Sito Eratostenesa w języku Python

- Wprowadzenie
- Przeczytaj
- Symulacja interaktywna
- Sprawdź się
- Dla nauczyciela

Źródło: congerdesign, domena publiczna.

Sito Eratostenesa jest algorytmem wyznaczającym liczby pierwsze z zadanego przedziału $\langle 2;\ n \rangle$. Liczby pierwsze znajdują zastosowanie między innymi w kryptografii, o czym przeczytasz m.in. w e-materiałach:

- Zastosowanie liczb pierwszych,
- Szyfr RSA.

Ten e-materiał poświęcony jest implementacji algorytmu sita Eratostenesa w języku Python.

Ciekawi cię, jak wyglądają implementacje w innych językach programowania? Możesz się z nimi zapoznać w dwóch pozostałych e-materiałach z tej serii:

- Sito Eratostenesa w języku C++,
- Sito Eratostenesa w języku Java.

Więcej zadań? Sięgnij do: Sito Eratostenesa – zadania maturalne.

Twoje cele

- Prześledzisz metodę wyznaczania liczb pierwszych za pomocą algorytmu sito Eratostenesa.
- Zaimplementujesz algorytm Eratostenesa w języku Python.

- Rozwiążesz kilka problemów wymagających użycia sita Eratostenesa.

Przeczytaj

Przykład implementacji sita Eratostenesa w języku Python

```
1  def sito_eratostenesa_for(n):
2      from math import sqrt
3      lista_liczb = []
4      lista = [0] + (n * [1])
5      max_liczba = int(sqrt(n))
6      for indeks in range(2,max_liczba+1):
7          if lista[indeks]:
8              for indeks_2 in range(indeks*indeks,n+1,indeks):
9                  lista[indeks_2] = 0
10
11     for element in range(2,n+1):
12         if lista[element]:
13             lista_liczb.append(element)
14
15     print(lista_liczb)
```

Najważniejsze elementy przedstawionego kodu:

- obiekt `lista_liczb` będzie wynikową listą liczb pierwszych w przeszukiwanym zakresie;
- indeksy obiektu `lista` są reprezentacją zbioru liczb naturalnych z zakresu `[0, n]`. Początkowo wszystkie elementy traktowane są jako liczby „pierwsze” i mają wartość `1` (poza elementem o indeksie `0`). Później (zgodnie z algorytmem) niektóre z nich będziemy eliminować („wykreślać”), zastępując je zerami;
- zmienna `max_liczba` ma wartość pierwiastka kwadratowego z `n`, który określa górną granicę liczby, której wielokrotność wykreślamy. Zgodnie z algorytmem pierwsza pętla działa od liczby `2` do pierwiastka z liczby `n` (więc jako argumenty funkcji `range()` musimy wpisać `range(2, max_liczba+1)`).

Jeśli podczas wykonywania pętli wartość elementu `lista[indeks]` (linia nr 7) będzie wynosić `1` (element będzie liczbą pierwszą), wówczas w wewnętrznej pętli usuwamy jej wielokrotności.

Na końcu sprawdzamy wszystkie elementy obiektu `lista`. Liczbami pierwszymi są indeksy elementów o wartości 1. Dopisujemy je do obiektu `lista_liczb`, a wynik wypisujemy.

Efekt działania programu dla liczb z zakresu od `2` do `120`:

```
sito_eratostenesa_for(120)
[2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53,
59, 61, 67, 71, 73, 79, 83, 89, 97, 101, 103, 107, 109, 113]
```

Problem 1

Możemy wprowadzić pewne elementy modyfikujące działanie funkcji. Spróbujmy zdefiniować funkcję `sito_eratostenesa_for(n)` w taki sposób, aby dla `n < 2` nie próbowała ona nic obliczać, ale zwracała wartość `False`.

```
```

```
```

Słownik

liczba pierwsza

liczba naturalna większa od 1, mająca tylko dwa dzielniki – 1 oraz siebie samą

pętla zagnieżdżona

pętla, która znajduje się wewnątrz innej pętli

Symulacja interaktywna

Polecenie 1

Napisz program wyszukujący w zadanym przedziale $\langle 2;\ n\rangle$ liczby pierwsze.

