |
      v
[Compare 4 and 2]
      |
      v
[Swap if needed] ---> [Numbers now: 2,4,7]
      |
      v
[Compare 4 and 7]
      |
      v
[Swap if needed] ---> [Numbers now: 2,4,7]
      |
      v
[Output: 2,4,7]
```

Co-funded by
the European Union

Ključne tačke

Flowchart = detaljna, korak-po-korak mapa algoritma.

- **Uključuje odluke, petlje, ulaze, izlaze.**
- **Odlično za kasnije pisanje pseudokoda ili kodiranje.**

Dijagram = vizuelni, jednostavan pregled.

- **Prikazuje glavne radnje i tok bez svih koraka.**
- **Pomaže u brzom razumevanju logike, posebno za vizuelne učenike.**

Co-funded by
the European Union

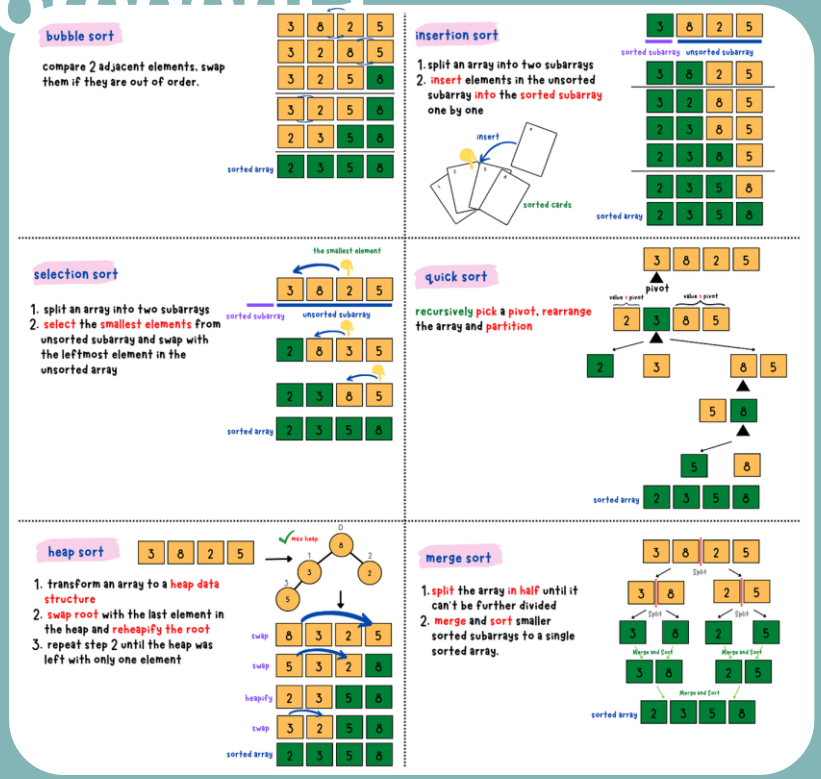
Važnost teorija algoritama

- **Vremenska složenost (notacija velikog O): Meri koliko brzo algoritam radi kako veličina ulaza raste.**
- **Prostorna složenost: Meri koliko memorije algoritam koristi.**
- **Uobičajeni tipovi algoritama :**
  - SOrting: sortiranje mehurićima, sortiranje spajanjem, brzo sortiranje.
  - Pretraživanje: Linearno pretraživanje, binarno pretraživanje.
  - Rekurzija: Algoritam poziva samog sebe (npr. faktorijel).
  - Pohlepni algoritmi: Pravljenje najboljeg izbora u svakom koraku (npr. najkraći put).
  - Dinamičko programiranje: Razbijanje problema na manje podprobleme.

