O2 Raport: Metodyka integracji metod komputerowych z nauczaniem przedmiotów ścisłych

Praca zbiorowa pod redakcją
dr hab. Marcina Kostura

Spis treści

ROZDZIAŁ 1

Wstęp

Książka ta jest dostępna w interaktywnej wersji online pod adresem:

- W wersji polskiej http://visual.icse.us.edu.pl/metodologia
- W wersji angielskiej http://visual.icse.us.edu.pl/methodology

Podczas realizacji projektu nauczyciele i uczniowie wytworzyli kilka tysięcy dokumentów - tzw. notatników w systemie SageMath. Dokumenty te znajdują się na dwóch dedykowanych szkołom serwerach i niektóre z nich zostały opublikowane pod adresami:

- https://sage01.icse.us.edu.pl/pub/
- https://sage03.icse.us.edu.pl/pub/

Rezultaty projektu zostały też podsumowane w filmie z realizacji projektu dostępnym na Youtube pod adresem:

https://www.youtube.com/watch?v=BAUCbMXWceI

1.1 O książce

Współczesne formy wykorzystania w pracy nauczyciela przedmiotów matematyczno-przyrodniczych elementów technologii informacyjnej i komunikacyjnej nie stanowi tylko prawnego obowiązku, ale przede wszystkim ma za zadanie urozmaicać lekcję i stanowić w jej odbiorze użyteczne narzędzie oraz element korygujący odbiór toku logicznego zajęć. Nowoczesny nauczyciel nie może wyobrażać sobie pracy, bez użycia współczesnych, co warte podkreślenia, aktualnie kształconym pokoleniom, technik informatycznych. Na bieżąco musi doskonalić swoje umiejętności obsługi programów komputerowych i języków programowania. Choć wiele realizowanych tematów będzie mu znanych, ponieważ w trakcie studiów magisterskich lub doktoranckich realizuje się zajęcia z języków programowania użytkowanych w chemii i fizyce, to jednak nabycie nowych umiejętności aplikacji komend językowych do rozwiązywania problemów natury rachunkowej, niedostępnych uczniom realizujących polską podstawę programową matematyki, nawet w zakresie rozszerzonym pozostaje wciąż w jego gestii. W codziennej, szkolnej pracy, nauczyciel wykorzystuje technologię komputerową głównie do tworzenia różnorodnych testów użytkowych, diagnoz, narzędzi badawczych i ankiet. Dzięki specjalistycznemu oprogramowaniu komercyjnemu, tworzy zaproszenia, plakaty, informatory szkolne oraz kreuje scenariusze lekcji lub apeli. Dzięki technologii komputerowej dokumentacja działań nauczyciela prowadzona jest przejrzyście, estetycznie i czytelnie. Własny warsztat pracy jest uporządkowany i na bieżąco udoskonalany, również dzięki zastosowaniu aplikacji i programów dziennika elektronicznego lub oprogramowania do tworzenia świadectw szkolnych. Wszystkie te programy i aplikacje charakteryzuje stosunkowo intuicyjna i oparta o proste relacje metoda ich konstrukcji, a co za tym brak możliwości zmian i wykonywania w nich innych niż umożliwione przez producenta operacji. Zarówno z punktu widzenia nauczyciela, jak i ucznia, będącego podmiotem procesu nauczania nie bez znaczenia pozostaje możliwość rozwiązywania problemów określonych w podstawie programowej oraz wykraczających poza jej ramy, w oparciu o szerokie spektrum metod znajdujących zastosowanie zarówno z punktu widzenia uczącego, jak i nauczanego. Nauczyciel podejmujący analizę zagadnień obecne oraz powstającej na skutek planowanej reformy systemu oświaty podstawy programowej przedmiotów takich jak fizyka i chemia napotyka na wiele treści, których zrozumienie wymaga zastosowania złożonego rachunku z elementami analizy matematycznej. Do popularnych dziś metod kształcenia należy podawanie bez udowadniania zależności opisujących dane zjawisko lub przedstawianie wykresów omawianych zależności bez prezentacji równań opisujących przestawiane krzywe. Dla ucznia oczekującego pełnego wyjaśnienia danego zjawiska i jego interpretacji, stanowi to jedno z ważniejszych utrudnień w przyswojeniu i w powiązaniu rozważań teoretycznych z interpretacją matematyczną. Większość zastosowań technologii informacyjnej w dydaktyce sprowadza się do pobierania danych lub ich wprowadzania i zautomatyzowanej obróbki, co stanowi skuteczne więzy dla rozwoju percepcji współczesnego ucznia. Środowisko języka programowania Python i Sage jest uwolnieniem podmiotu nauczania od narzuconych mu więzów, zaś umiejętności programowania nabywane przez ucznia w trakcie procesu nauczania skutecznie rozwijają jego zdolności konkludowania i ewaluowania w trakcie powyższego procesu. Zastosowanie proponowanego środowiska programowania pozwala sprowadzać złożone zależności i równania z poziomu wiedzy akademickiej, do poziomu niewiele wykraczającego ponad obowiązujące podstawy programowe nauczania przedmiotów ścisłych i jest w pełni aprobowane i zalecane jako narzędzie pracy, zarówno przez Ministerstwo Edukacji Narodowej, jak i Ministerstwo Cyfryzacji.

Niniejsza metodologia jest wynikiem działań, związanych z projektem “icse4school - Zintegrowane nauczanie z perspektywą obliczeniową" realizowanego w latach 2014-2017. realizowany jest w ramach Programu Erasmus+, akcja 2 Partnerstwa strategiczne. Projekt zrealizowano w międzynarodowej współpracy międzysektorowej, obejmującej instytucje szkolnictwa wyższego (Uniwersytet Śląski w Katowicach, Uniwersytet w Augsburgu, Simula School of Research and Innovation), szkoły ponadgimnazjalne ((2 AZSO w Chorzowie oraz 117 ZS - XXXIII LO w Warszawie) oraz organizację pozarządową (Fundacja na rzecz rozwoju Śląskiego Międzyuczelnianego Centrum Edukacji i Badań Interdyscyplinarnych). Celem głównym było opracowanie w ponadnarodowym konsorcjum metodyki zintegrowanego z informatyką nauczania matematyki i fizyki w szkołach ponadgimnazjalnych na podstawie realizacji pilotażowego programu zajęć w wybranych szkołach.

Wypracowane w projekcie icse4school rezultaty pracy intelektualnej stanowią pomoc w realizacji podstawy programowej narodowej jak i przypisanej dla uczestników Programu Dyplomowego Matury Międzynarodowej. Jakościowo, stanowią rozszerzenie bazowej wiedzy ucznia, dostosowując zdolności rachunkowe uczestnika do wymaganego podczas udziału w konkursach i olimpiadach wykraczających poza obie podstawy programowe. Nakierowują uczestnika na innowacyjne metody rozwiązywania klasycznych problemów i zagadnień naukowych z dziedziny matematyki, informatyki i fizyki. W przypadku uczniów realizujących zajęcia w szkolnych laboratoriach badawczych z dziedziny fizyki lub chemii stanowią ułatwienie podczas procesu opracowywania danych laboratoryjnych, zgromadzonych podczas zajęć. Obrazują mniej rozpowszechnione metody takie jak metody numeryczne czy rozwiązywanie skomplikowanych równań różniczkowych, które nie są objęta obiema podstawami programowymi. Innowacyjność projektu polega na wskazaniu zastosowania powyższych metod do rozwiązywania klasycznych problemów nie wyłączając z nich zadań egzaminacyjnych. Istotą rezultatów intelektualnych winien być wzrost poziomu wynikowego egzaminów realizowanych przez uczestników oraz większa pula osiąganych tytułów finalistów i laureatów konkursów przedmiotowych z zakresu nauk przyrodniczych i ścisłych, wśród czynnych odbiorców projektu.

Przedmiotem projektu była nauka programowania w języku Python z użyciem środowiska Sage oraz wykorzystanie tej umiejętności do rozwiązywania problemów, dotyczących przedmiotów: matematyka, fizyka, informatyka. Projekt adresowany był do uczniów szkół ponadgimnazjalnych. Potrzeba realizacji projektu ICSE4school wynikała z rosnącego zainteresowania zagadnieniem programowania oraz zebranej wiedzy pokazującej brak informacji na temat jednego z najczęściej na świecie używanych języków programowania Python. Ponadto obserwacje nauczycieli - zarówno akademickich, jak i prowadzących klasy o rozszerzonym programie nauczania matematyki, informatyki i fizyki w szkołach ponadgimnazjalnych wskazywały na bezradność uczniów i studentów w stosowaniu technologii informacyjnej i metod numerycznych w trakcie rozwiązywania problemów z pogranicza zagadnień naukowych.

Chcąc wykorzystać nasze materiały oraz nie mając podstawowych umiejętności w zakresie obsługi języka Python, proponujemy zacząć od zapoznania się tym językiem oraz ze środowiskiem SageMath dostępnym pod adresem SageMath.org. W środowisku Sage uczeń ma szansę, prócz programowania, wykonywania ćwiczeń na przykładowych materiałach tam zamieszczonych z możliwością modyfikowania tekstów źródłowych.

Opracowane w projekcie scenariusze lekcji z wykorzystaniem języka Python rekomendujemy do stosowania w różnoraki sposób, w zależności od potrzeb i możliwości uczniów oraz nauczycieli. Tak więc mogą one być wykorzystywane bezpośrednio podczas zajęć lekcyjnych lub zajęć dodatkowych w formie warsztatów, pod warunkiem odpowiedniej liczby godzin dydaktycznych do dyspozycji nauczyciela. Materiały lub ich fragmenty mogą być także wykorzystane w ramach pracy domowej przy zastosowaniu nowatorskiej metody „odwróconego nauczania" („odwróconej klasy", „flipped teaching") polegającej na tym, że uczniowie w ramach zadania domowego, analizując wskazany materiał, zgłębiają podstawową wiedzę na temat nowego zagadnienia, a czas lekcyjny przeznacza się na utrwalanie oraz pogłębianie zdobytej wiedzy i umiejętności (w przeciwieństwie do klasycznych metod, według których podczas lekcji wprowadza się podstawową wiedzę, a utrwalanie i dostateczne pogłębianie materiału uczeń wykonuje samodzielnie w domu). Wreszcie materiały nasze mogą mieć zastosowanie do zgłębiania i poszerzania wiedzy oraz umiejętności przez uczniów i studentów o specjalnych potrzebach edukacyjnych (uczniów i studentów uzdolnionych, posiadających potrzebę samorozwoju lub mających potrzebę zastosowania metod numerycznych podczas rozwiązywania problemów z użyciem ogólnodostępnego narzędzia), także przez uczniów realizujących obowiązek nauki poza szkołą do samodzielnej nauki, jak również przez wszystkich zainteresowanych powtórzeniem, utrwaleniem i poszerzeniem wiedzy.

Scenariusze z projektu iCSE4school mogą być także wykorzystywane w pracy z uczniami o mniejszym potencjale w zakresie przedmiotów ścisłych poprzez zaprezentowanie przez nauczyciela fragmentów naszych materiałów, jak również w postaci pracy warsztatowej, podczas których uczniowie samodzielnie modyfikują fragmenty tekstów źródłowych, obserwując efekty tych zmian i formułując stosowne wnioski. Taki sposób wykorzystania materiałów daje nie tylko możliwość rozwoju nauczyciela i młodzieży, ale także efektywnego wykorzystania scenariuszy zajęć z wykorzystaniem języka Python w pracy z uczniami młodszymi na wcześniejszych etapach edukacyjnych.

Materiały z projektu iCSE4 mogą być wykorzystywane w formie e-learningu[1], blended-learningu[2], jak i m-learningu[3], z wykorzystaniem komputerów lub innych urządzeń (np. smartfonów, tabletów); przy dostępie do internetu lub off-line.

Obserwacja działań uczniów polegających na tworzeniu prostych programów z użyciem języka Python w celu rozwiązania danego problemu, podczas realizacji projektu, potwierdziła duże możliwości indywidualizacji ich pracy. Indywidualnie dobierany poziom wymagań, dostosowane do indywidualnych potrzeb tempo pracy oraz indywidualna pomoc nauczyciela wpływają korzystnie na rozwój każdego z uczniów. Ważnym czynnikiem mobilizującym uczniów do działań i nauki jest także obserwowane ostatnio zapotrzebowanie na naukę programowania (kodowania) oraz świadomość znacznego braku ludzi potrafiących programować na rynku pracy. Zachęcając do wykorzystania opracowanych w projekcie ICSE4 materiałów dydaktycznych, zapraszamy do zgłaszania uwag.

W projekcie oraz tworzeniu niniejszej metodologii wzięli udział przedstawiciele następujących instytucji:

- Nauczyciele: Jolanta Drogoń, Łukasz Głaz, Krzysztof Jarczewski, Mirosław Malinowski, Justyna Matejczyk, Adam Ogaza, Krzysztof Oleś, Katarzyna Sikora, Hanna Stachera, Mariola Strojny,
- Jonas van den Brink, Vigdis Holta, Marie Roald, Freyja Jørgensen, z Simula School of Research and Innovation, Oslo
- Manuel Milling, Severin Wünsch i profesor Gert Ingold z Uniwersytetu w Augsburgu

[1] https://pl.wikipedia.org/wiki/E-learning
[2] https://pl.wikipedia.org/wiki/Blended_learning
[3] https://pl.wikipedia.org/wiki/M-learning

- Marcin Kostur, Uniwersytet Śląski
- Magdalena Hampel, Joanna Klekowska i Marta Margiel z Fundacji EduRes

Autorzy!

