

Numer:	<b>MATINF 1/2023</b>	Stron:	6
Data wydania:	28 sierpnia 2023	Druk:	3
Adresy Redakcji:	<a href="mailto:matinf@poczta.onet.pl">matinf@poczta.onet.pl</a>	Witryny informacyjne:	<a href="https://github.com/czasopismo-MATINF/czasopismo-MATINF">https://github.com/czasopismo-MATINF/czasopismo-MATINF</a>
Czytelnik:	osoba samodzielnie ucząca się, student		
Cel:	systematyczne kursy podstaw programowania, matematyki, algebry i analizy matematycznej, artykuły o takowej tematyce; recenzje książek do nauki, narzędzi do programowania, listy zasobów informatycznych, serwisów internetowych; po dwudziestu latach wybuchu programistycznego niepotrzebna już promocja programowania, ale powrót na jego podstawowy poziom i znalezienie miejsca dla jego podstaw przy potwornie zwiększającym się stopniu skomplikowania algorytmów oraz narzędzi programistycznych; systematyczny zbiór powszechnie znanych pomysłów i idei;		

## Wstęp do bieżącego numeru:

W tym numerze wrześniowym Redakcja publikuje pierwsze szkice z kursów algebry i analizy matematycznej w oparciu o książki S.Lang'a „Algebra” i G.M. Fichtenholz'a „Rachunek różniczkowy i całkowy” oraz pierwsze sprawozdania z kursów poprzez języki programowania Python oraz Java.

## Spis aktualnie rozwijanej zawartości:

kurs programowania w języku Java	narzędzia programistyczne	kurs programowania w języku Python	szkice kursu z analizy matematycznej	historia matematyki przełomu XVIII i XIX wieku
listy serwisów internetowych	recenzje książek	algorytmika	szkice kursu z algebry	indeks wiadomości technicznych

# Kurs programowania poprzez język Python.

## Sprawozdanie 1.

Jest rok 2023. Nie można nauczyć się programować bez papierowej książki w rękę. Książek jest mnóstwo, Redakcja wybór pozostawia czytelnikowi. Po wyszukaniu hasła "Python tutorial" na pierwszej stronie wyszukiwarki internetowej, odrzuceniu nie mających czytelnego spisu treści lub tłumaczenia na język polski wyników wyszukiwania, Redakcja wybrała dwa kursy do nauki, których adresy są zamieszczone poniżej. W starych wynikach wyszukiwania korzystania Redakcji z internetu odnalazł się jeszcze jeden adres ze spisem treści zaawansowanych treści. Również zamieszczony poniżej. Redakcja postanowiła dokładnie przeczytać kurs z dokumentacji, ponieważ w ten sam sposób uczyła się programować w innych językach.

Języki programowania służą do tego, by dawać rozkazy robotom.

Rozdziały 1-3. Wygląda na to, że język Python jest popularny z powodu SPL ("Standard Python Library"). Biblioteka jest zestawem umiejętności robota. Programiści języka Python mogą do tej biblioteki dodawać własne przez siebie stworzone umiejętności - moduły. Język Python pozwala łatwiej wyrażać skomplikowane instrukcje niż inne języki programowania w początkach XX wieku: powiedzieć więcej w małej ilości słów. Podobno łatwo go wbudować do urządzeń tak by można je było programować: na przykład napisać program do włączenia światła i czajnika do zaparzania herbaty zaraz po zadziałaniu porannego budzika.

Po przejrzeniu wstępu do języka Python Redakcji udało się: obliczyć ilość sekund w XIX i w XX wieku (ponad trzy miliardy i sto milionów), skorzystać z napisów (string), skorzystać z list (list); były też błędy.