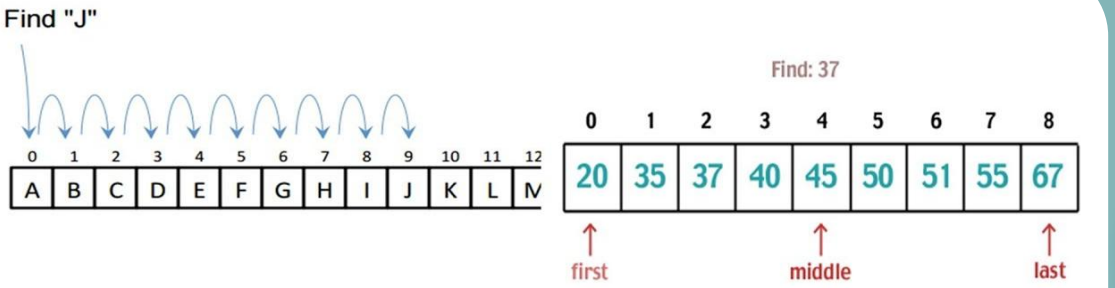
- **Strukture podataka: Algoritmima su često potrebne liste, nizovi, stabla ili grafikoni.**

Co-funded by the European Union

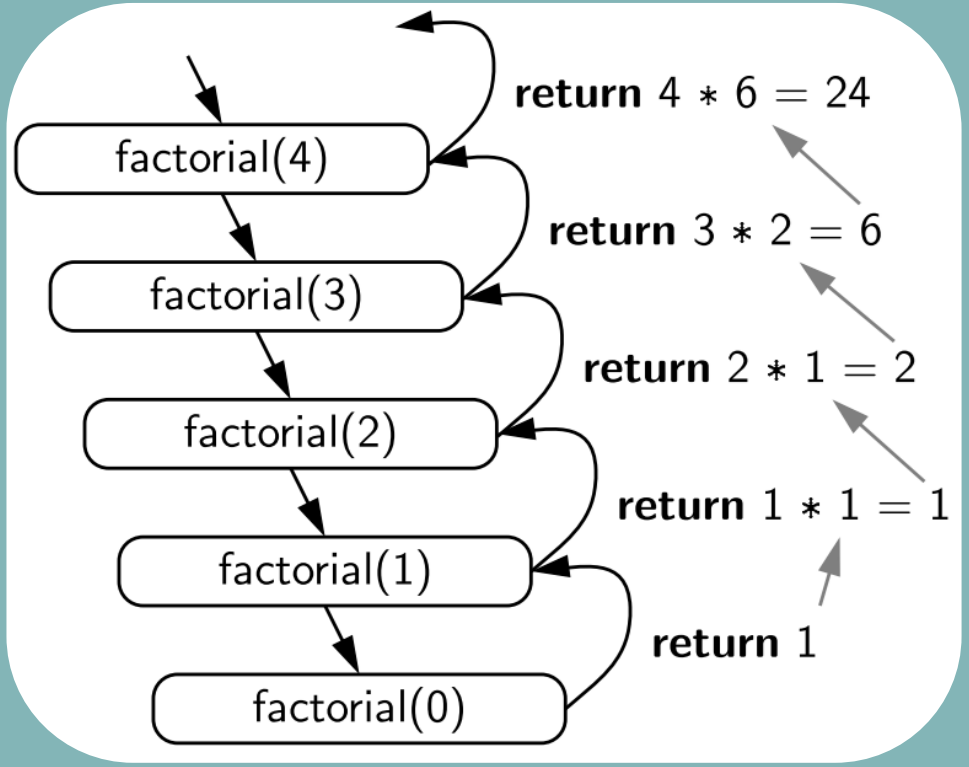
Sortiranje na osnovu



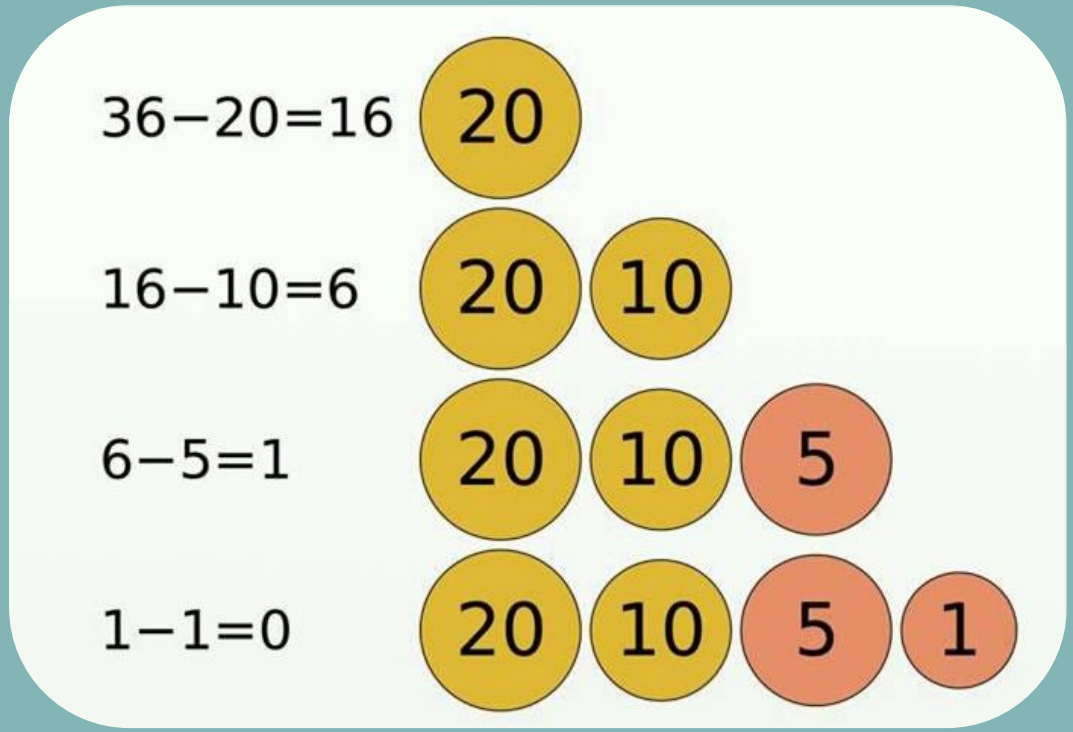
Sortiranje na osnovu poređenja