Swoje rozwiązanie przetestuj dla n wynoszącego 20.

Specyfikacja problemu:

Dane:

- n – liczba naturalna dodatnia; górna granica przedziału

Wynik:

Program na standardowym wyjściu wypisuje kolejne liczby pierwsze z zakresu $\langle 2;\ n\rangle$.

Sprawdź poprawność swojego rozwiązania, wykorzystując symulację.

```
# Tutaj dodaj własny kod. Użyj funkcji
# print()
```

```
```

Polecenie 2

Przeanalizuj działanie sita Eratostenesa. Zwróć uwagę, że wykreślamy liczby, zaczynając dopiero od kwadratu kolejnych liczb pierwszych.

Zasób interaktywny dostępny pod adresem https://zpe.gov.pl/a/D16IUjS9a

Źródło: Contentplus.pl sp. z o.o., licencja: CC BY-SA 3.0.

Polecenie 3

Przetestuj działanie symulacji interaktywnej. Poszukaj informacji na temat innych metod wyszukiwania liczb pierwszych. Porównaj, jak szybko wyszukują liczby pierwsze w stosunku do algorytmu Eratostenesa.

Sprawdź się

Pokaż ćwiczenia:

Ćwiczenie 1

Piotr Bajtocki prowadzi firmę budowlaną. Zgodnie z prawem ma obowiązek zaopatrzyć swoich pracowników w wodę. Butelki wody o pojemności 1,5 litra są pakowane w zgrzewkach. W każdej ze zgrzewek zapakowano pewną liczbę butelek. Liczba ta jest liczbą pierwszą. Najmniejsza zgrzewka opakowuje `m` butelek; największa `n`. Litr wody kosztuje `x` złotych. Budżet Piotra wynosi `y` złotych. Jak dużą zgrzewkę wody może kupić?

Jeśli liczba butelek, które w zgrzewce może kupić pan Bajtocki, jest większa od `n`, oznacza to, że jego budżet pozwala na zakup największej zgrzewki.

W swoim rozwiązaniu nie wykorzystuj funkcji `sqrt`.

Swoje rozwiązanie przetestuj dla `x` wynoszącego 0,80; `y` równego 70, `n` równego 97, `m` wynoszącego 5.

Specyfikacja problemu:

Dane:

- `x` – cena litra wody; liczba zmiennoprzecinkowa dodatnia
- `y` – budżet Piotra; liczba zmiennoprzecinkowa dodatnia
- `n` – maksymalna liczba butelek w zgrzewce; liczba naturalna; $n > 11$
- `m` – liczba butelek w najmniejszej zgrzewce; liczba naturalna; $m \geq 5$

Wynik:

Na standardowym wyjściu program wypisuje największą liczbę butelek w zgrzewce, która mieści się w budżecie Piotra.

Twoje zadania

1. Program wypisuje największą liczbę butelek w zgrzewce, która mieści się w budżecie Piotra.

```python
n = 97
m = 5
x = 0.8
y = 70.00
# Tutaj dodaj kod.
# Do wypisywania użyj funkcji: print()
```

```python

```

Ćwiczenie 2

Zygmunt Bajtek otrzymał od rodziców na urodziny prezent – zegarek analogowy. Zafascynowany tym podarkiem postanowił policzyć, ile razy wskazówka minutowa wskaże liczbę pierwszą. Uzyskawszy tę wiedzę, policzył kąty ϕ między osią biegnącą od środka tarczy do minutowego 0, a wskazówką minutową wskazującą liczby pierwsze.

Zapoznaj się z ilustracją załączoną do zadania, przedstawiono na niej przykładowe oznaczenie kąta.

Napisz program, który w kolejnych liniach wypisze kolejne wartości ϕ zaczynając od 3-ciej minuty.