1.2 Szkoła w cyfrowej epoce

We współczesnej szkole coraz mniejszą rolę zdaje się odgrywać autorytet oparty na przymusie, nakłanianiu czy uprawnieniach nauczyciela. Co ciekawe, zmienia się też zakres oraz sposób oceniania kompetencji (wiedzy i umiejętności) nauczyciela. Podczas gdy dawniej nauczyciel w klasie w zasadzie stanowił jedyne i najważniejsze źródło wiedzy, obecnie, dzięki powszechnemu dostępowi do Internetu, jego słowa mogą być natychmiastowo przez uczniów zweryfikowane, a często podważone. Zadaniem nauczyciela współczesnej młodzieży jest zatem budowanie autorytetu na umiejętnościach innych niż te związane ściśle i jedynie z posiadaniem wiedzy. Obecnie nauczyciel powinien również wykazywać się otwartością i kompetencjami w zakresie używania nowych technologii do zdobywania i weryfikowania wiedzy. Powszechny dostęp do Internetu sprawił, że to nie sama wiedza nauczyciela pozwala budować autorytet, lecz elastyczność i umiejętność dostosowania się do aktualnych potrzeb i zainteresowań ucznia, w tym także poprzez wspólne zgłębianie tematów i zagadnień, gdy nierzadko sami uczniowie mogą stanowić inspirację dla nauczycieli. Jednym z najważniejszych zadań nauczyciela w cyfrowej rzeczywistości jest pokazanie i nauczenie swoich uczniów jak z tej rzeczywistości korzystać; jak dbać o bezpieczeństwo i prywatność w Internecie, jak weryfikować znalezione tam informacje, jakie strategie i jakie narzędzia stosować do organizacji wiadomości napotkanych w Internecie oraz w jaki sposób uzupełniać wiedzę zdobytą w szkolę wiadomościami z cyfrowych źródeł.

Już na początku lat 90 ubiegłego wieku zmiana w postrzeganiu roli nauczyciela opisana powyżej została zidentyfikowana w artykule Alison King pod znamiennym tytułem 'From Sage on the Stage to Guide on the Side' (King 1993). Artykuł ten odnosił się do stylu prowadzenia zajęć w uniwersytetach amerykańskich, a jego celem było zachęcić wykładowców do zrezygnowania z funkcji podającego wiedzę 'mędrca na katedrze' na rzecz 'przewodnika stojącego z boku', który towarzyszy studentom w zdobywaniu wiedzy. Taka zmiana roli nauczyciela czy wykładowcy wydaje się doskonale współgrać ze współczesnymi potrzebami pokolenia Y, które wysoko ceni konkretne umiejętności praktyczne oraz możliwość uczenia się tego, co ich w danej chwili interesuje lub tego, co postrzegają jako im potrzebne, zamiast tego, co wynika z programu nauczania, decyzji nauczyciela czy przewidywań na temat tego, co być może przyda im się w przyszłości. Umieją oni także i lubią pracować w grupach czy zespołach i wykorzystują do tego nowe technologie, które pozwalają im być w kontakcie ze sobą poza godzinami spędzonymi w szkole.

Ponieważ uczniowie z pokolenia Y są określani jako cyfrowi tubylcy (digital natives; Prensky 2001), czyli osoby, które czują się w Internecie jak u siebie w domu i nie znają świata, gdzie nie było do niego dostępu, to dużo łatwiej jest do nich dotrzeć i ich zmotywować do zdobywania wiedzy przy pomocy nowych technologii. Dużo łatwiej jest też nauczycielowi (często będącego jednak cyfrowym imigrantem, jak określiłby to Mark Prensky) budować własny autorytet, jeśli umie z jednej strony docenić sprawność uczniów w posługiwaniu się nowymi technologiami, a z drugiej gdy potrafi wskazać im sposoby zdobywania, weryfikowania i organizowania wiedzy zyskanej z cyfrowych źródeł. By tak się stało, musi on zrezygnować ze swojej roli nieomylnego autorytetu na rzecz autorytetu osobistego, często opartego na tzw. umiejętnościach miękkich.

Dla kategorii zapamiętywanie są to na przykład tworzenie fiszek online, tworzenie notatek w formie cyfrowej (przy użyciu edytorów tekstu jak i ich odpowiedników online), współtworzenie tekstu przez kilku lub kilkunastu uczniów itp. Dla wykazania zrozumienia materiału uczeń może prowadzić blog z podsumowaniem materiału z lekcji czy też pisać komentarze do wypowiedzi nauczyciela i innych uczniów na forum. Zastosowanie przejawia się w umiejętności edytowania zawartości sieciowej, korzystania z programów komputerowych i webowych itp. Analizowanie to na przykład przypisywanie tagów i kategoryzowanie zawartości cyfrowej. Ocenianie to umiejętność weryfikacji i testowania wiedzy i programów, a tworzenie to programowanie, tworzenie filmów wideo, animacji, podcastów oraz ich publikacja w sieci. Do każdego z tych stadiów można przypisać konkretne narzędzia cyfrowe, które pozwolą nauczycielowi i uczniom współpracować w środowisku cyfrowym. Dobrze przedstawia to następująca grafika:

Musimy jednak pamiętać, że aplikacje internetowe i programy zmieniają się i wychodzą z użycia bardzo szybko, więc nauczyciel, który chce wspierać swoich uczniów w pożytecznym wykorzystywaniu nowych technologii musi trzymać rękę na pulsie i cały czas dokształcać się formalnie i nieformalnie w tym zakresie.

Pokolenie Y i jego potrzeby

Obecna młodzież to tzw. pokolenie Y (Generation Y), noszące też inną nazwę: milenialsi (Millennials). Według Wikipedii jest to pokolenie ludzi urodzonych w Polsce od 1984 roku do 1997 roku, a w innych krajach np. w USA pokolenie wyżu demograficznego z lat 80. i 90. XX wieku. Nazywane jest ono również „pokoleniem Milenium”, „następną generacją”, „pokoleniem cyfrowym” oraz „pokoleniem klapek i iPodów”. W odróżnieniu od poprzedniej generacji, określanej mianem generacji X, „oswoili” oni nowinki technologiczne i aktywnie korzystają z mediów i technologii cyfrowych. Milenialsi uznawani są za generację zuchwałą, otwartą na nowe wyzwania, co wywiera piętno na sposób poznawania przez nich świata i uczenia się. Niektóre cechy charakterystyczne przedstawicieli pokolenia Y to:

- aktywnie i w każdej dziedzinie życia korzystają z technologii i mediów cyfrowych;
- żyją w „globalnej wiosce” dzięki dostępowi do Internetu mają znajomości na całym świecie;
- cechuje ich duża pewność siebie;
- są dobrze wykształceni i gotowi dalej się rozwijać;
- według badań przeprowadzonych w University of New Hampshire, cechuje ich wysokie mniemanie o swoich umiejętnościach, przekonanie o własnej wyjątkowości, nadmierne oczekiwania oraz silna awersja wobec krytyki.

Zmiana cech pokolenia uczącego się, jak również zmiany w następujących zakresach:

- technologia (pojawiły się nowe urządzenia),
- pedagogika (nauka stała się bardziej indywidualna),
- treść (podawana jest krótsza treść i zmieniły się jej nośniki),

wygenerowały konieczność poszukiwania nowych metod i form nauczania. Wśród wykorzystywanych współcześnie i efektywnych ze względu na dopasowanie do potrzeb pokolenia Y metod nauczania można wymienić e-learning, blended learning, m-learning, Flipped Teaching i metoda projektu (w tym WebQuestu). Są to metody oparte na obserwacji i działaniu, a zatem w dużym stopniu skuteczne w zakresie rozumienia i zapamiętywania nowych wiadomości.

ROZDZIAŁ 2

Przedmioty ścisłe z SageMath

2.1 Wybór narzędzia

2.1.1 Innowacyjność metodologii

Innowacyjność projektu wpisuje się w ogólnoświatowe trendy dydaktyczne. W ostatnich latach zostało przeprowadzonych wiele projektów komputeryzujących nauczanie przedmiotów ścisłych. Warto nadmienić np. Compadre, Open Source Physics, finansowane przez NSF w USA.

Komunikat dyrektora Centralnej Komisji Egzaminacyjnej z 9 września 2016 r. w sprawie listy systemów operacyjnych, programów użytkowych oraz języków programowania w przypadku egzaminu maturalnego z informatyki w 2017 roku zawiera zapis: „Od roku szkolnego 2017/2018 na maturze z informatyki nie będzie już możliwości wyboru języka programowania Pascal, natomiast od roku 2018/2019 będzie można wybrać język programowania Python". Poszerzenie zakresu wiedzy i umiejętności uczniów w trakcie realizacji projektu iCSE4school o naukę programowania w języku Python wyprzedziło ustalenia dyrektora Centralnej Komisji Egzaminacyjnej. Wyprzedziło także zmiany w nauczaniu informatyki na poziomie rozszerzonym w szkołach kończących się maturą w Polsce. Więcej czasu poświęconego na naukę programowania oraz dodatkowe działania uczniów podczas rozwiązywania problemów matematycznych, fizycznych i informatycznych z użyciem języka Python spowodowały pełniejszą interioryzację wiedzy, pogłębiły umiejętności związane z posługiwaniem się językiem Python i przygotowały uczniów do efektywnego zmierzenia się z wymaganiami egzaminu maturalnego z informatyki na poziomie rozszerzonym od 2018/2019 roku. Miały one również wpływ na wcześniejsze podjęcie kroków przez nauczycieli zaangażowanych w projekt, mających na celu własne doskonalenie zawodowe w zakresie znajomości języka Python.

Realizacji projektu towarzyszyło zastosowanie jednej ze współcześnie uznawanych teorii uczenia się, poznawania i zdobywania wiedzy jaką jest konstruktywizm pedagogiczny. Filozofia konstruktywizmu osadzona w poznawczej koncepcji człowieka zakłada, że uczeń jest samodzielnym i aktywnym podmiotem, który korzystając z różnych źródeł informacji konstruuje swój własny system wiedzy i swoją osobowość. Konstruktywizm uważa, że zdobywanie wiedzy to proces, który odbywa się w interakcji ze środowiskiem edukacyjnym, że nauczyciel nie może przekazać uczniom pojęć poprzez samo ich objaśnianie, że nauczanie nie polega na tym, iż nauczyciel podaje gotową wiedzę, a uczeń ma się tego nauczyć, zapamiętać i potem odtwarzać, tylko w możliwie dużym stopniu wiedza powinna być odkrywana przez ucznia. Następuje odrzucenie werbalnego przekazywania wiedzy przez nauczyciela (między innymi wykładu) jako metody nieskutecznej i niezgodnej z naturą przyswajania wiedzy. Według konstruktywistów wiedza to nie tylko fakty i informacje, to także umiejętność wykorzystania ich w praktyce. Uczący się sami tworzą wiedzę, która w nich rezyduje i dlatego wiedza osobista ma charakter tak indywidualny, jak różne są osobowości uczących się. Wśród głównych założeń konstruktywizmu istotna jest zmiana relacji nauczyciel-uczeń. Nauczyciel przestaje być źródłem wiedzy i „mentorem”, bliżej mu do trenera, tutora czy „coacha”. Nauczyciel przygotowuje dla ucznia sytuację dydaktyczną, zadanie, problem, które musi wymagać twórczego i samodzielnego myślenia. Ingerowanie nauczyciela w proces rozwiązywania zadania sprowadza się do wskazówek i jest ograniczone do minimum. Podejście konstruktywistyczne należy do nurtu pedagogiki, który kładzie nacisk na aktywność osoby uczącej się, co miało zastosowanie podczas realizacji zadań w projekcie iCSE4school.

2.1.2 Systemy komputerowe dla nauczania przedmiotów ścisłych

Od wielu lat czynione są próby stosowania narzędzi informatycznych i nawet programowania na lekcjach przedmiotów ścisłych. Najczęściej zostaje wybierane przez fachowców w jednej dziedzinie specjalistyczne oprogramowanie. Bywa, że wybór jest lobbowany przez producenta danego systemu. W efekcie prowadzi to do nieskoordynowanych działań o ograniczonych do poszczególnych przedmiotów. Uczeń poznaje na lekcjach informatyki narzędzia i języki nie przydatne podczas innych przedmiotów. Lekcje fizyki i matematyki są wzbogacane dedykowanym oprogramowaniem, które nie jest stosowane na informatyce. Taki sposób działania nie jest zły sam w sobie - używamy właściwych narzędzi do poszczególnych zadań. Co jeśli istnieje wspólne narzędzie i język w których praktycznie bezkompromisowo można zastosować do całego spektrum zagadnień w szkolnej (i nie tylko) edukacji?

Zastanówmy się, jakie cechy powinien mieć system komputerowy, by przełamać powyższy stereotyp? Poszukajmy rozwiązania jednocześnie spełniającego następujące cechy:

1. WIDE: System powinien być oparty na popularnym i otwartym języku programowania szerokiego przeznaczenia.

   Języki programowania szerokiego przeznaczenia mogą być wykorzystane go tworzenia gier komputerowych, jak i aplikacji naukowych czy edukacji. Z drugiej strony istnieje wiele tak zwanych języków dziedzinowych stworzonych na potrzeby pojedyńczej aplikacji. Takie języki znakomicie spełniają swoją rolę, jednak z reguły nie nadają się do innych zadań. Przykładem może być język Matlab, który pomimo swojej popularności nie jest rozwiązaniem przyjętym w nauczaniu informatyki. Języki takie jak Python umożliwiają praktycznie wykonanie wszystkich zadań które są właściwe dla Matlab, jednak ich specyfika pozwala na zostosowanie ich m.in. do nauczania informatyki. Ważną cechą systemu jest uniknięcie uzależnienia od dostawcy, to ma często miejsce w przypadku stosowania języków dziedzinowych.

2. INTERACT: Język programowania powinien umożliwiać pracę interaktywną.

   Takie żądanie praktycznie eliminuje języki kompilowane (C/C++). Chcąc wykorzystać system komputerowy interaktywnie, najstosowniejsze wydają się języki z dynamicznym typowaniem i mechanizmami typu introspekcja. Wymaganie to spełnia więszkość języków dziedzinowych dostarczanych przez producentów systemów klasy CAS, ale również języki ogólnego przeznaczenia takie jak Python.

3. FREE - System powinien być powszechnie dostępny.

   Nieskrępowana dostępność do systemu jest najlepiej zagrawantowana przez oprogramowanie otwarte. Dodatkowo oprogramowanie takie daję możliwość wglądu w każdy zastosowany algorytm co ma znaczenie zarówno w nauce, jak i posiada walory edukacyjne. Dostępność jest również związana z technicznymi aspektami związanymi z instalacja oprogramowania. Możliwość skorzystania z pracy w „chmurze” z pośrednictwem jedynie przeglądarki internetowej jest bardzo pożądaną cechą takiego systemu.

4. POWER - Możliwości systemu powinny bezkompromisowo zawierać wszystkie elementy niezbędne do zastosowania go na wszystkich przedmiotach ścisłych.

