

Meteo Ústav SR získal od partnerov dataset denných meraní teplôt z rôznych meteorologických staníc po celom Slovensku.

Potrebuju od vás vyvinúť rýchly analytický prototyp, ktorý:

- načíta a vygeneruje simulované merania (matica),
- vypočíta základné prehľadové štatistiky,
- nájde extrémne hodnoty (napr. veľmi vysoké či nízke teploty),
- skontroluje výskyt anomálií rekurzívnym algoritmom,
- pripraví prehľadný report.

Každý riadok matice = **jedna meteorologická stanica**

Každý stĺpec = **teplota nameraná v konkrétny deň**

Napríklad matica  $5 \times 7$  simuluje 5 staníc počas 7 dní.

---

## MODULY PRE TÍM (konkrétne zadania)

---

### ♦ MODUL A – Generátor klimatických dát

(študent A – polia, matice, sekvencia)

Zadanie:

Naprogramuj funkciu, ktorá:

1. Načíta od používateľa počet staníc (m) a počet dní merania (n).
2. Vygeneruje **maticu teplôt**  $m \times n$ , kde každé meranie je náhodné číslo od **-15 do +40 °C**.
3. Vypíše tabuľku vo formáte:

Stanica\Deň	1	2	3	4	5
-------------	---	---	---	---	---

St1	12	14	18	20	21
-----	----	----	----	----	----

St2	10	11	17	19	23
-----	----	----	----	----	----

...

4. Odovzdá maticu ďalšiemu modulu.
-

## ♦ MODUL B – Analytik

(študent B – cykly, podmienky, polia)

Zadanie:

Naprogramuj funkcie:

### 1. Priemerná teplota každej stanice

– výstup: pole dĺžky m

– napr.: [12.4, 15.2, 8.7, ...]

### 2. Priemerná teplota pre každý deň

– výstup: pole dĺžky n

### 3. Najteplejší deň (celkovo)

– funkcia vráti (max teplota, stanica, deň)

### 4. Najchladnejší deň

– podobne ako vyššie

### 5. Počet dní so zamrznutím (teplota < 0) pre každú stanicu

– výstup: pole m

Toto všetko sa bude zobrazovať v reporte.

---

## ♦ MODUL C – Detektor extrémnych oblastí (rekurzia – DFS)

Študent C dostane úlohu, ktorá je:

- reálna,
- používaná v GIS/Meteo systémoch,
- učí pekný algoritmus (DFS),
- zapadá do témy.

Cieľ:

V matici hľadať súvislé oblasti extrémnych teplôt, kde každý extrém je hodnota:

- 35 °C (vlny horúčav)  
alebo
- < -10 °C (extrémny mráz)

Oblasť je súvislá, ak sú bunky spojené hore/dole/vľavo/vpravo.

Úlohy modulu:

---

**1** *Rekurzívna funkcia je\_extrem(x)*

Jednoduchá funkcia rozhodne, či ide o extrém.

---

**2** *Rekurzívne prehľadávanie matice (DFS)*

Implementovať funkciu:

def dfs(mat, r, c, navstivene):

Označí všetky bunky danej extrémnej oblasti.

Úloha je:

- ak bunka je extrém → označiť ju ako navštívenú,
  - rekurzívne zavolať DFS pre susedov,
  - zistiť veľkosť tejto oblasti.
- 

**3** *Nájsť a spočítať všetky extrémne oblasti v matici*

Funkcia:

def najdi\_extremne\_oblasti(mat):

Nájde počet súvislých extrémnych oblastí + ich veľkosti.

Výstup napr.:

Našli sa 2 extrémne oblasti:

- Oblasť 1: veľkosť 3 bunky
  - Oblasť 2: veľkosť 5 buniek
- 

Prečo je toto realistické?

Lebo veľmi podobný algoritmus sa používa v:

- detekcii búrok na radarových snímkoch,
- detekcii ľadových plôch,
- detekcii požiarov zo satelitných dát,

- segmentácii obrázkov,
- hľadání zhlukov v GPS dátach,
- v Google Maps pri vyznačovaní záplavových oblastí.

Toto je *reálna úloha pre juniorského analytika alebo GIS programátora* – nie umelý príklad.

#### ♦ MODUL D – Reportovací modul (integrátor)

(Študent D – formátovanie, testovanie, main())

Zadanie:

- Priprav "main" program.
- Zavolá všetky funkcie z modulov A, B, C.
- Zobrazí výsledky v prehľadnom formáte:

#### Poznámka: Čo je DFS? (Depth-First Search)

**DFS = „Depth-First Search“ → hĺbkové prehľadávanie.**

Je to algoritmus, ktorý prehľadáva **graf** alebo **mrežu (maticu)** tak, že:

1. začne v nejakom bode,
2. ide **čo najhlbšie** jedným smerom,
3. až keď už nemôže ísť ďalej, vráti sa späť,
4. a pokračuje ďalšou cestou.

Je to podobné, ako keď v labyrinte:

- vždy ideš dopredu,
- ak uviazneš, vrátiš sa,
- a skúsiš inú odbočku.

#### Ako to súvisí s rekuriou?

DFS sa často implementuje pomocou **rekurzie**, pretože každé volanie funkcie „ide o úroveň hlbšie“.

Rekurzívna verzia DFS vyzerá zhruba takto:

dfs(pozícia):

    označ túto pozíciu ako navštívenú

    pre každého suseda:

        ak sused ešte nebol navštívený:

            rekurzívne zavolaj dfs(sused)

---

## Kde sa DFS používa v realite?

Toto nie je školská vec — DFS používa obrovské množstvo systémov:

### v mapách (Google Maps)

- hľadanie ciest,
- spájanie oblastí (povodne, dopravné zápchy, lesy).

### v meteorológii

- zisťovanie, či sú extrémne hodnoty súvislé,
- segmentácia radarových snímok,
- práca s teplotnými mapami.

### v počítačovej grafike

- flood fill (ako vedro vo Photoshope),
- rozpoznávanie objektov na obrázku.

### v hrách

- AI hľadá cestu bludiskom,
- zisťovanie súvislých území (napr. v strategických hrách).

### v analýze dát

- vyhľadávanie klastrov (zhlukov).
- 

## DFS na matici (pre náš projekt)

Predstav si maticu ako **mapu**, kde niektoré bunky sú „extrémne“.

Chceme zistiť:

- či sú extrémny izolované body,
- alebo tvoria väčšie súvislé celky.

DFS urobí to, že:

1. nájde extrémny bod,
2. rekurzívne preskúma jeho susedov,
3. tých susedov znova preskúma ďalej,
4. kým neprejde celú extrémnu oblasť.

Takto získame:

- **počet extrémnych oblastí,**
- **veľkosť každej oblasti,**
- **najväčšiu oblasť.**

### Jednoduchý príklad

Matica:

[ 1, X, X, 0 ]

[ 0, X, 0, 0 ]

[ X, 0, X, X ]

Ak X sú extrémny:

DFS zistí:

- 1. oblasť: tri bunky hore (0,1)(0,2)(1,1)
- 2. oblasť: jedna bunka (2,0)
- 3. oblasť: dve bunky (2,2)(2,3)