<https://docs.python.org/3/tutorial/>  
<https://www.learnpython.org/pl/>  
<https://fullstackpython.com>

## Notatka 1. O organizacjach matematycznych na Ziemi.

Redakcja postanowiła zrobić listę towarzystw, kół i związków matematyków. Po przejrzeniu kilku początkowych stron z wynikami wyszukiwania w przeglądarce internetowej Redakcja znalazła stronę internetową z przygotowaną listą takich organizacji. Redakcja wyświetliła strony Towarzystwa Matematycznego Republiki Chińskiej oraz Japońskiego Towarzystwa Matematycznego:

<https://www.cimpa.info/en/node/64#submenu-1>  
<https://www.cms.org.cn/>  
<https://www.mathsoc.jp/>

## Kurs programowania poprzez język Java.

### Sprawozdanie 1.

Podobnie do kursu języka Python (.2023) w tym czasopiśmie Redakcja przejrzała wyniki z pierwszej strony wyszukiwarki internetowej dla hasła "Java tutorial" i pomijając te, które nie miały czytelnych spisów treści, wybrała dwa kursy: pierwszy - na stronie oracle.com - i drugi - na stronie geeksforgeeks.org . Kursy zamieszczone na stronach w3schools.com, tutorialspoint.com i javatpoint.com są uproszczone: dla osób, które programują, a chcą przejrzeć strukturę języka lub przypomnieć fragment kodu do użycia. Kurs na stronie learn.microsoft.com ma spis treści, ale jest to plan na kilka lat lub miesięcy, ścieżka zawierająca linki do wysoko płatnych treści, co nie oznacza że nie warto z nich korzystać, ponieważ według Redakcji - nie da się nauczyć programowania bez papierowej książki w ręku.

W historii internetowej Redakcji jest dużo linków do stron dotyczących Javy - Redakcja przejrzy tę historię do końca teraźniejszego miesiąca.

Kurs umieszczony na stronie oracle.com Redakcja czytała już kilkakrotnie (bez tworzenia graficznego interfejsu użytkownika (GUI - Graphical User Interface) i niektórych treści zaawansowanych). Kurs ten był napisany dla wersji ósmej języka Java, ale w przyjemnej swojej formie jest łatwo przyswajalny.

Kurs w tym czasopiśmie obejmuje pierwsze pięć punktów z rozdziału "Trails Covering the Basics" raczej z ich uproszczeniem do wykorzystania w środowisku do wizualizacji artystycznej Processing. Podrozdziały "Deployment" i o egzaminie są dla specjalistów - korzystając z zaawansowanego środowiska programistycznego (IDE: Integrated Development Environment) jak n.p. Eclipse prawie nie widzi się plików z rozszerzeniem .class czy .jar . Programista wprowadza program i naciśnięciem przycisku w skonfigurowanym środowisku programistycznym uruchamia testowanie i wysyła go do miejsc, w których będzie uruchomiony.

Artykuł o znaczeniu na pewno nie tylko historycznym - napisany przez twórców języka: <https://www.oracle.com/java/technologies/introduction-to-java.html> - Redakcja przy tym czytaniu pomija. Język Java jest tłumaczony (kompilowany - compile) (?) do plików .class grupowanych w katalogi .jar, które mogą być uruchamiane na platformach (platform), na których znajduje się maszyna wirtualna Javy (JVM) (architecture neutral, portable). To jest opisane w pierwszym rozdziale "Overview of Java" w kursie na stronie geeksforgeeks.org . JVM jest maszyną-robotem, który wykonuje kod bajtowy (byte code) (WORA: write once run anywhere), który powstaje automatycznie w wyniku kompilacji programu w języku Java; i nie tylko: prawdopodobnie dowolny język może być kompilowany do kodu bajtowego (również język Python). Tworzeniem maszyn wirtualnych Javy działających na platformach zajmują się specjaliści. Powodów dla których firma Google zamieniła w środowisku Android Studio język Java na język Kotlin Redakcja nie zna (robust: łatwy przy tworzeniu oprogramowania, czytelny i łatwy do zrozumienia dla ludzi, bardziej zwięzły, część programu jest zarządzana automatycznie: w Javie n.p. garbage collection; high performance) (?).

"Buzzword'y" w informacjach o oporogramowaniu są celami (paradygmatami: cechami języków, narzędzi, instytucji), które osiągnąć próbują twórcy.

<https://www.javatpoint.com/java-tutorial>  
Przykłady IDE: Processing, Microsoft Visual Studio Code, Sublime, Eclipse, JetBrains IntelliJ IDEA, Apache NetBeans, Oracle JDeveloper, Google Android Studio, ... (?)