   Takie wymaganie eliminuje języki, które nie są na tyle rozpowszechnione, by były w nich zaimplementowane wszystkie najważniejsze metody obliczeniowe lub wizualizacyjne. Python jest szczególnie interesującym przykładem, ponieważ istnieje w nim powszechnie wykorzystywana łatwość do tworzenia interfaceów przeróżnych bibliotek napisanych w innych językach. Z tej cechy korzysta system SageMath, który zawiera w sobie setki bibliotek naukowych połączonych wspólnym sposobem użycia z pomocą właśnie Python-a.

5. PROF - System powinien umożliwiać płynne przejście od pracy na lekcjach w szkole do profesjonalnych zastosowań w badaniach naukowych czy w przemyśle.

   Nie ma żadnego powodu, by w szkole uczyć na „małym” systemie, a na studiach czy w pracy poznawać dopiero ten „duży”. Z powodzeniem można użyć nawet w szkole podstawowej tego samego języka i systemu, który jest używany przez naukowców oczywiście ograniczając się do wykorzystania jego niewielkiej części. Oszczędza to dużo czasu i wyrabia od razu dobre nawyki od najwcześniejszego okresu nauki. Należy podkreślić, że często koszty licencji oprogramowania dla systemów stosowanych profesjonalnie są znacznie wyższe od dedykowanych systemów edukacyjnych. Problem ten nie istnieje, gdy wybierze się oprogramowanie otwarte.

System	WIDE	INTERACT	FREE	POWER	PROF
SageMath	TAK	TAK	TAK	TAK	TAK
Python/Scipy	TAK	TAK	TAK	TAK	TAK
Mathematica	TAK	TAK	NIE	TAK	TAK
C/C++	TAK	NIE	TAK	TAK/NIE	TAK
Geogebra	NIE	TAK	TAK	NIE	NIE
Java	TAK	NIE	TAK	TAK/NIE	TAK

Z powyższej analizy wynika, że rozwiązania oparte o język Python spełniają wszystkie wymagania. Co więcej, Python jest językiem o rosnącym znaczeniu w braży informatycznej. Zarówno stosowanie standardowego interpretera Python, jak i systemu algebry komputerowej

SageMath może dać takie same efekty. Zdecydowanie na lekcji matematyki czy fizyki system SageMath będzie - jako system algebry komputerowej - oferował krótszą drogę do rozwiązania. Zanim jednak omówimy te systemy odpowiedzmy sobie na pytanie co to jest system algebry komputerowej?

Czym jest system algebry komputerowej?

Pod pojęciem system algebry komputerowej (ang. Computer Algebra System lub CAS) rozumie się program komputerowy wspomagający obliczenia symboliczne. Rozważmy na przykład poniższy kod w języku Python:

```
a = 23.0
b = 3.0
print ( (a/34+1/b)**2 )
```

Program ten wypisze na ekranie przybliżoną wartość wyrażenia po podstawieniu zmiennych $a = 23, b = 3$ liczbę 1.0197. Niewykonanie dwóch pierwszych podstawień zastutkuje błędem interpretera Python.

Nieco inaczej sytuacja wygląda w przypadku systemu CAS. Tutaj jedynie informujemy system, że zmienne a, b będą symbolami i możemy rozwinąć wyrażenie algebraiczne zawierające te symbole. Wykonując:

```
var('a,b')
show( expand( (a/34+1/b)**2)   )
```

Otrzymamy w wyniku: $\frac{1}{1156}\,a^2 + \frac{a}{17\,b} + \frac{1}{b^2}$

Współczesne systemy algebry komputerowej nie ograniczają się do manipulacji wzorami matematycznymi. Z reguły są wyposażone w system obliczeń numerycznych i bogaty zestaw narzędzi wizualizacyjnych. Na dzień dzisiejszy możliwości większości systemów CAS są zbliżone i główne różnice polegają na języku programowania i licencji na k†órej dostępne jest oprogramowanie.

W proponowanym podejściu opieramy się na systemie SageMath, który jest wolnym i otwartym oprogramowaniem. Eliminuje to koszty licencji. Ponadto SageMath korzysta z popularnego języka Python, który uczniowie mogą uczyć się podczas lekcji informatyki.

2.1.3 Od języka Python do systemu SageMath

Python

Python rozwijał się już od lat dziewięćdziesiątych ubiegłego stulecia. Jednak jego niesłychana popularność przypada na czasy obecne. W Stanach Zjednoczonych większość projektów programistycznych dotyczy właśnie tego języka programowania. Python posiada rozbudowany pakiet bibliotek standardowych, cechuje go czytelność i klarowność kodu przez co jego składnia jest przejrzysta i zwięzła. Poza tym Python wspiera różne sposoby programowania: proceduralny, obiektowy oraz funkcyjny. Dzięki tym zaletom Norwegia jako pierwszy kraj europejski systemowo wprowadziła wspomniany język programowania do szkół. Uczniowie zdobywają kolejne certyfikaty powierdzające umiejętność programowania na danym poziomie.

Ekosystem Scipy

Python jest językiem intensywnie używanym do pracy naukowej i edukacji. Zestaw najbardziej powszechnych narzędzi zwany jest ekosystemem scipy. W zakres wchodzą m.in.:

- NumPy, podstawowy pakiet do obliczeń numerycznych wzorowany w swojej koncepcji na oprogramowaniu Matlab
- The SciPy biblioteka metod numerycznych
- Matplotlib, pakiet rysujący wykresy
- SymPy, biblioteka do obliczeń symbolicznych (CAS)

SageMath

SageMath jest systemem algebry komputerowej. Pierwsza wersja SageMath została wydany w dniu 24 lutego 2005 roku jako wolne i otwarte oprogramowanie zgodnie z warunkami GNU General Public License. Można powiedzieć, że Sage jest „nakładką” na Pythona, która integruje wiele specjalistycznych matematycznych pakietów oraz setki tysięcy unikalnych linii kodu dodawania nowych funkcji. Możliwości i elastyczność SageMath są przeogromne, dlatego warto wdrożyć powyższy język programowania także w szkole. Nie bez znaczenia jest fakt, że jest to oprogramowaniem otwarte i jak dotychczas darmowe. Nauczyciel i uczniowie mogą mieć dostęp do platformy w każdym miejscu i czasie, jeśli tylko mają dostęp do internetu. Poniżej w kilku punktach pokazane są najważniejsze zalety i możliwości zastosowania Sage’a w szkole na lekcjach przedmiotów ścisłych.

Ekosystem Scipy vs SageMath

System algebry komputerowej SageMath jest olbrzymim zbiorem narzędzi i zawiera w sobie między innymi narzędzią z ekosystemu Scipy. Zasadniczą różnicą jest jednak wspólny interface do wszystkich narzędzi. Sposób użycia SageMath jest zoptymalizowany na pracę interaktywną i wygodę widzianą w punktu widzenia matematyka (czy fizyka). Uchuchamiając system SageMath mamy do dyspozycji interpreter Pythona 2.7 z dwoma kluczowymi różnicami:

1. Każde polecenie jest przerabiane przez tzw. preparser zanim zostanie wysłane do interpretera Pythona. Preparser zmienia m.in.:
   - zapis potęgi `2^3` na zgodny ze składnią Pythona `2**3`
   - napis `1` na `Integer(1)`
   - napis `1.0` na `RealNumber(1.0)`
2. Automatycznie wczytywane jest ok. 2000 pożytecznych funkcji takich jak `plot`, `simplify`, itp. oraz definiowana jest zmienna symboliczna `x`.

Dlatego by np. rozwiązać równanie kwadratowe w SageMath, wystarczy napisać `solve(x^2+2*x+1==0,x)` i otrzymamy odpowiedź. Korzytając z podejścia prezentowanego przez ekosystem scipy należałoby wybrać i załadować odpowiedni moduł, zdefiniować zmienną i dopiero wtedy przystąpić do właściwego rozwiązywania.

Powyższe zalety SageMath skłoniły nas do zastosowania właśnie tego systemu na lekcjach fizyki, matematyki i chemii. Należy jednak podkreślić, że posługiwanie się SageMath jest **faktycznie programowaniem w języku Python** i jeśli uczniowie posiądą tą umiejętność na lekcjach informatyki to jedyną dodatkową niezbędną wiedzą są dwa powyższe punkty. W efekcie rozwiązanie oparte na systemie SageMath dostarczą o wiele bardziej efektywnego narzędzia a z drugiej strony nie nakłada praktycznie żadnych dodatkowych wymagań na ucznia, który uczył się Pythona na informatyce.

2.2 SageMath w pigułce

2.2.1 Bogaty i szybki kalkulator naukowy.

Praktycznie, każda ważna funkcja, wzór matematyczny są już zaimplementowane w języku SageMath. Poniżej tylko niektóre instrukcje, które można wykorzystać w szkole średniej:

- wartość bezwzględna – *abs*,
- rozkład na czynniki pierwsze - *factor*,
- silnia – *factorial*,
- symbol Newtona – *binomial*,
- rozwiąż równanie – *solve*,
- narysuj wykres – *plot*,
- następna liczba pierwsza – *next_prime*,
- NWD – *gcd*, NWW – *lcm*,
- pochodna – *diff*,
- całka – *integrate*.

Pierwszy przykład pokazuje możliwości rachunkowe Sage'a. Można używać go do sprawdzania: przeprowadzonych rachunków, zadań domowych przez uczniów. Jeżeli nauczyciel pozna powyższy język programowania, to może stworzyć kod, który umożliwia rozwiązywanie zadań rachunkowych „krok po kroku".

```
print "(4/3+5/5)-(5/2-4/6) =", (4/3+5/5)-(5/2-4/6)
print "(3^15-3^13)/(3^13+3^14) =", (3^15-3^13)/(3^13+3^14)
print "1001 =", factor(1001)
print "(sqrt(8)-sqrt(2))^2 =", (sqrt(8)-sqrt(2))^2
print "5! =", factorial(5)
print "NWD(354,222) =", gcd(354, 222)
```

Wykonując ten kod otrzymamy następujący wynik:

```
(4/3+5/5)-(5/2-4/6) = 1/2
(3^15-3^13)/(3^13+3^14) = 2
1001 = 7 * 11 * 13
(sqrt(8)-sqrt(2))^2 = 2
5! = 120
NWD(354,222) = 6
```

SageMath ma wbudowane różne systemy arytmetyczne i może np. przybliżać liczby niewymierne z dowolną precyzją. Te możliwości wykorzystaliśmy w naszym projekcie, przed wszystkim w szyfrowaniu RSA oraz w rozdziale dotyczących przybliżeń wyrażeń niewymiernych.

```
show(sqrt(2), "=", N(sqrt(2), digits=60))
show(pi, "=", N(pi, digits=60))
show(2^168+5^80)
```

Wykonując ten kod otrzymamy następujący wynik:

$$\sqrt{2} = 1.41421356237309504880168872420969807856967187537694807317668$$
$$\pi = 3.14159265358979323846264338327950288419716939937510582097494$$
$$82718435399721924198287929350313460725034243008818892481$$

2.2.2 Działania na wyrażeniach logicznych.

SageMath a także sam język Python, umożliwia wykonanie działań na wyrazeniach logicznych. Może się do okazać przydatne w wielu dziedzinach. Dla przykładu rozważmy zagadkę:

Zagadka o kłamcach

Sa dwa rodzaje ludzi - jeden zawsze kłamie a jeden mówi zawsze prawdę. Ala i Bolek należą do jednej z tych kategorii. Ala powiedziała: ja i Bolek jesteśmy kłamcami. Kto jest kłamcą a kto mówi prawdę?

Stosując SageMath możemy przyjąć następującą interpretacje: niech a będzie prawdą jeśli Ala jest prawdomówna a b będzie prawdą jeśli Bolek jest prawdomówny. Wtedy możemy w Sage zapisać:

```
f = propcalc.formula("a&(~a&~b) | ~a&(~(~a&~b))")
show(f)
print(f.truthtable())
```

Wykonując ten kod otrzymamy następujący wynik:

```
a      b      value
False  False  False
False  True   True
True   False  False
True   True   False
```

Od razu widać, że jedynym rozwiązaniem jest takie w którym Ala kłamie a Bolek mówi prawdę.

2.2.3 Działania na wyrażeniach algebraicznych.

Jedną z ważniejszych możliwości, którą można zastosować na lekcjach matematyki, fizyki oraz chemii jest przeprowadzanie rachunków nie tylko na liczbach, ale także na zmiennych.

SageMath doskonale sobie radzi z obliczeniami symbolicznymi, czyli potrafi przeprowadzać obliczenia, przekształcenia na wyrażeniach algebraicznych. Przez co możemy modyfikować postać wzoru, obliczać jedną zmienną przy pomocy innych, wyprowadzać wzory – ogólne rozwiązania. Poniżej pokazane są proste przykłady dotyczące wzorów skróconego mnożenia oraz wyrażeń wymiernych.