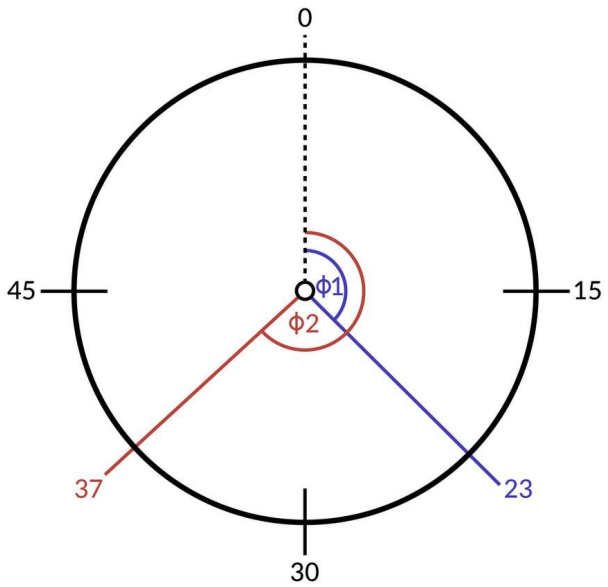
Specyfikacja problemu:

Dane:

- `A` – 61-elementowa tablica wartości logicznych wypełniona wartościami `true`

Wynik:

- Na standardowym wyjściu program wypisuje w kolejnych wierszach wartości kąta ϕ.



Źródło: Contentplus.pl sp. z o.o., licencja: CC BY-SA 3.0.

Twoje zadania

1. Program wypisuje kolejne wartości kąta fi.

```
A = [True] * 61
# Tutaj dodaj kod. Program wypisuje kolejne wartości kąta fi.
# Do wypisywania użyj funkcji: print()
```

```
```

Ćwiczenie 3

Hipoteza Goldbacha zakłada, że każda liczba naturalna parzysta większa od 2 jest sumą dwóch liczb pierwszych. Napisz program weryfikujący hipotezę Goldbacha dla liczb parzystych nie większych od n.

Swoje rozwiązanie zweryfikuj dla n wynoszącego 1000.

Specyfikacja problemu:

Dane:

- n – parzysta liczba naturalna dodatnia

Wynik:

Na standardowym wyjściu program wypisuje równania potwierdzające hipotezę Goldbacha, zgodnie z poniższym schematem; *NIE* w przeciwnym przypadku.

Przykład wyjścia:

Lewy składnik dodawania jest mniejszy od prawego; kolejne sumy wypisywane są rosnąco:

```
2 + 2 = 4
3 + 3 = 6
3 + 5 = 8
...
7 + 991 = 998
```

Twoje zadania

1. Program wypisuje kolejne równania potwierdzające hipotezę Goldbacha; słowo NIE w przeciwnym przypadku.

```
n = 1000
```

```
# Tutaj dopisz swój kod. Do wydruku każdego z równań lub słowa NIE użyj funkcji print()
```

```
```

Dla nauczyciela

Autor: Adam Jurkiewicz
Przedmiot: Informatyka

Temat: Sito Eratostenesa w języku Python

Grupa docelowa:

Liceum ogólnokształcące i technikum, liceum ogólnokształcące, technikum, zakres rozszerzony

Podstawa programowa:

Cele kształcenia – wymagania ogólne

I. Rozumienie, analizowanie i rozwiązywanie problemów na bazie logicznego i abstrakcyjnego myślenia, myślenia algorytmicznego i sposobów reprezentowania informacji.

II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera oraz innych urządzeń cyfrowych: układanie i programowanie algorytmów, organizowanie, wyszukiwanie i udostępnianie informacji, posługiwanie się aplikacjami komputerowymi.

Treści nauczania – wymagania szczegółowe

II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera i innych urządzeń cyfrowych.

Zakres rozszerzony. Uczeń spełnia wymagania określone dla zakresu podstawowego, a ponadto:

3) sprawnie posługuje się zintegrowanym środowiskiem programistycznym przy pisaniu, uruchamianiu i testowaniu programów;

I + II. Zakres rozszerzony. Uczeń spełnia wymagania określone dla zakresu podstawowego, a ponadto:

1) zapisuje za pomocą listy kroków, schematu blokowego lub pseudokodu, i implementuje w wybranym języku programowania, algorytmy poznane na wcześniejszych etapach oraz algorytmy:

c) generowania liczb pierwszych metodą sita Eratostenesa,

Kształtowane kompetencje kluczowe:

- kompetencje cyfrowe;
- kompetencje osobiste, społeczne i w zakresie umiejętności uczenia się;
- kompetencje matematyczne oraz kompetencje w zakresie nauk przyrodniczych, technologii i inżynierii.