Język Java nie jest prosty (Simple). Jest zorientowany obiektowo (object oriented), zorientowany wątkowo (thread oriented), rozproszony (distributed oriented). Według Redakcji to jego semantyka i biblioteki. JVM wykonuje program, JRE zawiera JVM i podstawowe biblioteki z algorytmami, JDK zawiera JRE oraz narzędzia do transformacji programów (development tools). W IDE-ach są wbudowane JDK. Przykładami narzędzi mogą być rozszerzenia IDE (extensions; wtyczki - plugins), które pomagają automatycznie testować programy, generować czytelnie wyglądającą ich dokumentację, umieszczać programy w miejscach gdzie będą działać, generować szkielet do wypełniania kodem. Prawdopodobnie każde IDE jest zbudowane modułowo (module: szuflada w skrzynce narzędziowej) i może mieć modyfikowaną funkcjonalność (rozszerzenia - extensions, wtyczki - plugins).

Biblioteka jest podręczną skrzynką narzędziową (toolkit), API jest pomocą samochodową do której trzeba zadzwonić. Różnica pomiędzy biblioteką, API a narzędziem jest niewielka: po jedne trzeba sięgnąć, do innych zadzwonić.

W środowisku Processing niewiele z tych elementów występuje. Wszystko jest uproszczone. Po przejrzeniu przykładów dołączonych do środowiska Processing Redakcja wyświetliła kilka białych kółek na ekranie.

<https://docs.oracle.com/javase/tutorial/>  
<https://www.geeksforgeeks.org/java/>  
<https://learn.microsoft.com/en-us/azure/developer/java/learning-resources/fundamentals>  
<https://www.w3schools.com/java/>  
<https://www.tutorialspoint.com/java/index.htm>

# Kurs analizy matematycznej.

## Szkic 1. ::GMFI-RI::

Szkice kursu analizy matematycznej opartego o książkę G.M.Fichtenholz'a (Григорий Михайлович Фихтенгольц) "Rachunek różniczkowy i całkowy" będące publikowane w tym czasopiśmie obejmą części tomu I z uwzględnieniem części tomu II. Należałoby tom II z uwzględnieniem tomu I. Początkowe rozdziały tomu I są powtórką materiału polskiej szkoły średniej końca XX wieku. Cała książka wydaje się być sumą analizy matematycznej XIX wieku: zebraniem oraz przeorganizowaniem treści listów oraz artykułów matematyków według występujących pojęć i teorii. Czytając Redakcja radzi przeorganizować tworzone notatki według funkcji i wzorów w niej występujących w co najmniej stukartkowym zeszycie formatu B5 koniecznie z twardą okładką. Taa książka nie nadaje się do czytania w tramwaju czy w pociągu. Raczej należy ponudzić się w domu, mieszkaniu, akademiku lub na kocu pod parasolem na łące, a najlepiej przy dużej białej suchościeralnej tablicy: inaczej powtarzanie definicji liczb wymiernych i rzeczywistych, granic oraz ciągłości według Heine'go (Heinrich Eduard Heine) oraz Cauchy'ego (Augustin Louis Cauchy) spowoduje niechęć i przeskakiwanie kroków obliczeń podczas czytania dalszych rozdziałów. Notując - grupując informacje - Redakcja radzi zapisywać odnośniki do tomu, rozdziału, paragrafu i strony, na której znajdują się rozproszone a zbierane w trakcie czytania informacje o funkcjach: stworzyć własny format takich odnośników (n.p. ::GMFII-RVIII-§5-str.70-p.290::).

::GMFI-RI:: to lista tematów wprowadzających liczby rzeczywiste:

liczby rzeczywiste

działania  
aksjomat Archimedes (Archimedes)  
przekroje Dedekinda (Julius Wilhelm Richard Dedekind)  
dobre uporządkowanie  
gęstość  
rozwinięcia na ułamki  
kresy zbiorów  
działania i ich zgodność z porządkiem  
wartość bezwzględna  
pierwiastki i potęgi  
logarytmy  
niewymierność

Czytając można zastanowić się nad aksjomatami spełnianymi przez te liczby oraz ich wartością bezwzględną, szukać innych typów liczb, ich wartości bezwzględnych oraz topologii. To zadanie jest trudne; szkic rozwiązania znajduje się w ::SLA-RXII::.