Wzory skróconego mnożenia.

```
var('a','b')
wzor1 = (a+b)^2
wzor2 = (a-b)^2
wzor3 = (a+b)*(a-b)
show (wzor1, "=", wzor1.canonicalize_radical())
show (wzor2, "=", wzor2.canonicalize_radical())
show (wzor3, "=", wzor3.canonicalize_radical())
a=sqrt(3)
b=2
wzor1=(a+b)^2
wzor2=(a-b)^2
wzor3=(a+b)*(a-b)
show (wzor1, "=", wzor1.canonicalize_radical())
show (wzor2, "=", wzor2.canonicalize_radical())
show (wzor3, "=", wzor3.canonicalize_radical())
```

Wykonując ten kod otrzymamy następujący wynik:

$$(a+b)^2 = a^2 + 2\,ab + b^2$$

$$(a-b)^2 = a^2 - 2\,ab + b^2$$

$$(a+b)(a-b) = a^2 - b^2$$

$$\left(\sqrt{3}+2\right)^2 = 4\,\sqrt{3} + 7$$

$$\left(\sqrt{3}-2\right)^2 = -\,4\,\sqrt{3} + 7$$

$$\left(\sqrt{3}+2\right)\left(\sqrt{3}-2\right) = -\,1$$

Zamiana postaci wyrażenia algebraicznego.

```
var('n')
wyr = n^3-(n-1)^3
show ("n=2")
show(wyr," = ", wyr.canonicalize_radical()," = ",wyr.substitute(n =
 ↪ 2))
```

Wykonując ten kod otrzymamy następujący wynik:

$$n = 2$$

$$-(n-1)^3 + n^3 \;=\; 3\,n^2 - 3\,n + 1 \;=\; 7$$

Zamiana postaci wyrażenia wymiernego.

```
var('z')
wyr = (z^2+3*z)/z
show (wyr)
show (wyr.canonicalize_radical())
show (wyr.subs(z=x+1))
show (wyr.subs(z=2))
```

Wykonując ten kod otrzymamy następujący wynik:

$$\frac{z^2 + 3\,z}{z}$$

$$z + 3$$

$$\frac{(x+1)^2 + 3\,x + 3}{x + 1}$$

$$5$$

Upraszczanie wyrażeń zawierających funkcje trygonometryczne wymaga użycia metod `.trig_simplify`. Na przykład aby wykorzystać jedynkę trygonometryczną należy wykonać:

```
( sin(x)^2+cos(x)^2 ).trig_simplify()
```

Jeśli chcemy udowodnić tożsamość trygonometryczną lepiej użyć `bool` niż próbować uprościć jedną ze stron by przypominała drugą:

```
expr = (2*sin(x)^2-1)/(sin(x)*cos(x)) == tan(x)-cot(x)
show(expr)
bool(expr)
```

Wykonując ten kod otrzymamy następujący wynik:

$$\frac{2\,\sin\left(x\right)^2 - 1}{\cos\left(x\right)\sin\left(x\right)} = -\cot\left(x\right) + \tan\left(x\right)$$

$$True$$

2.2.4 Rozwiązywanie równań i układów równań.

Największą ilość zadań z przedmiotów ścisłych jaką uczeń musi wykonać to rozwiązywanie równań i układów równań. Oczywiście żadne narzędzie nie zastąpi samodzielnego rozwiązywania zadań przez uczniów, ale może być bardzo przydatne do ćwiczeń, sprawdzania wyników, czy też rozwiązywania równań, które uczeń musi samodzielnie wyprowadzić na podstawie zadań tekstowych. Powyższy język umożliwia rozwiązywanie nawet trudnych równań i układów równań przy pomocy jednej instrukcji – `solve`.

Poniżej zamieszczamy przykłady, które demonstrują użycie instrukcji na podstawie równania kwadratowego oraz prostego układu równań z dwoma niewiadomymi. Dla nauczycieli prowadzących zajęcia dodatkowe z matematyki dla uczniów zdolnych nie bez znaczenia będzie fakt, ze Sage rozwiązuje równania w zbiorze liczb zespolonych oraz macierzowe.

Równanie kwadratowe.

```
var('a','b','c')
r_kwadr = a*x^2 + b*x + c == 0
show(solve(r_kwadr, x))
a = 1
b = 4
c = -5
r_kwadr = a*x^2 + b*x + c == 0
show (solve(r_kwadr, x))
```

Wykonując ten kod otrzymamy następujący wynik:

$$\left[x = -\frac{b + \sqrt{b^2 - 4\,ac}}{2\,a}, x = -\frac{b - \sqrt{b^2 - 4\,ac}}{2\,a}\right]$$

$$[x = (-5)\,, x = 1]$$

Układ równań z dwoma niewiadomymi.

```
var('x','y')
solve([x-3*y==2, x-2*y==8],x,y)
```

Wykonując ten kod otrzymamy następujący wynik:

$$[[x == 20, y == 6]]$$

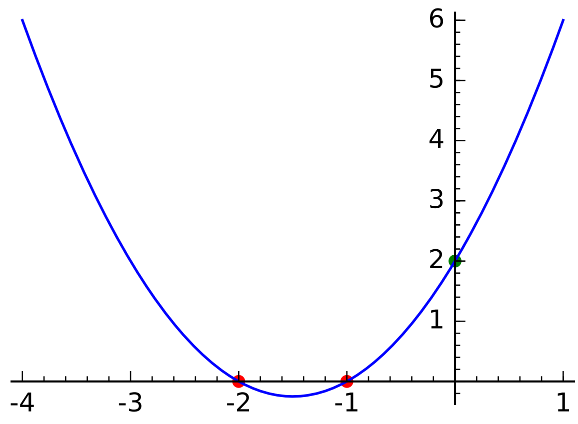
2.2.5 Wizualizacja.

Uczniowie dzięki stronom internetowym czy platformom społecznościowym odbierają świat „obrazkowo", czyli wiążą krótkie informacje z odpowiednim obrazkiem, zdjęciem, wykresem. Dlatego też wizualizacja dla obecnego pokolenia młodzieży jest bardzo ważna. Sage umożliwia rysowanie wykresów funkcji w prosty sposób. Zatem możemy szybko przedstawiać rozwiązania na wykresie lub też rysować interesujące nas funkcje podczas lekcji. Uczniowie mogą modyfikować już istniejący kod programu i analizować otrzymane funkcje. Można to zastosować nie tylko na matematyce ale także na pozostałych przedmiotach ścisłych.

Poniższy program dotyczy miejsc zerowych funkcji kwadratowej. Obliczono w nim pierwiastki funkcji kwadratowej, punkt przecięcia funkcji z osią Y następnie narysowano wykres funkcji i zaznaczono wyróżnione punkty.

```
a = 1
b = 3
c = 2
d = b*b - 4*a*c
f(x) = a*x*x + b*x + c
if d < 0:
    print "Brak rozwiązania dla liczb rzeczywistych!"
    xmin,xmax =-5, 5
    x1,x2 = 0,0

if d > 0:
    x1 = float((-b-sqrt(d))/(2*a))
    x2 = float((-b+sqrt(d))/(2*a))

    print "x1=", x1, ", ", "x2=", x2

    if x1<x2:
        xmin,xmax = x1-2,x2+2
    else:
        xmin,xmax = x2-2,x1+2

p1 = point((x1,0), color="red", size=35)
p2 = point((x2,0), color="red", size=35)
p3 = point((0, c), color="green", size=35)
q = plot(f(x),(x,xmin,xmax))
show(p1+p2+p3+q, figsize=4)
```

Wynikiem działania powyższego kodu jest wykres Rys. 2.1.



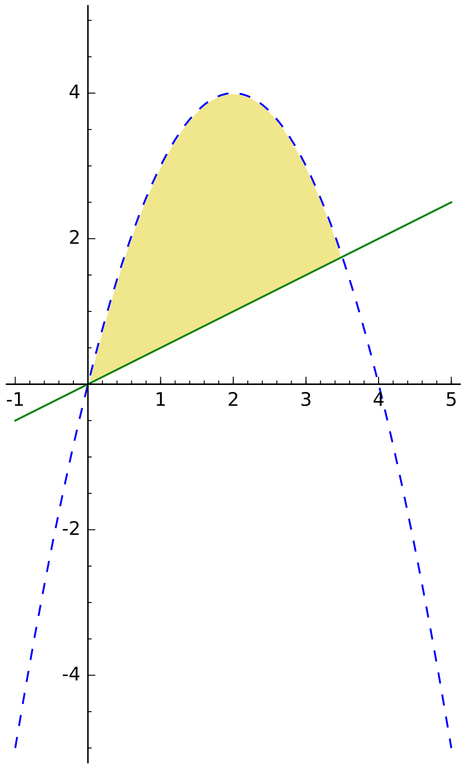
Rys. 2.1: Parabola z miejscami zerowymi.

Dzięki instrukcji `region_plot` możemy na wykresie przedstawiać także rozwiązanie układów nierówności.

```
var('x','y')
g1 = -x^2/4+1*x
```

```
g2 = 0.25*x
f1 = plot(g1, (x,-0.4,4.5), linestyle="--")
f2 = plot(g2,(x,-0.4,4.5), linestyle="-", color="green")
rp = region_plot([y<g1,y>=g2],(x,-0.3,4.5),(y,-1,1.2), incol="khaki")
show(f1 + f2 + rp, figsize=5)
```

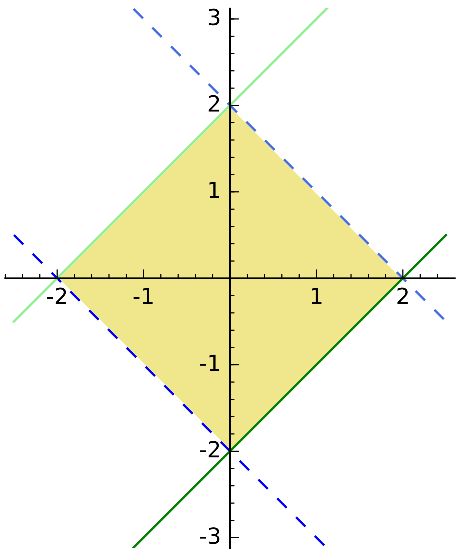
Wynikiem działania powyższego kodu jest wykres Rys. 2.2.



Rys. 2.2: Przykład wizualizacji nierówności z pomocą *region_plot*

```
var('x','y')
g1 = -x-2
g2 = -x+2
g3 = x-2
g4 = x+2
f1 = plot(g1, (x,-2.5,2.5), linestyle="--")
f2 = plot(g2, (x,-2.5,2.5), linestyle="--", color="royalblue")
f3 = plot(g3, (x,-2.5,2.5), linestyle="-", color="green")
f4 = plot(g4, (x,-2.5,2.5), linestyle="-", color="lightgreen")
rp = region_plot([y>g1,y<g2,y>=g3,y<=g4],\
      (x,-2,2),(y,-2,2), incol="khaki")
show(f1 + f2 + f3 + f4 + rp, figsize=5,ymax=3,ymin=-3)
```

Wynikiem działania powyższego kodu jest wykres Rys. 2.3.



Rys. 2.3: Wizualizacja rozwiązania nierówności za pomocą *region_plot*.

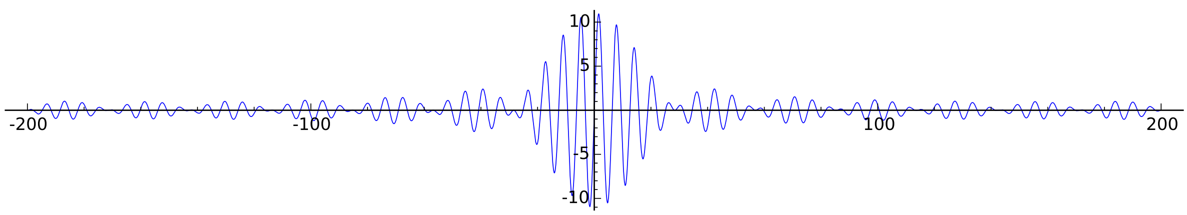
W systemie SageMath rysowaną funkcję możemy stworzyć również algorytmicznie. Wyobraźmy sobie, że chcemy zobaczyć jak wygląda wykres:

$$f(x) = \sum_{i=0}^{N} \sin(\omega_i x)$$

dla dużych wartosci N. W systemie Sage możemy zsumować wiele funkcji wykorzystując pętlę. Tworzenie takich złożonych funkcji ma zastosowanie na lekcjach fizyki podczas omawiania zjawisk falowych. Poniższy kod doda fale o zbliżonych częstościach:

```
f = sum([sin(w*x) for w in srange(0.9,1.101,0.02)])
plot(f,(x,-200,200),figsize=(10,2),thickness=0.5)
```

Wynikiem działania powyższego kodu jest wykres Rys. 2.4.

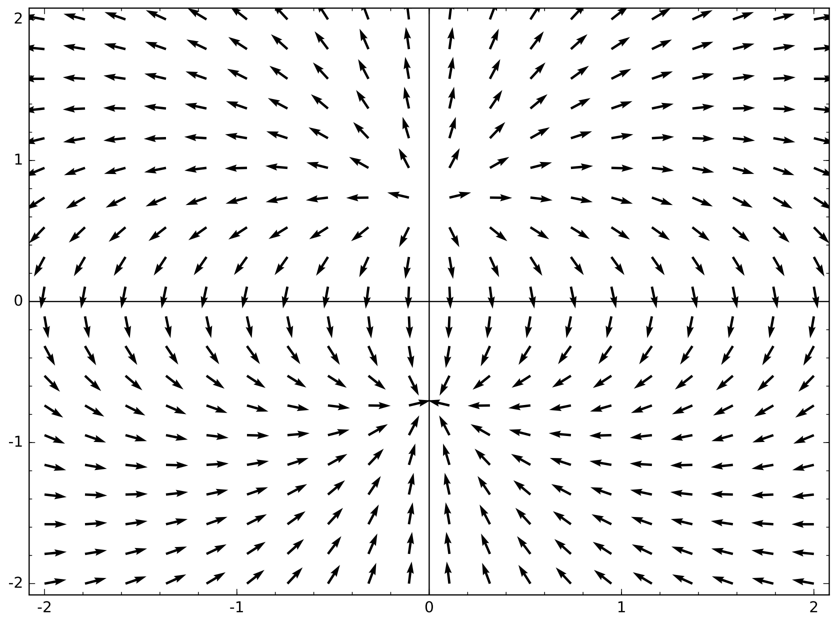


Rys. 2.4: Wizualizacja paczki falowej.

Ciekawym przykładem wizualizacji jest graficzne przedstawianie pól wektorowych. Podczas lekcji fizyki, można wykorzystać możliwości SagaMath do rysowania pół wektorowych. Poniżej zamieszczony jest przykład pola pochodządzego od dipola magnetycznego. Pole jest trójwymiarowe, w przykładzie przedstawiony jest przekrój takiego pola przez płaszczyznę $x = 0$

```
var('x y z',domain='real')
m = 1
r = sqrt(x^2+y^2+z^2+1e-6)
Bx = 3*m*x*z/(r^5)
By = 3*m*y*z/(r^5)
Bz = 3*m*z^2/(r^5)-m/r^3
B = vector( [Bx,By,Bz])
Bmod = B.subs(x==0)[1:].norm()
plot_vector_field(B.subs(x==0)[1:]/Bmod,(y,-2,2),(z,-2,2))
```

Wynikiem działania powyższego kodu jest wykres Rys. 2.5.



Rys. 2.5: Wizualizacja pola wektorowego.

2.2.6 Interakcja

Duże walory edukacyjne mają programy komputerowe pozwalające wykonać animację lub dynamicznie zmieniać parametr i obserwować jak wpływa on na rozwiązanie. Tego typu elementy, najczęściej wykonane są w technologii Flash lub javascript są atrakcyjną cyfrową pomocą naukową. Zazwyczaj jednak uczeń ograniczony jest jedynie do interakcji z takim programem. System SageMath pozwala pojść jeden krok dalej - pozwala na bardzo łatwe tworzenie tych elementów. Korzystając z prostych funkcji uczeń może samodzielnie stworzyć interaktywną applikację, która może ilustrować badane zagadnienie.