Cele operacyjne (językiem ucznia):

- Prześledzisz metodę wyznaczania liczb pierwszych za pomocą algorytmu sito Eratostenesa.
- Zaimplementujesz algorytm Eratostenesa w języku Python.
- Rozwiążesz kilka problemów wymagających użycia sita Eratostenesa.

Strategie nauczania:

- konstruktywizm;
- konektywizm.

Metody i techniki nauczania:

- dyskusja;
- rozmowa nauczająca z wykorzystaniem multimedium i ćwiczeń interaktywnych;
- ćwiczenia praktyczne.

Formy pracy:

- praca indywidualna;
- praca w parach;
- praca w grupach;
- praca całego zespołu klasowego.

Środki dydaktyczne:

- komputery z głośnikami, słuchawkami i dostępem do internetu;
- zasoby multimedialne zawarte w e-materiale;
- tablica interaktywna/tablica, pisak/kreda;
- oprogramowanie dla języka Python 3 (lub nowszej wersji), w tym PyCharm lub IDLE.

Przebieg lekcji

Przed lekcją:

1. **Przygotowanie do zajęć.** Nauczyciel loguje się na platformie i udostępnia e-materiał: „Sito Eratostenesa w języku Python”. Nauczyciel prosi uczniów o zapoznanie się z treściami w sekcji „Przeczytaj”.

Faza wstępna:

1. Nauczyciel wyświetla i odczytuje temat lekcji oraz cele zajęć. Prosi uczniów o sformułowanie kryteriów sukcesu.

Faza realizacyjna:

1. **Praca z multimedium.** Nauczyciel wyświetla zawartość sekcji „Symulacja interaktywna". Uczniowie zapoznają się z poleceniem 1. W parach tworzą rozwiązanie, a następnie sprawdzają je, wykorzystując symulację.
   Następnie wykonują polecenie 2: „Przeanalizuj działanie sita Eratostenesa. Zwróć uwagę że wykreślamy liczby, zaczynając dopiero od kwadratu kolejnych liczb pierwszych".
2. **Ćwiczenie umiejętności.** Prowadzący zapowiada uczniom, że będą rozwiązywać ćwiczenie nr 1 z sekcji „Sprawdź się". Każdy z uczniów robi to samodzielnie. Po ustalonym czasie następuje porównanie napisanych kodów podczas wspólnego omówienia rozwiązań.
3. Nauczyciel dzieli uczniów na grupy czteroosobowe. Uczniowie wykonują ćwiczenie nr 2 z sekcji „Sprawdź się", a następnie każda grupa wyznacza jedną osobę, którą wymienia się z inną grupą. W nowych zespołach uczniowie porównują swój kod i wybierają najbardziej efektywne rozwiązanie.

Faza podsumowująca:

1. Na koniec zajęć z programowania w języku Python nauczyciel prosi uczniów o rozwinięcie zdania: „Na dzisiejszych zajęciach nauczyłam/łem się jak...".

Praca domowa:

1. Napisz program, który będzie generować liczby pierwsze metodą sita Eratostenesa z zakresu między `n1` a `n` podanymi przez użytkownika. Wyliczone liczby powinny zostać wypisane na ekranie jedna po drugiej, z przecinkami pomiędzy nimi, w postaci długiego napisu.

Materiały pomocnicze:

- Oficjalna dokumentacja techniczna dla języka Python 3 (lub nowszej wersji).
- Oficjalna dokumentacja techniczna dla oprogramowania PyCharm lub IDLE.

Wskazówki metodyczne:

- Uczniowie mogą wykorzystać multimedium w sekcji „Symulacja interaktywna" do przygotowania się do lekcji powtórkowej.