Według Redakcji GMF to dobry, przekrojowy, okrągły i doszczętny wstęp do podstaw matematycznych (wszystkich ?) teorii (fizycznych, chemicznych, ekonomicznych ?) opartych o pojęcie nieskończenie małej. Książka nie przedstawia uogólnionego pojęcia przestrzeni Banacha (Stefan Banach), co jest i plusem i minusem tej książki. Taki wykład trudno połączyć z obliczeniami wzorów funkcji specjalnych, jest skomplikowany dla osób po raz pierwszy czytających książkę o analizie matematycznej, ale ułatwia zrozumienie ogólnych pojęć. Niestety, najlepszych znanych Redakcji wykładów z teorii przestrzeni Banacha oraz teorii miary Redakcja nie widzi wydrukowanych.

Książka G.M.Fichtenholz'a jest książką kucharską XIX-wiecznej matematyki.

## Kurs algebry.

### Szkic 1. ::SLA-RVII::

Książka Serg'a Lang'a (Serge Lang) "Algebra" jest trudna, ale jest bardzo satysfakcjonująca. Zaskoczeniem jest ilość ich zdefiniowanych typów z wydaje się bardzo prostego pojęcia "grupy" (group): zestawu elementów, które można parami mnożyć. Czym różnią się dodawanie i mnożenie? Czy ta książka jest opisem wszystkich możliwych do wyobrażenia zestawów liczb? Skoro istnieją liczby uproszczone, to czy liczby znane ze szkoły są uproszczeniem? I czy prawdziwa jest hipoteza:

Każda algebra jest komutatywna. (?) (!)

Książka podzielona na trzy rozdziały dotyczące teorii grup, teorii Galois (Évariste Galois), algebry liniowej. W związku z bardzo zwięzłym spisem treści książki Redakcja postanowiła w tym roku, w tym czasopiśmie publikować dokładniejsze spisy treści, fragmenty notatek oraz wskazówki do paragrafów rozdziału drugiego dotyczącego klasyfikacji zestawów liczb, w których mnożenie jest przemienne (commutative): teorii Galois.

Struktura przemiennych pierścieni, ciał i modułów (module) jest przedmiotem algebry przemiennej stosowanym w geometrii algebraicznej (algebraic geometry). (?) (!)

## Rozdział VII. Rozszerzenia algebraiczne.

Rozdział służy wprowadzeniu pojęcia ciała (field): zestawu liczb podlegających przemennym i łącznym działaniom dodawania oraz mnożenia - rozdzielnym względem siebie - wraz z 1 i 0 oraz z możliwością odwracania elementów (... bez dzielenia przez 0). Przedstawiona klasyfikacja takich zestawów polega na wyznaczeniu wszystkich ciał prostych (prime field): albo  $1 + \dots + 1 = 0$  (charakterystyka = 0) albo nie (liczby wymierne znane ze szkoły) oraz ich skończonych rozszerzeń (finite extensions). Ciałami pierwszego typu są  $\{0, 1, 2, \dots, p\}$  dla liczb pierwszych  $p$ . Jeśli  $p$  nie jest liczbą pierwszą, wtedy wynikiem nietrywialnego mnożenia może być 0:  $4 * 5 \bmod 10 = 0$ . Ze względu na tę cykliczność takie zestawy liczb nazwane były pierścieniami (ring). Wiele pojęć matematycznych ma nietrafne nazwy, ponieważ te zostały wprowadzone dla prostszych obiektów przed poznaniem ich uogólnień.

\$1: wielomiany nierozkładalne (irreducible polynomials) i ich pierwiastki (zeros), rozszerzenia ciał i ich automorfizmy, rozszerzenia skończone, algebraiczne oraz skończenie generowane (finitely generated).

Motywnym przewodnim paragrafu jest pojęcie wyróżnionej klasy rozszerzeń w prosty sposób reprezentowalnej na diagramach. Rozszerzenia skończenie generowane nie tworzą wyróżnionej klasy rozszerzeń. Pomimo dużej ilości z premedytacją wstawionych literówek w posiadanych przez Redakcję wydaniach diagramy w książce są bardzo ważne i należy je zrozumieć tak jak są podane: bez uproszczeń przy trudnościach w zrozumieniu, bez wymiany monomorfizmów na izomorfizmy. Bardzo często (!) lematy są ważniejsze od twierdzeń. Elementy przestępne (transcendent) i ich rozszerzenia omówione są w Rozdziale X (::SLA-RX::).