Zilustrujemy na przykładzie następującego problemu:

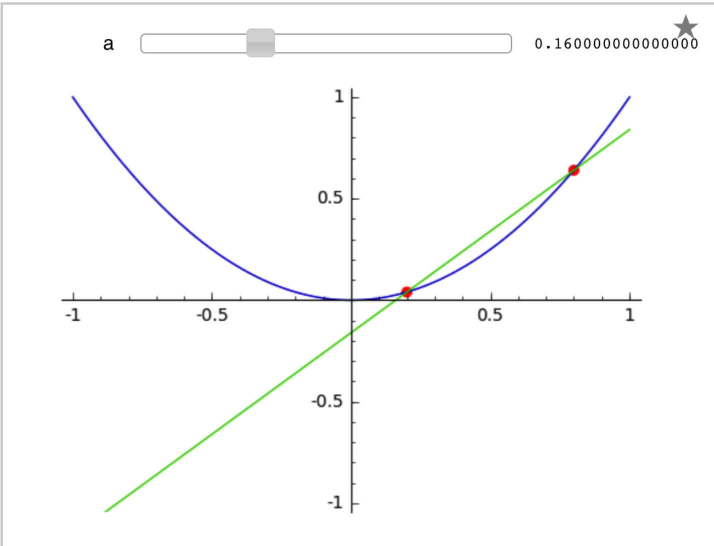
Zadanie

Zbadaj ile ma rozwiązań równanie $x^2 = x - a$ w zależności od parametru $a \in (0, \frac{1}{2})$?

W SageMath możemy narysować wykresy zarówno prostej $y = x - a$ jak i paraboli $y = x^2$ i zaznaczyć na nich pierwiastki równania $x^2 = x - a$. Wystarczy taki stworzony kod programu opakować w funkcję i dodać tzw. dekorator `@interact`. SageMath wygeneruje nam interaktywną aplikację, w której będzie można zmieniać myszką wartość parametru i obserwować jak zmienia się wykres.

```
@interact
def fun(a=slider(0,1/2,0.01)):
    p = plot([x^2,x-a],(x,-1,1),figsize=5,ymax=1,ymin=-1)
    assume(x,'real')
    pkt = [(x.subs(s),x.subs(s)-a) for s in solve(x^2==x-a,x)]
    if pkt:
        p += point(pkt,size=40,color='red')
    else:
        print "Nie ma pierwiastkow"
    show(p)
```

Wynikiem działania powyższego kodu jest Rys. 2.6.



Rys. 2.6: Interaktywna ilustracja równania $x^2 = x - a$.

2.2.7 Sage = interdyscyplinarność.

Podsumowując SageMath umożliwia:

1. Szybkie i dokładne obliczenia nawet dowolnie dużych liczb.
2. Przeprowadzanie obliczeń na wyrażeniach algebraicznych. rozwiązywanie równań w zbiorze liczb zespolonych, rozwiązywanie układów równań i nierówności.
3. Wizualizację rozwiązań na wykresach, rysowanie wykresów funkcji.
4. Obliczanie pochodnych, całek i wielu innych działań matematycznych.

Podanto SageMath to bardzo dobre i bogate narzędzie programistyczne, dzięki któremu możemy łączyć przedmioty ścisłe: informatykę, programowanie, matematykę, fizykę, chemię. Czy istnieją ograniczenia dla Sage'a? Tak, ale pewnie wcześniej natrafimy na ograniczenia naszej wyobraźni.

2.3 Narzędzia i metody pracy

2.3.1 Technologiczne aspekty korzystania z SageMath

Czym jest SageMath

Pakiet SageMath jest wolnym i otwartym (GNU GPL) oprogramowaniem matematycznym. Został on oparty na języku Python i oferuje pełne spektrum interfejsów dostępu, począwszy od terminala tekstowego po interfejs graficzny wykorzystujący przeglądarkę internetową.

SageMath jest dystrybuowany jako dość pokaźny pakiet instalacyjny (ok 1GB), zawierający następujące narzędzia:

- Python 2.7
- NumPy
- SciPy
- matplotlib
- Sympy
- Maxima
- GAP
- FLINT
- R

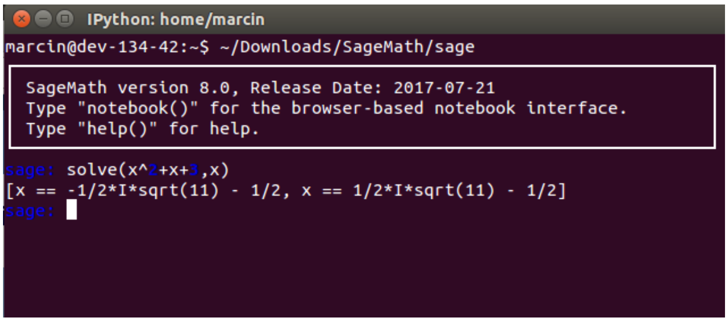
Ambicją grupy programistów tworzących SageMath jest stworzenie pełnowartościowej otwartej alternatywy dla komercyjnych pakietów: Magma, Maple, Mathematica oraz Matlab.

Linia poleceń

Terminal tekstowy jest tradycyjną metodą korzystania z systemów komputerowych. Pomimo, że terminal jest technologią sięgającą swych początków w latach siedemdziesiątych XX wieku, to sposób pracy ugruntował sobie miejsce w nowoczesnych świecie technologii informatycznych. Współczesnie, terminal to tak zwany softwarowy emulator terminala tekstowego. Sposób korzystania jest oparty na koncepcji edycji pojedyńczej linii i wysłania jej do wykonania programowi, który jest uruchomiony. Przykładem może być powłoka systemu operacyjnego (windows shell, bash) lub interpreter Pythona lub SageMath. Praca w takim trybie jest szczególnie przydatna podczas eksploracji naukowych i jest powszechnie stosowania na całym świecie.

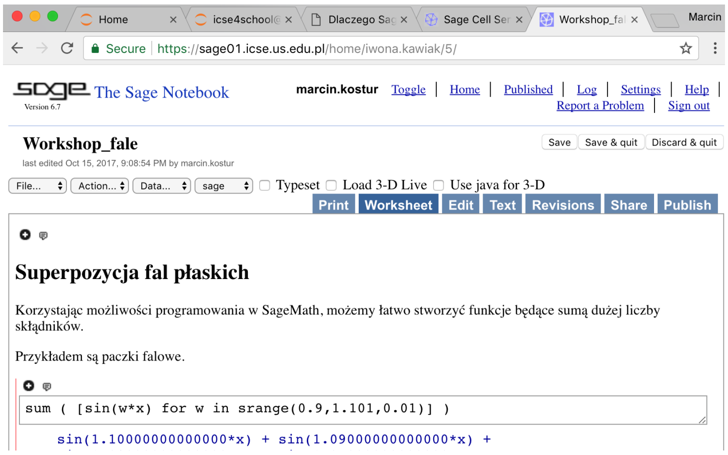
Notatnik

System typu notatnik łączy w sobie dwie koncepcje. Pierwszą jest dokument zawierający bogaty (np. ilustrowany) sformatowany tekst oraz kod programu komputerowego. Ponadto tzw. wyjscie programu może być zarówno tekstem (jak w przypadku terminala tekstowego) jak i formułą, rysunkiem czy też interaktywnych elementem (np. suwakiem). Drugą koncepcją



Rys. 2.7: Przykładowa sesja w linii poleceń w systemie SageMath.

jest system dostępny przez przeglądarkę internetową pozwalającą na edycję powyższego dokumentu i wykonywanie zawartych w mim bloków kodu.



Rys. 2.8: Przykładowy notebook w systemie SageMath.

W projekcje iCSE4school notatnik SageMath (tzw. sagenb) był podstawowym narzędziem pracy. Jako aplikacja „webowa" oferował on dwa sposoby pracy:

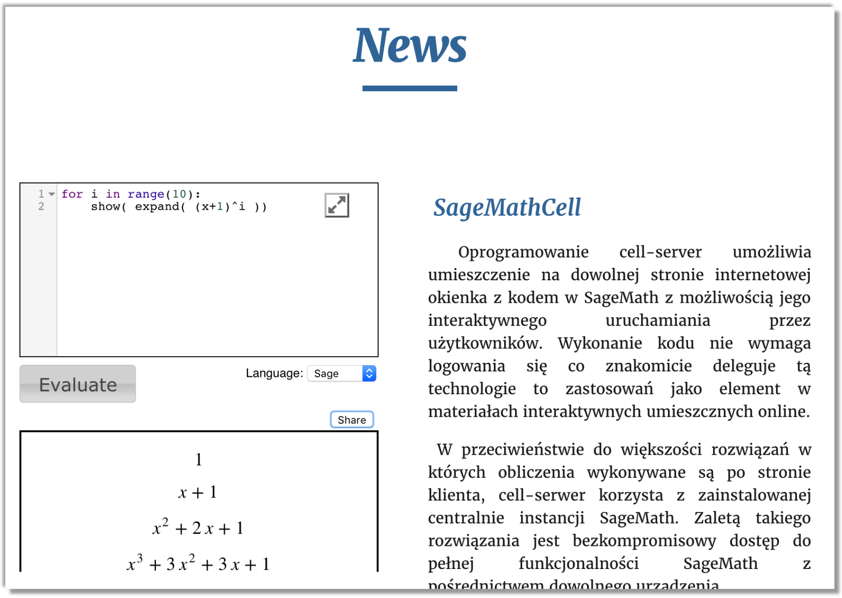
- praca w „chmurze" - korzystanie ze wspólnego serwera notatników skonfigurowanego dla szkół partnerskich.
- praca lokalna na na własnej instalacji SageMath zawierająca notatnik sagenb.

Pierwszy sposób pracy oferuje oprócz samego notatnika system logowania, uwspólniania dokumentów oraz zarządzania kontami. Okazało się to bardzo przydatne w pracy na lekcji. Uczeń i nauczyciel mając jakiekolwiek urządzenie z dostępem do internetu mógł bez wykonywania żadnych instalacji oprogramowania w dowolnej chwili rozpocząć pracę z SageMath. Podandto bradzo przydatną cechą była możliwość uwspólniania notatników i ich publikacji. Na przykład można podejrzeć listę tzw. opublikowanych prac nauczycieli i uczniów na serwerach:

- https://sage01.icse.us.edu.pl/pub/.
- https://sage03.icse.us.edu.pl/pub/.

Interaktywne strony internetowe: system SageMathCell

Oprogramowanie SageMathCell umożliwia umieszczenie na dowolnej stronie internetowej okienka z kodem w SageMath z możliwością jego interaktywnego uruchamiania przez użytkowników (patrz rys. Rys. 2.9.). Wykonanie kodu nie wymaga logowania się co znakomicie deleguje tą technologie to zastosowań jako element w materiałach interaktywnych umieszcznych online. W przeciwieństwie do większości rozwiązań w których obliczenia wykonywane są po stronie klienta, SageMathCell korzysta z zainstalowanej centralnie instancji SageMath. Zaletą takiego rozwiązania jest bezkompromisowy dostęp do pełnej funkcjonalności SageMath z pośrednictwem dowolnego urządzenia.



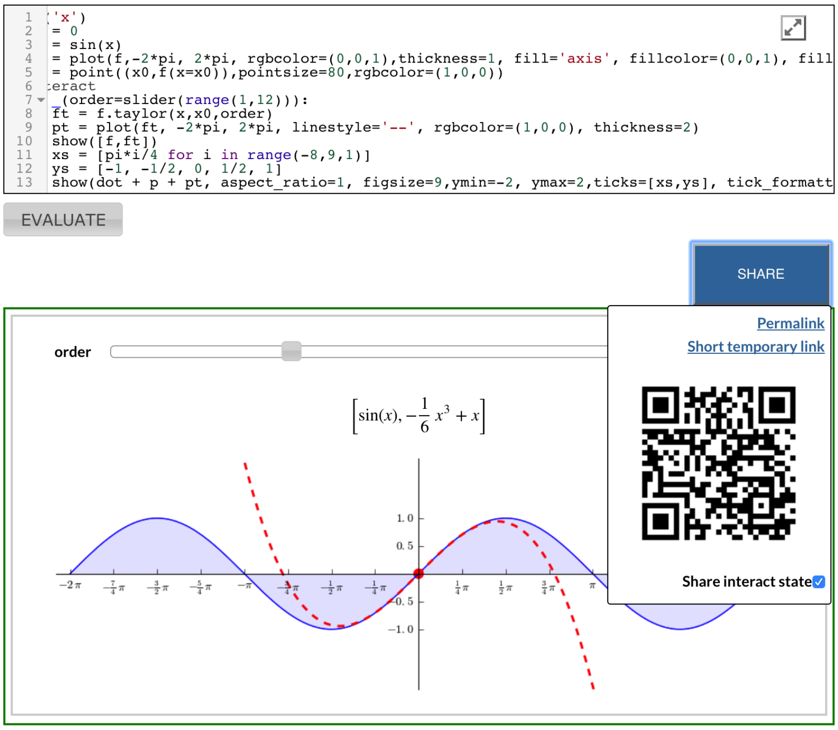
Rys. 2.9: Przykładowa strona (dla przykładu stworzona w google sites) zawierająca *iframe* z zanurzoną komórką SageMathCell.

SageMathCell różni się tym od notatnika SageMath, że nie posiada możliwości zapisu edytowanego kodu. Istnieje jednak możliwość odtworzenia całej komórki wraz z kodem za pomocą linka (patrz rys. Rys. 2.10). Przekazanie takiego linka może odbyć się za pomocą kodu QR i wystarczy telefon komórkowy z dostępem do internetu by móc kod ten wykonać i edytować.

