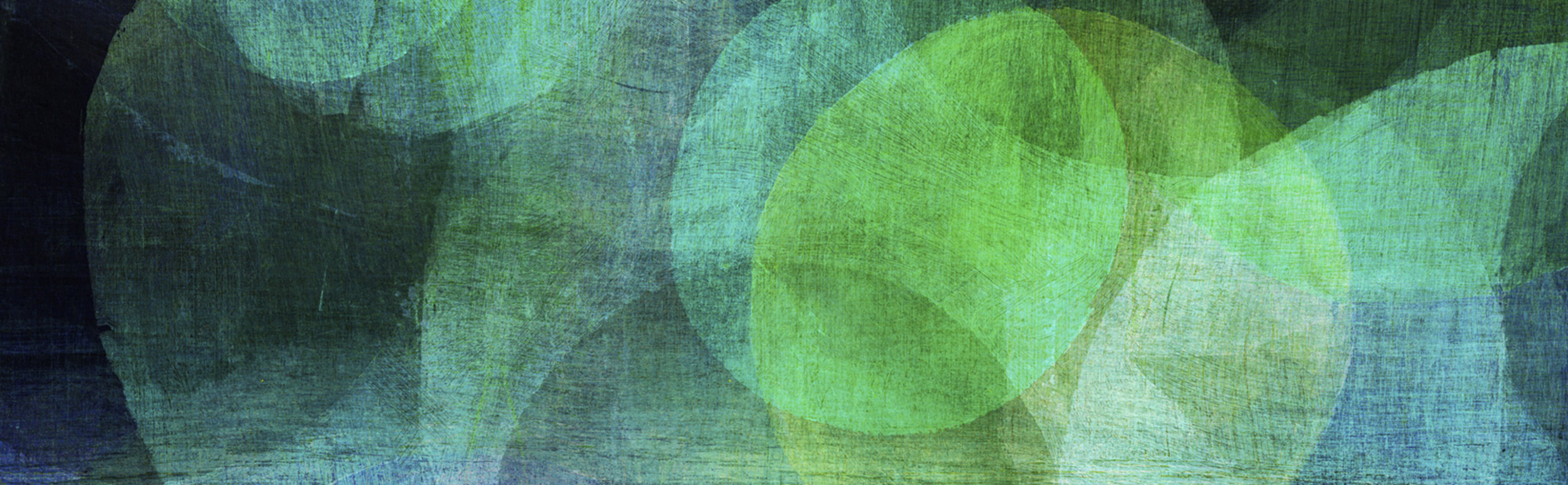




Algoritmizace (NPRG062)

Cvičení 3: Teorie čísel



Cvičení 3.1: Eratosthenovo síto

- ❖ Vylepšete implementaci algoritmu Eratosthenova síta tak, aby seznam `je_prvocislo` neevidoval sudá čísla
- ❖ Chceme uspořit paměť i čas

```
1 # Eratosthenovo síto pro hledání prvočísel
2 horni_mez = int(input("Zadejte horní mez prvočísel: "))
3
4 prvocisla = [] # Inicializace prázdného seznamu prvočísel
5 je_prvocislo = [False, False] + [True] * (horni_mez - 1)
6
7 for cislo in range(2, horni_mez + 1):
8     if je_prvocislo[cislo]:
9         prvocisla.append(cislo)
10        for nasobek in range(cislo ** 2, horni_mez + 1, cislo):
11            je_prvocislo[nasobek] = False # Násobek čísla není prvočíslem
12
13 # Výpis nalezených prvočísel
14 print(f"Prvočísla do {horni_mez} jsou:")
15 for prvocislo in prvocisla:
16     print(prvocislo)
```

Cvičení 3.2: Prvočíselný rozklad

- ❖ Napište *program* v jazyce Python, který k zadanému přirozenému číslu n *nalezne* jeho *prvočíselný rozklad*

- ❖ **Těžší varianta**

- ❖ Vytvořte funkci



Cvičení 3.3: Nejmenší společný násobek

- ❖ Navrhněte efektivní *algoritmus výpočtu nejmenšího společného násobku* dvou zadaných přirozených čísel
- ❖ Vstupem algoritmu budou dvě hodnoty, pro které hledáme nejmenší společný násobek
- ❖ Výstupem bude nejmenší společný násobek
- ❖ Návod: Využijte Euklidův algoritmus pro nalezení největšího společného dělitele

```
1 # Euklidův algoritmus pro výpočet největšího společného dělitele
2 prvni_cislo = int(input("Zadejte první číslo: "))
3 druhe_cislo = int(input("Zadejte druhé číslo: "))
4
5 while druhe_cislo > 0:
6     prvni_cislo, druhe_cislo = druhe_cislo, prvni_cislo % druhe_cislo
7
8 print(f"Největším společným dělitelem čísel je {prvni_cislo}")
```