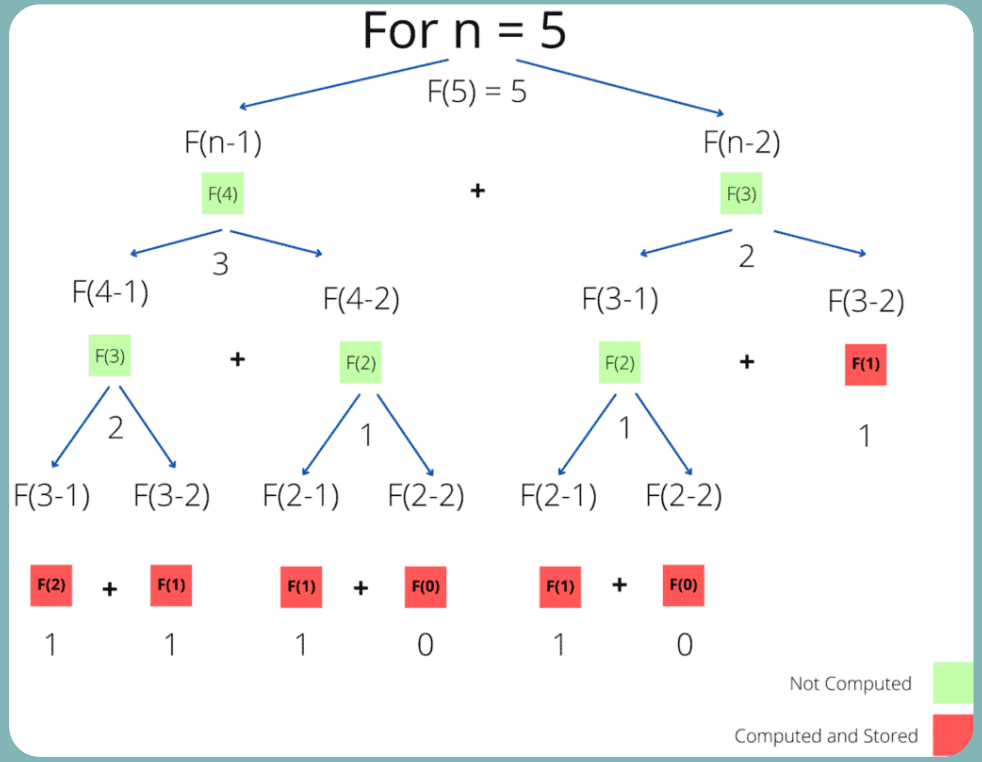
Sortiranje na osnovu poređenja



Sortiranje na osnovu poređenja



Sortiranje na osnovu poređenja



Ideja za interaktivnu vežbu (praktično učenje)

DEAF YOUNG CODE

Cilj

- Pomoći učenicima da vežbaju dizajn algoritama na zabavan i vizuelan način.