Stosowanie SageMath w praktyce szkolnej

Podczas pracy z SageMath w trakcie projektu iCSE4school, wykrystalizowało się kilka sposobów jego użycia. Po pierwsze dostęp przez konsolę - jako najbardziej zaawansowany - nie został użyty. Powszechnie stosowano notatnik *sagenb*. Spotkał się on z bardzo pozytywnym przyjęciem przez zarówno uczniów jak i nauczycieli. W naturalny sposób nauczyciele używali go do różnych celów. Można wyróżnić następujace rodzaje pracy:

- Praca wyłącznie z komórkami z kodem Sage lub Python. W tym trybie notatnik praktycznie pełnił rolę oprogramowania znanego pod nazwą IDE (Integrated Development Environment). Jest to oprogramowie umożliwiające pisanie i wykonywanie kodu.
- Intensywne użycie komórek tekstowych i narzędzi formatowania tekstu. W tym trybie



Rys. 2.10: Przykładowy element typu *interact* uruchomiony w systemie cell-server. Widoczne jest pojedyńcze okno z kodem programu a poniżej wynik jego działania: interaktywny suwak i wykres. Po prawej stronie rysunku widzimy możliwość stworzenia linku zawierającego powyższy program. Kod QR zawiera ten sam link. Link występuje w dwóch postaciach, krótkiej i długiej - zawierającej w url spakowany cały kod.

zdarzało się wykorzystywać notatnik Sage jako edytor tekstu. Przydatną cechą okazała się możliwość opublikowania i uwspólniania dokumetnów na serwerze.

- Stosowanie zarówno komórek tekstowych ji kodu. Duża część notatników zapisanych przez uczniów zalicza się do tej kategorii. Jednym z przykładów takiego użycia są sprawozdania z pracowni fizycznej.

Trzecia metoda interakcji z SageMath - cell server była stosowana podczas tworzenia materiałów - takich jak na przykład ten manuskrypt.

Należy zdecydowanie stwierdzić, że centralna instalacja serwera notatników jest bardzo przydatnym rozwiązaniem. Wpisująca się najnowsze trendy praca w „chmurze" ma następujące zalety dla użytkowników:

- Wszechobecność materiałów. Można pracować w systemie SageMath wszędzie tam gdzie jest przeglądarka internetowa. Dzięki dostępności taniego internetu mobilnego jest to warunek spełniony praktycznie wszędzie.
- Mniejsze ryzyko utraty danych przez zwykłe zgubienie lub awarię dysku twardego. Oczywiście to tego niezbędny jest sprawny system kopii zapasowowych działający na instalacji chmurowej. W naszym przypadku zastosowano rozwiązanie umożliwiające zapis codziennego stanu serwera. W przypadku całkowitego zniszczenia serweru lub np. skasowania danych przez złośliwe oprogramowanie czy cyberatak, można było odtworzyć stan dokumentów z dowolnego dnia przed tym wydarzeniem.
- System notatnik «sagenb» ma cechy systemu e-learningowego, umożliwia dystrybuję materiałów, możliwy jest wgląd nauczyciela w pracę ucznia oraz tzw. publikacja materiałów przez zarówno uczniów jak i nauczycieli. Zaobserwowano intesywne wykorzystywanie tych możliwości podczas projektu.
- Warto zauważyć, że największa obawa przed korzystaniem z rozwiązań chmurowych czyli wyciek danych nie stanowi problemu w przypadku używania SageMath w pracy. Z reguły bowiem treści notatników nie sa poufne. Jedynym zagrożeniem jest utrata danych, która może być praktycznie wyeliminowana poprzez stosowanie systemu kopii zapasowych.

Notatnik oparty na przeglądarce internetowej jest technologią, która jest intensywnie rozwijana w świecie nauki. Najnowocześniejszym rozwiązaniem jest tzw. Jupyter notebook[1]. Został on włączony do projektu SageMath i w najnowszych wersjach zastępuje klasyczny notatnik sagenb. W trakcie projektu Jupyter notebook nie oferował jeszcze krytycznych dla realizacji dydaktyki cech takich jak uwspólnianie dokumentów czy ich publikacja. Dlatego projekt został przeprowadzony na poprzednim rozwiązaniu.

2.3.2 Modele i metody nauczania z wykorzystaniem SageMath

Z punktu widzenia aktywności ucznia modele nauczania można podzielić na grupy:

- podający (uczeń jest biernym słuchaczem),
- podający interaktywny, (uczy myślenia logicznego, wnioskowania),

[1] Projekt ma stronę internetową http://jupyter.org

- bezpośredni (uczeń wykonuje ćwiczenia, poznaje algorytmy i procedury, dyskutuje),
- poszukujący (twórcze rozwiązywanie problemów),
- współpracy z innymi, w tym uczenie innych.

Flipped Classroom (Flipped Teaching) - model odwróconej szkoły (klasy)

Odwrócone uczenie, bardzo ogólnie, stanowi zamianę tego, co tradycyjnie przekazywane było uczniom w postaci wykładu oraz dyrektywnych instrukcji, do czasu poza lekcją. Uczniowie w dowolnym czasie (rano, wieczorem, w podróży, itp.) zapoznają się ze wskazanymi przez nauczyciela materiałami multimedialnymi (filmy edukacyjne, nagrane wykłady, prezentacje, podcasty, ebooki, itp.) zamieszczonymi w Internecie, analizują je dyskutując w razie potrzeby z rówieśnikami, a czas lekcyjny poświęca się na dodatkowe wyjaśnienia nauczyciela oraz wykonywanie ćwiczeń pogłębiających wiedzę i utrwalających umiejętności. Uczniowie uczący się szybko mogą w krótszym czasie zgłębić daną wiedzę, z kolei uczniowie pracujący powoli mogą dostosować naukę do swojego tempa (zapoznając się z materiałami w Internecie mogą zatrzymać nauczyciela, cofnąć nagranie, przewinąć je wielokrotnie). Podczas lekcji uczniowie pracują w oparciu o przeanalizowany wcześniej materiał zachęcani przez nauczyciela do myślenia na wyższym poziomie. Czas lekcyjny przy zastosowaniu metody Flipped Teaching można dzięki temu wykorzystać mądrzej i bardziej wartościowo, dopasowując go do indywidualnych potrzeb uczniów. Nauczyciele stosujący metodę wskazują na nietypowy wygląd takich lekcji, podczas których każdy z uczniów pracuje we własnym tempie. Dla obserwatora mogą one stwarzać wrażenie chaosu, choć jest to zamierzone i dla indywidualnych uczniów efektywne. Metoda ta to przykład „blended learningu”, kiedy wiedza teoretyczna dostarczona w postaci e-contentu poprzedza warsztat z trenerem. Podczas stosowania metody Flipped Teaching uczniowie wyposażeni są w narzędzia, umiejętność myślenia, analizowania i przetwarzania informacji oraz rozwiązywania problemów, niezależnie od tego, co w przyszłości będą robić. Metoda ta skutecznie przygotowuje do samodzielności i uczenia się przez całe życie (LLL), zatem jest uniwersalna, niezależnie od przedmiotu na którym jest stosowana.

Metody nauczania

W literaturze przedmiotu możemy napotkać wiele sposobów podziału metod nauczania. Dla przykładu Franciszek Szlosek proponuje podział metod nauczania na pięć głównych grup:

- Podające: wykład, opis, wyjaśnienia.
- Eksponujące: prezentacja, pokaz, demonstracja, film, ekspozycja.
- Programowane: z użyciem edukacyjnego programu komputerowego, podręcznika interaktywnego lub programowanych urządzeń dydaktycznych.
- Problemowe: wykład problemowy, wykład konwersatoryjny, klasyczna metoda problemowa, symulacje, metody aktywizujące.
- Praktyczne: ćwiczenia przedmiotowe, ćwiczenia laboratoryjne, warsztaty, eksperymenty, metoda projektów.

2.3.3 Przykłady zastosowań poszczególnych metod

Prezentacja i pokaz możliwości zastosowań SageMath

Metody eksponujące w postaci pokazów, prezentacji i demonstracji zostały wykorzystane jako pierwszy kontakt uczniów z aplikacją SageMath. Miały na celu zaciekawienie uczniów, zaintrygowanie ich a także przekonanie, że z SageMath będą w stanie sprawdzić każdą pracę domową z matematyki, czy innego przedmiotu, wymagającą obliczeń czy wykresów.

W XXXIII LO M. Kopernika w Warszawie uczniowie mogli zobaczyć różnorodne możliwości SageMath podczas pierwszych zaplanowanych dla uczniów uczestników projektu Erasmus+ zajęć. Zajęcia trwały jedną godzinę, wybrane do prezentacji zagadnienia w sposób bardzo atrakcyjny pokazywały kolorowe wykresy i animacje, ciekawe problemy rozwiązane z wykorzystaniem narzędzi SageMath - problem "wilki i króliki", szyfr RSA czy symulacje z fizyki. Ale co najważniejsze, uczniowie mogli ze swoich telefonów komórkowych czy tabletów sprawdzić sami i wykonać polecenia w SageMath.

Na tych zajęciach został także zaprezentowany szkolny serwer SageMath, uczniowie otrzymali konta, omówione zostały kolejne zajęcia a także projekt do samodzielnego wykonania na serwerze SageMath.

Uwaga metodyczna:

Warto zadbać aby uczniowie na początku cyklu zajęć widzieli ich cel w postaci zadania, jakie stawia przed nimi nauczyciel. Równie ważne jest aby uczniowie zobaczyli jakim sposobem ten cel można osiągnąć a także zainteresowali się twórczo nowo poznawanym tematem. Jeśli chodzi o nowe technologie czy narzędzia informatyczne bardzo ważne jest aby nauczyciel zapytał uczniów jakie oni sami mieliby pomysły na zastosowanie i wykorzystanie nowego narzędzia. Uczniowie wówczas mogą się wykazać czasem zaskakującą kreatywnością, czasem bardzo praktycznym podejściem.

Warsztaty - ćwiczenia praktyczne z wykorzystaniem SageMath

Warsztaty umożliwiają kształtowanie umiejętności zastosowania wiedzy w praktyce. Polegają przykładowo na rozwiązywaniu zadań, wykonywaniu doświadczeń i eksperymentów, planowaniu i wykonywaniu pomiarów, obliczeń oraz interpretowaniu wyników badań, wykonywaniu symulacji praktycznych i teoretycznych, analizowaniu i praktycznym poznawaniu zjawisk z różnych dziedzin nauki.

Warsztaty służą kształtowaniu umiejętności twórczego wykorzystania wiedzy w praktyce (np. samodzielne poznawanie cech konstrukcji, systemów, procesów, zjawisk), co zmusza ucznia do odkrywania, analizowania, pomysłowości, rozwija naturalną ciekawość, zadawanie pytań i poszukiwanie odpowiedzi.

W XXXIII LO im. M. Kopernika w Warszawie metoda warsztatów została wykorzystana jako kolejne zajęcia po prezentacji możliwości SageMath. Posłużyła do nauki praktycznego wykorzystania i sprawdzenia przez uczniów prezentowanych możliwości. Nauczyciel wybrał najpierw zestaw poleceń do wykonania przez uczniów jednocześnie prezentując na ekranie z rzutnika ich wykonanie. Następnie nauczyciel przedstawił uczniom zestaw zagadnień do samodzielnego wykonania. Takie zajęcia odbywały się zarówno podczas zajęć lekcyjnych jak i pozalekcyjnych dla grupy uczniów uczestników projektu Erasmus+. Zajęcia warsztatowe zostały również przeprowadzone do nauki tworzenia skryptów Python, które były uruchamiane w środowisku SageMath.

Zakres godzinowy i tematyczny zajęć warsztatowych był różnorodny, były prowadzone w wielu grupach. Zajęcia zostały poddane ewaluacji. W ankietach ewaluacyjnych zostały zbadane poszczególne elementy warsztatów: trudność zagadnień, przystępność materiałów dla ucznia, przydatność SageMath z punktu widzenia wykorzystania jego narzędzi do prac domowych, projektów czy przyszłych zastosowań. Uczniowie wypowiadali się także temat sposobu przeprowadzenia zajęć. Wszyscy wypowiedzieli się za tym, aby więcej było zadań do samodzielnego wykonania w grupach.

Uwaga metodyczna:

Podczas warsztatów takie polecenia dla ucznia, które są w formie powtarzania poleceń wykonywanych przez nauczyciela nie mogą trwać długo, ponieważ uczniowie poczują się znudzeni. Takie zajęcia muszą być przeplatane aktywnym zadaniem dla ucznia, wymagającym od niego kreatywności. Uczniowie preferują pracę w grupach podczas warsztatów.

Metoda projektu, projekty grupowe

Spośród metod praktycznych stosowanych podczas zajęć szkolnych, na szczególną uwagę zasługuje metoda projektów. Aktywizuje ucznia do kreatywnych poszukiwań i rozwiązywania problemów, uczy współpracy i odpowiedzialności oraz dokumentowania i prezentowania wyników prac. Dlatego warto sięgać po tę metodę na każdym etapie edukacyjnym.

Potrzeby społeczne są niżej w piramidzie potrzeb i dlatego każda praca w grupie angażuje ucznia bardziej niż praca indywidualna, każdej pracy w grupie towarzyszą emocje, a emocje z kolei sprawiają, że uczenie się nabiera innego oblicza, dlatego uczniowie często nie określają swojej pracy w projekcie jako „uczenie się”.

Metoda projektów powstała w latach 20-tych, jako przeciwwaga do nauczania przedmiotowego i systemu klasowo-lekcyjnego Taki system nauczania zrywał z przedmiotowym układem, skupiał naukę z różnych dziedzin w jeden problem do rozwiązania zagadnień np. badawczych i wiązał działalność praktyczną z pracą intelektualną. Twórcą metody projektów był W. H. Kilpatrick, którego ideą było uczenie się przez działanie. Obecnie nauczanie zintegrowane, które jest wykorzystywane w wielu szkołach niepublicznych, nawiązuje do tej metody.

Założeniem metody projektów jest wdrażanie uczniów do twórczego i problemowego myślenia i działania. Pomaga przygotowywać uczniów do rozwiązywania realnych problemów, korzystania z różnorodnych źródeł informacji, pozwala dostrzegać związki pomiędzy różnymi dyscyplinami nauki, pomaga łączyć teorię i praktykę oraz myślenie i działanie - daje możliwość uczenia się za pomocą wielu aktywności.

- Metoda projektów stwarza pole do działań ucznia:
- rozpoznanie i opis sytuacji problemowej,
- formułowanie celów i zadań,
- kreatywność, generowanie pomysłów,
- integrowanie wiedzy z różnych przedmiotów nauczania,

- uruchamianie wyobraźni,
- odpowiedzialność, samodzielność,
- planowanie zadań, ocena złożoności i trudności zadań,
- wytrwałość w poszukiwaniu rozwiązań i realizacji zadań,
- samokształcenie,
- przygotowanie i prowadzenie publicznych wystąpień,

W zakresie zdobywania informacji:

- korzystanie z różnych źródeł informacji,
- analizowanie jakości informacji i ocena ich wiarygodności,
- klasyfikowanie przydatności informacji z punktu widzenia celów,
- wykorzystanie informacji zgodnie z prawem autorskim,
- prezentowanie informacji.