Po przeczytaniu rozdziału Czytelnik rozumie dorzucanie nieistniejących pierwiastków wielomianu (części lub wszystkich), zależność pomiędzy algebraicznością (algebraic) liczby a skończonością jej rozszerzenia, stopniami wielomianu minimalnego i rozszerzenia ciała, tworzenie piętrowych rozszerzeń skończonych.

§2: przy pierwszym czytaniu, być może, należy pominąć czytanie treści dowodów, a czytając rozróżnić pojęcia algebraicznego domknięcia ciała oraz ciała algebraicznie domkniętego. Rozdział ten może sprawiać trudności związane z podstawami matematyki oraz aksjomatyką teorii zbiorów.

§3: rozszerzenia normalne: ich piętra działają zupełnie niezależnie pod względem grup automorfizmów ciał (trochę jak te okrągłe zamki szyfrowe w sejfach?). Nie jest to klasa wyróżniona. Powstają przez dorzucenie wszystkich pierwiastków zestawu wielomianów.

§4, §7: opisują dwie klasy wyróżnione rozszerzeń: rozdzielcze (seperable), i radykalne (z inną czasem używaną nazwą czysto nierozdzielczych) (radical, pure inseperable). Rozszerzenia nierozdzielcze mogą zawierać elementy rozdzielcze, ale również, nierozdzielcze. Rozszerzenia radykalne nie zawierają elementów rozdzielczych. Rozszerzenie skończone jest rozdzielcze w.i.t.w., gdy wszystkie jego elementy są rozdzielcze, w.i.t.w., gdy jego stopień nierozdzielczy jest równy 1. Stopień rozdzielczy może być podzielny przez charakterystykę ciała  $p$ . Stopień rozszerzenia, stopień rozdzielczy rozszerzenia i stopień nierozdzielczy rozszerzenia są multiplikatywne. Wystarczy udowodnić dwa.

§5: o ciałach skończonych w następnym szkicu.

§6: twierdzenie o elemencie prymitywnym. (!!)

Numer:	<b>MATINF 0/2023</b>		
Data wydania:	31 lipca 2023	Druk:	3
Adresy Redakcji:	<a href="mailto:matinf@poczta.onet.pl">matinf@poczta.onet.pl</a>	Witryny informacyjne:	<a href="https://github.com/czasopismo-MATINF/czasopismo-MATINF">https://github.com/czasopismo-MATINF/czasopismo-MATINF</a>
Czytelnik:	osoba samodzielnie ucząca się, student		
Cel:	systematyczne kursy podstaw programowania, matematyki, algebry i analizy matematycznej, artykuły o takowej tematyce; recenzje książek do nauki, narzędzi do programowania, listy zasobów informatycznych, serwisów internetowych; po dwudziestu latach wybuchu programistycznego niepotrzebna już promocja programowania, ale powrót na jego podstawowy poziom i znalezienie miejsca dla jego podstaw przy potwornie zwiększającym się stopniu skomplikowania algorytmów oraz narzędzi programistycznych; systematyczny zbiór powszechnie znanych pomysłów i idei;		

## Wstęp do bieżącego numeru:

W numerze wrześniowym czasopisma rozpoczną ukazywać się pierwsze części kursów programowania w językach Python i Java. Kursy te będą zawierały opisy instrukcji języków oraz obsługi dostępnych narzędzi do programowania w tych językach i zgodnie z przesłaniem MATINF: będą odpowiednie - dla osób, które samodzielnie się uczą - jako systematyczna pomoc w organizacji nauki i nabywania praktycznych umiejętności. Kurs języka Python będzie odpowiedni dla osób bez znajomości programowania, ponieważ oddelegowany pracownik redakcji też będzie uczył się tego języka, a z kursów korzystając dla powtórzenia wiadomości i sprawdzenia swoich umiejętności. Również po letnich wakacjach zaczną się pojawiać szkice z kursów analizy matematycznej i algebry z wykorzystaniem polskich, i polskich tłumaczeń znanych i popularnych podręczników.

## Spis aktualnie rozwijanej zawartości:

kurs programowania w języku Java	narzędzia programistyczne	kurs programowania w języku Python	szkice kursu z analizy matematycznej	historia matematyki przełomu XVIII i XIX wieku
listy serwisów internetowych	recenzje książek	algorytmika	szkice kursu z algebry	indeks wiadomości technicznych