- Učvrstiti razumevanje koraka, odluka, petlji i efikasnosti.

Ideja za interaktivnu vežbu (praktično učenje)

DEAF YOUNG CODE

Korak 1: Izaberite jednostavan zadatak

- Primeri svakodnevnih zadataka:
- Spremanje čaja
- Sortiranje knjiga po visini
- Jutarnja rutina (buđenje → pranje zuba → doručak → oblačenje)
- Pakovanje ranca
- Sređivanje voća po veličini

Ideja za interaktivnu vežbu (praktično učenje)

DEAF YOUNG CODE

Korak 2: Nacrtajte dijagram toka

- Uputstva za učenike:
- Start → Nacrtajte simbol početka (oval).
- Input / Output → Prikažite podatke ili rezultat (paralelogram).
- Actions / Steps → Koristite pravougaonike za svaku akciju.
- Decision Points→ Koristite rombove za pitanja sa Da/Ne.
- Loops → Nacrtajte strelice koje se vraćaju na ponavljanje koraka ako je potrebno.
- End → Prikažite konačni rezultat ili završetak (oval).
- Savet: Koristite boje ili lepljive beleške za akcije, odluke i izlaze.

Ideja za interaktivnu vežbu (praktično učenje)

Korak 3: Deljenje i upoređivanje

- Zamolite svakog učenika ili grupu da predstavi svoj flowchart.

Diskutujte:

- Da li su uključili sve važne korake?
- Da li postoje nepotrebni koraci koji se mogu ukloniti?
- Da li se zadatak može obaviti brže ili efikasnije?

Ideja za interaktivnu vežbu (praktično učenje)

DEAF YOUNG CODE

Korak 4: Opcioni izazov (Efikasnost)

- Predstavite ideju efikasnosti algoritma:
  - Manje koraka = brži algoritam
  - Ponavljanje nepotrebnih koraka = sporiji algoritam

Ideja za interaktivnu vežbu (praktično učenje)

Korak 4: Opcioni izazov (Efikasnost)

- Primer: Sortiranje 3 knjige
  - **Učenik A:** Proverava sve parove dva puta → 6 koraka
  - **Učenik B:** Samo pametne zamene → 3 koraka
- Pitajte: **Koji je algoritam bolji? Zašto?**
- Pokažite da čak i **jednostavni svakodnevni zadaci** mogu imati „efikasnije“ algoritme..

Ideja za interaktivnu vežbu (praktično učenje)

Korak 5: Praktični bonus (kinestetičko učenje)

- Koristite **stvarne predmete**: knjige, šolje, lepljive papiriće.
- Učenici fizički **odglumljuju korake**, prateći sopstveni dijagram toka.
- Pomaže **vizuelnim i taktilnim učenicima** da razumeju redosled, petlje i odluke.

Hvala vam na pažnji!

DEAF YOUNG CODE

Pratite nas na sledećoj prezentaciji!

Finansirano od strane Evropske unije. Izraženi stavovi i mišljenja su, međutim, isključivo stavovi i mišljenja autora/autora i ne odražavaju nužno stavove Evropske unije ili Izvršne agencije za obrazovanje i kulturu (EACEA). Ni Evropska unija ni EACEA ne mogu biti odgovorne za njih.

Boj projekta : 2023-2-IT03-KA220-YOU-000179130

Co-funded by the European Union