Projekty grupowe pozwalają dodatkowo kształtować umiejętności współdziałania:

- komunikowania się, (także elektronicznego),
- planowania i organizowania własnej pracy i pracy w grupie,
- wymiany zasobów, (np. elektronicznej)
- wyrażanie własnych opinii i korzystania z opinii wyrażanych przez innych członków grupy,
- rozwiązywanie konfliktów.

Założeniem metody projektów jest wdrażanie uczniów do twórczego i problemowego myślenia i działania. Polega na planowaniu i wykonywaniu przez uczniów zadań określonych w ramach projektu (np. w instrukcji do projektu), poprzez samodzielne poszukiwanie i rozwiązywanie problemów pod opieką nauczyciela. Opiera się na praktycznym działaniu: rozpoznawaniu problemów, stawianiu tez i pytań, dowodzeniu, poszukiwaniu odpowiedzi przez obserwacje, badania, analizy, obliczenia, symulacje, eksperymenty czy inne aktywności, np. działania lokalne, społeczne. Uczestnicy realizują temat projektu rozłożony w czasie, pracują samodzielnie lub w zespołach, czy grupach np. klasy, szkoły, z innych szkół czy krajów.

Przygotowanie przez nauczyciela projektu przedmiotowego (lub międzyprzedmiotowego) obejmuje:

- wybór zagadnienia do realizacji z wykorzystaniem metody projektów na podstawie analizy efektów kształcenia i ewentualnych możliwości podejmowania działań międzyprzedmiotowych,
- przygotowanie instrukcji dla uczniów, zawierającej: określenie celów, metod pracy, terminy realizacji poszczególnych etapów i całości, zadań uczniów, wymagań co do rezultatu pracy, sposobu prezentacji wykonanych zadań i kryteria oceniania,
- przygotowanie uczniów do pracy metodą projektów, szczególnie jeśli wcześniej nie wykonywali projektów, omówienie z uczniami zadań i wyników prac,

- motywowanie uczniów do zaangażowania się w projekt, podanie przykładów tematów projektów, badań wykonanych przez uczniów, odpowiedzi na pytania problemowe, pokazanie opisów projektów, prezentacji, sprawozdań czy filmów zrealizowanych przez innych uczniów.
- wprowadzenie uczniów w wybrane zagadnienie wzbudzenie ich zainteresowania, wskazanie możliwych do rozważenia problemów, przykłady narzędzi, które można użyć do realizacji projektu.
- przygotowanie planu doboru grup do realizacji projektów – nauczyciel wybiera sposób podziału na grupy, szczególnie jeśli chciałby zbalansować grupy według wybranego kryterium. Mogą to być:
- grupy jednorodne ze względu na wybrane kryterium np. osiągnięcia szkolne, aktywność, umiejętności lub zainteresowania,
- grupy o pełnym zróżnicowaniu - każda grupa ma pełny zbiór wg założonego kryterium,
- grupy koleżeńskie, chętnie wybierane przez uczniów, ale trudniejsze do zarządzania przez nauczyciela i niekiedy powodujące problemy integracyjne klasy,
- grupy doboru celowego lub zadaniowego,
- grupy według kolejności na liście klasy,
- grupy losowe

Metoda projektów wymaga od nauczyciela wcielenia się w nieco inną rolę. Z osoby dominującej, wyznaczającej tok pracy ucznia oraz głównego źródła informacji (szczególnie jeśli nauczyciel pracuje najczęściej metodami podającymi, mało zostawiając miejsca na aktywność i samodzielność uczniów) - nauczyciel powinien się zmienić w dyskretnego przewodnika, obserwatora i pomocnika. Warto tak zorganizować projekt, aby lwia część prac została wykonana jako praca domowa uczniów i poświęcić np. 15 minut kilku lekcji na pokaz postępów prac. Uczniowie mogą zaplanować wspólne spotkania w szkole, poza szkołą, albo wykorzystać techniki informacyjne i komunikacyjne. Nauczyciel monitoruje postępy realizacji projektu, zgłasza uwagi i doradza.

Z moich obserwacji wynika, że zarówno praca w grupach dwuosobowych, jak i praca w większych grupach jest przez uczniów bardzo chętnie podejmowana. Uczniowie lubią wyzwania, inspirują się wzajemnie, uczą się od siebie, poddają pomysły krytycznej ocenie, w grupie są bardziej aktywni i twórczy. Ale z punktu widzenia nauczyciela praca grupowa uczniów jest trudniejsza do przygotowania i zarządzania, wymaga wnikliwej analizy przy wyborze celów i przemyśleń sposobu ich realizowania.

Wielokrotnie namawiam do współpracy w projekcie międzyprzedmiotowym nauczycieli innych przedmiotów. Dopytuję ich, czy realizują projekty na swoich lekcjach i niestety z przykrością muszę stwierdzić, że nie jest to metoda chętnie wybierana przez nauczycieli. Na pytanie „dlaczego nie?", odpowiadają najczęściej, że „projekty zabierają wiele godzin, które powinni wykorzystać na realizację materiału" lub, że „projekty niczego nie uczą i na takie zabawy nie mają czasu" albo, że uczniowie znajdują jednego pracowitego ucznia w grupie, który wszystko zrobi a reszta nie robi nic, albo też, że nauczyciel przesuwa termin oddania projektu po raz kolejny, uczniowie się tłumaczą, że część projektu jeszcze nie jest gotowa, ponieważ ktoś był chory albo ma angielski po południu i nie mogli się spotkać. Nauczyciele uważają tę metodę za zbyt pracochłonną i trudną do realizacji. Dlatego niezbyt chętnie sięgają po metodę projektu. Rzeczywiście to niełatwe i wymaga wnikliwych przemyśleń, a sama metoda ma też wady i pułapki, czyhające zarówno na nauczyciela, jak i na uczniów.

Najczęściej podnoszone jest pytanie, czy metodę projektów da się zastosować do skutecznej realizacji obowiązkowego programu nauczania czyli zawartej w nim wiedzy (pojęć, faktów). Szczególnie w kontekście wielu godzin poświęconych na omawianie, wykonywanie i prezentowanie projektów. Wątpliwości budzi także mała skuteczność zdobywania wiedzy przez ucznia na podstawie prezentacji projektów wykonanych przez inne grupy czy innych uczniów.

Ale chyba największy problem jest taki, że duża część nauczycieli jest przywiązana do tradycyjnych metod nauczania i rzadziej wybiera metody aktywne podczas zajęć.

Na obronę metody projektów należy podkreślić, że większości zagrożeń da się uniknąć, jeśli się je zna.

W XXX III LO im. M. Kopernika w Warszawie metoda projektu została wykorzystana podczas zajęć informatyki. Projekty zostały wykonane w trzech grupach uczniów z klasy drugiej, tematem projektu było badanie funkcji.

Zaprezentowane zostały materiały w postaci instrukcji do projektu, opisu przeprowadzonych zajęć i przykładowych prac uczniów. Ponadto realizacja projektu w trzech grupach została zaplanowana tak, aby posłużyła do wykonania badań ewaluacyjnych porównujących wykorzystane metody. To badanie pokazało, że projekt został bardzo dobrze oceniony przez uczniów, uznali, że dużo się nauczyli przydatnych rzeczy i była to dla nich twórcze zadanie.

Uwaga metodyczna:

SageMath ma bardzo bogate możliwości, które mogą posłużyć nauczycielowi do zaplanowania zadań o szerszym charakterze, niż pojedyncza lekcja czy cykl lekcji. Można zaplanować długoterminowe prace o charakterze problemowym, kiedy uczniowie sami dochodzą do zbadania lub udowodnienia teorii, praw czy zasad. Sformułowane problemy, pytania, zagadnienia, łącznie z poznaniem teorii można zlecić uczniom jako tematy do odwróconych lekcji czy prac grupowych lub projektów indywidualnych. Ponadto w oddziałach, w których uczniowie znają język Python można zaplanować zagadnienia wymagające napisania skryptów, które pozwolą na realizację zaplanowanego algorytmu do rozwiązania problemu. W ten sposób można zrealizować wiele celów zarówno dotyczących realizacji materiału jak i dać uczniom okazję do kreatywności. Warto nabywać doświadczeń i w każdym kolejnym projekcie eliminować napotkane problemy. Zaś zdobywane przez ucznia umiejętności podczas pracy metodą projektów są ogromnie istotne w procesie nauczania jako całości.

2.3.4 Podsumowanie tego rozdziału

Nasze doświadczenia pokazują, że znajomość możliwości SageMath pozwala na zorganizowanie ciekawych zajęć zarówno lekcyjnych, jak i pozalekcyjnych, w formie warsztatów, pracy problemowej, w formie odwróconej lekcji czy projektów przedmiotowych, międzyprzedmiotowych, indywidualnych i grupowych. Jeśli nauczyciel chciałby urozmaicać metody dydaktyczne, sięgać po nowe technologie oparte na doświadczeniach innych nauczycieli aby stale rozwijać zainteresowania uczniów - z pewnością znajdzie w proponowanej metodyce i przygotowanych materiałach (rezultatach naszego projektu) cenne inspiracje wzbogacające jego warsztat pracy.

2.4 Potencjalne zastosowania SageMath

2.4.1 Zastosowania SageMath w matematyce

Matematyka uczy logicznego myślenia, wnioskowania i umiejętności precyzyjnego wysławiania się. Rozwijając swoje umiejętności matematyczne uczeń osiąga stawiane przed nim cele edukacyjne, przez co rozwija się również intelektualnie. Osiągnięcie przez ucznia założonych celów jest możliwe dzięki zastosowaniu na lekcjach matematyki takich metod nauczania i sposobów przekazywania wiedzy, które w jak największym stopniu uatrakcyjniają przedmiot oraz motywują uczniów do samodzielnego rozwiązywania problemów matematycznych.

W literaturze dydaktycznej został opisany szereg metod pracy z uczniem. W dzisiejszych czasach, kiedy praktycznie wszyscy uczniowie mają dostęp do komputera, tabletu lub smartfona, szczególnego znaczenia nabierają metody aktywizujące, które zakładają wykorzystanie tych urządzeń w procesie edukacyjnym.

Metodą znajdującą coraz szersze zastosowanie jest praca z komputerem, czyli odpowiednie zastosowanie technologii informacyjnej w celu zwiększenia efektów nauczania matematyki. Istnieje wiele pakietów matematycznych, które nauczyciel może wykorzystać podczas lekcji. Jednym z nich jest platforma SageMath - bezpłatny (open-source) system oprogramowania matematycznego oparty na języku programowania Python, wspomagający nauczanie matematyki. SageMath można zastosować we wszystkich działach programu nauczania matematyki w szkole średniej na poziomie podstawowym i rozszerzonym.

Zastosowanie SageMath w poszczególnych działach programu nauczania matematyki w szkole średniej przedstawimy na przykładzie programu autorstwa Marcina Kurczaba, Elżbiety Kurczab i Elżbiety Świdy „Matematyka. Program nauczania. Zakres rozszerzony.”

Tabela 2.1: Zakres potencjalnego zastosowania SageMath: matematyka klasa 1 liceum.

Logika. Zbiory. Zbiory liczbowe	Przy pomocy pakietu SageMath uczeń może poznać i utrwalić podstawowe prawa logiki, takie jak negacja alternatywy i negacja koniunkcji, ocenić wartość logiczną zdań złożonych oraz zastosować poznane prawa logiczne (patrz np.: *Działania na wyrażeniach logicznych.* Uczeń może również nauczyć się wyznaczać część wspólną, sumę i różnicę zbiorów oraz dopełnienie zbioru patrz np. zadanie *O graficznym i algebraicznym rozwiązywaniu nierówności* czy *Nierówności - graficznie!*
Działania w zbiorach liczbowych	Ponieważ SageMath może służyć jako kalkulator, uczeń wykorzystując ten pakiet może eksperymentować w większościa zagadnień. Wiele przykładów podano w rozdziale *Bogaty i szybki kalkulator naukowy.*

Continued on next page

Tabela 2.1 – continued from previous page

Wyrażenia algebraiczne	Ponieważ w SageMath można wykonywać obliczenia symboliczne, uczeń wykorzystując ten pakiet może osiągnąć wszystkie cele edukacyjne w tym dziale. Przykładami mogą być rozdział *Działania na wyrażeniach algebraicznych.* oraz zadania: • *O tym jak solve wykona pracę za nas i kiedy nas zawiedzie* • *Czworokąt wpisany w okrąg z wykorzystaniem algebry komputerowej*
Geometria płaska: • pojęcia wstępne • trójkąty • pole koła, pole trójkąta.	Przy pomocy środowiska SageMath uczeń może wizualizować obiekty geometryczne, badać ich wzajemne zależności oraz może rozwiązywać zadania dotyczące własności figur geometrycznych wykorzystując poznane w tym dziale twierdzenia. Przykładem mogą być zadania • *Czworokąt wpisany w okrąg z wykorzystaniem algebry komputerowej* • *Solve and substitute* Wykorzystując środowisko SageMath uczeń może utrwalić swoje umiejętności m. in. w zakresie korzystania z przybliżonych wartości funkcji trygonometrycznych, rozwiązywania zadań geometrycznych z wykorzystaniem funkcji trygonometrycznych kąta ostrego w trójkącie prostokątnym oraz zastosowania twierdzenia sinusów i twierdzenia cosinusów.
Trygonometria	SageMath jest niezwykle przydatny w zarówno obliczaniu wartości arytmetycznych funkcji trygonometrycznych jak i upraszczniu tożsamości, patrze np.: *Działania na wyrażeniach algebraicznych.*. Należy zwrócić uwagę na szczególną przydatność Sage w rysowaniu wykresów, zwracają uwagę na typowe problemy związane z rozbieżnościami - patrz: • *O tym jak nakłonić funkcję plot do współpracy* • *Jak skomplikować proste i popaść w kłopoty z funkcją tangens* • *O tożsamości trygonometrycznej i założeniach* • *O kątach i rysowaniu*
Funkcja i jej własności	Przy pomocy środowiska SageMath uczeń może osiągnąć wszystkie cele edukacyjne i zdobyć wszystkie umiejętności zakładane w tych działach. Typowymi cechami SageMath są rysowanie wykresów funkcji, obliczanie pochodnych i badanie przebiegu funkcji. Przykładami mogą być: • *Wizualizacja.* • *O tym jak technologia @interact pomoże rozwiązać równanie z parametrem* • *Pomocna ręka wykresu*

Continued on next page

Tabela 2.1 – continued from previous page

Przekształcanie wykresów funkcji	W sytemie algebry komputerowej jest bardzo łatwo napisać `plot(f(-x),(x,0,1))` i zastąpić znak minus np. pomnożeniem przez liczbę czy dodać wartość bezwzględną. Takie operacje są przydatne w budowaniu intuicji np. czy odejmując jeden od argumentu wykres przesunie się w lewo czy w prawo?

Tabela 2.2: Zakres potencjalnego zastosowania SageMath: matematyka klasa 3 liceum.

Klasa 3	
Funkcja wykładnicza i funkcja logarytmiczna	Uczeń wykorzystując środowisko SageMath może rozwiązywać zadania opisujące zjawiska fizyczne, chemiczne i biologiczne oraz modelować zagadnienia z kontekstem rzeczywistym posługując się funkcjami wykładniczymi i funkcjami logarytmicznymi. Przykładem może być zadanie - *O logarytmach i upraszczaniu*
Geometria analityczna	Z pomocą SageMath można wykonać precyzyjne rysunki, animacje i interakcje obejmujące zagadnienia omawiane w tym dziale. Przykłady: • *O kątach i rysowaniu* • *O prostopadłości i równoległości prostych* • *Punkty odbicia i działania na wektorach* • *Czworokąt wpisany w okrąg z wykorzystaniem algebry komputerowej* • *Solve and substitute*
Kombinatoryka i rachunek prawdopodobieństwa	Wykorzystując środowisko SageMath można skonstruować symulacje procesów losowych i przyprowadzić eksperymenty numeryczne. Przykładem mogą być materiały ze strony *http://visual.icse.us.edu.pl/szkola*: Podejmowanie decyzji w grupie oraz Paradoks Monty-Halla. Histogram został również wykorzystany w scenariuszu fale.
Elementy statystyki opisowej	System SageMath posiada wiele funkcji pozwalający na obliczanie statystycznych wielkości np. wartości średnie czy histogram.
Geometria przestrzenna	Wykorzystując możliwości graficzne pakietu SageMath uczeń może ułatwić sobie rozwiązywanie zadań z geometrii przestrzennej poprzez wizualizację obiektów geometrycznych opisywanych w tych zadaniach.
Elementy analizy matematycznej	Wszystkie zagadnienia omawiane w tym dziale uczeń może opanować i utrwalić wykorzystując możliwości jakie daje środowisko SageMath oraz język programowania Python.

Na szczególną uwagę zasługuje zastosowanie SageMath do: badania własności funkcji, rozwiązywania równań, nierówności i układów równań i nierówności, prostego modelowania matematycznego, badania własności figur płaskich („odkrywanie" twierdzeń), rozwiązywania zadań z geometrii przestrzennej, wykorzystania rachunku pochodnych do analizy zjawisk opisanych wzorami różnych funkcji (w tym rozwiązywania zadań optymalizacyjnych).

Zastosowanie SageMath w procesie nauczania matematyki może odbywać się na kilka sposobów:

1. Podczas zajęć w szkole
2. Prezentacja - za pomocą komputera i rzutnika multimedialnego nauczyciel prezentuje uczniom przygotowany wcześniej statyczny pokaz,
3. Prezentacja interaktywna - nauczyciel prezentuje uczniom dynamiczny pokaz (wykorzystując elementy interaktywne przygotowane przez siebie lub znalezione w Internecie),
4. Prezentacja interaktywna z udziałem uczniów - nauczyciel prezentuje dynamiczny pokaz, w którym niektóre czynności wykonują wybrani uczniowie,
5. Zajęcia w pracowni komputerowej - uczniowie pracują indywidualnie przy komputerach lub, w przypadku niewystarczającej liczby dostępnych komputerów, są podzieleni na niewielkie grupy.
2. Podczas pracy własnej ucznia w domu
3. Uczeń logując się do serwera SageMath pracuje z notatnikiem przygotowanym i udostępnionym mu przez nauczyciela, zawierającym teoretyczne opracowanie zagadnienie, które uczeń jest zobowiązany opanować (statycznie),
4. Uczeń po zalogowaniu się do serwera SageMath uzyskuje dostęp do notatnika, w którym - aby opanować określone zagadnienie - musi zapoznać się z teorią na ten temat, przeanalizować rozwiązane przykłady oraz modyfikować je w celu rozwiązania zadanych przez nauczyciela zadań.

Z wyborem metod nauczania ściśle wiąże się odpowiedni dobór form organizacji procesu edukacyjnego:

1. Praca w grupach - w oczywisty sposób przebiegająca głównie podczas lekcji w szkole, polegająca na podziale klasy na kilkuosobowe zespoły i przydzieleniu im problemu do rozwiązania.
2. Praca indywidualna - każdy uczeń pracuje samodzielnie, pod kierunkiem nauczyciela podczas lekcji lub samodzielnie w domu. Korzyści są następujące:
3. Uczeń może we właściwym dla siebie tempie samodzielnie uzyskać odpowiedzi na postawione pytania poprzez analizę problemu i potencjalnych metod jego rozwiązania,
4. Uczeń może w większym stopniu utrwalić zdobytą wiedzę,
5. Uczeń nabywa i utrwala umiejętność samodzielnego zdobywania wiedzy.

Praca indywidualna wyrabia też nawyk sumiennego wykonania powierzonego zadania, odpowiedzialności za siebie, za swoją wiedzę i umiejętności.

2.4.2 Zastosowania SageMath w fizyce

Fizyka w szkołach realizujących polską podstawę programową

Polska podstawa programowa do fizyki nie zawiera w ogóle słowa “komputer”. Nie formułuje ona żadnych bezpośrednich sugestii stosowania TI w procesie nauczania tego przedmiotu. Nie oznacza to jednak, że stosowanie komputerów w dydaktyce fizyki jest zabronione.

Spójrzmy na wymagania przekrojowe dla zakresu rozszerzonego w IV etapie edukacyjnym (LO), zawarte w części 12. podstawy programowej:

Oprócz wiedzy z wybranych działów fizyki, uczeń:

1) przedstawia jednostki wielkości fizycznych wymienionych w podstawie programowej, opisuje ich związki z jednostkami podstawowymi;

2) samodzielnie wykonuje poprawne wykresy (właściwe oznaczenie i opis osi, wybór skali, oznaczenie niepewności punktów pomiarowych);

3) przeprowadza złożone obliczenia liczbowe, posługując się kalkulatorem;

4) interpoluje, ocenia orientacyjnie wartość pośrednią (interpolowaną) między danymi w tabeli, także za pomocą wykresu;

5) dopasowuje prostą y = ax + b do wykresu i ocenia trafność tego postępowania; oblicza wartości współczynników a i b (ocena ich niepewności nie jest wymagana);

6) opisuje podstawowe zasady niepewności pomiaru (szacowanie niepewności pomiaru, obliczanie niepewności względnej, wskazywanie wielkości, której pomiar ma decydujący wkład na niepewność otrzymanego wyniku wyznaczanej wielkości fizycznej);

7) szacuje wartość spodziewanego wyniku obliczeń, krytycznie analizuje realność otrzymanego wyniku;

8) przedstawia własnymi słowami główne tezy poznanego artykułu popularnonaukowego z dziedziny fizyki lub astronomii.

Poza punktem pierwszym i ostatnim, wymagania te idealnie wpisują się w możliwości oferowane przez środowisko SageMath. Python w środowisku SageMath oferuje wspaniałe możliwości graficzne. Format dowolnych elementów wykresów może być programowo zmieniany w szerokim zakresie. Same dane do wykresów mogą pochodzić ze skomplikowanych obliczeń na dużych zbiorach danych, do przeprowadzania których Python nadaje się bez porównania lepiej niż sugerowany przez podstawę programową kalkulator prosty. Przykładowo, użycie pętli do obliczeń zwalnia ucznia z mozolnych, nudnych i mało twórczych, wielokrotnie powtarzanych czynności. Zaoszczędzony czas uczeń może poświęcić na myślenie koncepcyjne, twórczą zabawę z parametrami i obserwację skutków tej zabawy.

Jedną z najczęściej wykorzystywanych na lekcjach fizyki opcji Pythona jest dopasowywanie nie tylko prostej, ale też dowolnej krzywej do punktów pomiarowych. Podstawa programowa zakłada, że uczeń powinien umieć “na oko” tak przyłożyć linijkę do wykresu, aby jak najlepiej wpasować się w widoczny na wykresie trend, po czym narysować prostą i z rysunku określić jej parametry liczbowe. Bezpośrednie stosowanie metody najmniejszych kwadratów w celu obliczenia współczynników prostej daleko wykracza poza wymagania programowe (nie wspominając już o odniesieniu jej do krzywych). Tymczasem w Pythonie wystarczą dwie proste linijki kodu, aby dopasować do danych dowolną krzywą. Uczeń wcale nie musi rozumieć, jak program to robi - dostaje skuteczne narzędzie do precyzyjnego wyliczenia wszystkich potrzebnych współczynników.

Kolejną często wykorzystywaną opcją jest szacowanie błędów pomiarowych. Uczniowie wprawdzie znają pojęcie odchylenia standardowego i procedurę jego obliczania, niemniej jest to żmudne i mało twórcze. Python pozwala rozwiązywać takie problemy jedną komendą.

Największe pole do nieskrępowanej twórczości naukowej Python daje chyba w dziedzinie symulowania i animowania zjawisk fizycznych. Pewne efekty z pogranicza matematyki i fizyki są bez komputera zupełnie nie do ogarnięcia. Na przykład licealiście trudno "na słowo" uwierzyć, że suma dwóch sinusów o mało różniących się okresach daje w efekcie dudnienia. W SageMath możemy to łatwo pokazać, umożliwiając dodatkowo łatwą interaktywną zmianę amplitud, częstotliwości i przesunięć fazowych za pomocą suwaków. Podobnie łatwo możemy pokazać, czym jest transformata Fouriera. Zabawa suwakami czy kodem źródłowym pozwala uczniom łatwo przyswoić sobie sens pojęć, których wytłumaczenie za pomocą czystego wykładu z matematyki byłoby kompletnie niemożliwe, nieskuteczne i bezcelowe. Python umożliwia o wiele szersze stosowanie matematyki w fizyce i bezpośrednie oglądanie efektów swojej pracy na żywo.

Fizyka w szkołach realizujących Program Dyplomowy Matury Międzynarodowej

Fizyka nauczana w klasach realizujących Program Dyplomowy Matury Międzynarodowej jest dziedzina nauki, w której zastosowanie technik komputerowych jest bezwzględnie wymagane przez komórkę nadzorującą realizację programu i zostało wskazane przez International Baccalaureate Organization do dokumentu odpowiadającego Polskiej Podstawie Programowej, zwanego International Baccalaureate Physics Higher and Standard Level Core.

Komputer, jako narzędzie pracy stanowi niezbędne narzędzie pracy fizyków i jest bezpośrednio stosowany przez uczniów na zajęciach teoretycznych jak i laboratoryjnych. Dzięki metodom informacyjnym możliwy jest dokładny pomiar wielkości fizycznych, ich zapis „on line" a co za tym idzie, natychmiastowa analiza wyników. Uzyskane wyniki mogą być łatwo prezentowane w formie graficznej stanowiącej postać wykresów, które mogą być wykonywane przy użyciu oprogramowania Sage. Zaletą tego języka jest możliwość łatwej zmiany skali lub układu osi w celu przejrzystego prezentowania badanej zależności. Obecnie wzrasta ranga środowiska Sage, szczególnie wśród nauczycieli fizyki w klasach realizujących Program Dyplomowy Matury Międzynarodowej. Wynika to bezpośrednio z faktu, iż możliwości wykorzystania notebooka na lekcjach fizyki są większe i bardziej atrakcyjne niż na innych przedmiotach wchodzących w skład dekagonu International Baccalaureate Dipploma Programme.

Do najbardziej rozpowszechnionych funkcji SageMath w nauczaniu fizyki należą: obliczenia i analiza wyników eksperymentalnych modelowanie pomiaru w doświadczeniach fizycznych, symulacja eksperymentów i procesów fizycznych, a rzadziej animacja lub modelowanie graficzne. W celu wykorzystania komputera jako przyrządu fizycznego opracowano wiele wersji oprzyrządowania informatycznego i fizyczne komputera oraz programy, które pozwalają wielokanałowo wykorzystywać komputer jako przyrząd. Może on pełnić rolę oscyloskopu z pamięcią, który jest bardzo skomplikowany, a koszty jego zakupu mogą stanowić nie lada wyzwanie dla budżetu szkoły. Może on pełnić rolę dokładnego stopera, termometru, dowolnego miernika elektrycznego, światłomierza, miernika kąta, symulatora rozpadu promieniotwórczego a nawet wiernie odtwarzać pracę elektrowni jądrowej. Jednak wszystkie tego typu aplikacje nie umożliwiają uczniowi samodzielnej konstrukcji a co za tym idzie zrozumienia metodologii rozwiązywania omawianych procesów i zjawisk fizycznych.

Dalszą aplikacją wykorzystania SageMath na lekcjach fizyki jest automatyczne prowadzenie pomiarów "on line" w czasie doświadczeń za pomocą przetworników analogowo - cyfrowych i opracowywanie danych w środowisku SageMath na bieżąco, zarówno przez uczniów jak i nauczyciela. Przykładem tego typu zastosowania jest wykonanie doświadczenia ujętego podstawą programową (Core) polegającego na badaniu słuszności prawa stygnięcia. Trudna interpretacja matematyczna polegająca na wprowadzeniu równań różniczkowych nie objętych programem podstawy matematyki zostaje zamieniona przez grupę badawczą na przykład w interpretację numeryczną, łatwą do wykonania w Sage. Z drugiej strony środowisko to umożliwia szybkie rozwiązanie równania różniczkowego, bez nadmiernej analizy teoretycznych aspektów samego rozwiązania. Środowisko SageMath pozwala również na szybką obróbkę wielkości mierzonych w układzie doświadczalnym i przejrzystą prezentację obliczonych wielkości pochodnych w postaci tabel, grafów lub wykresów. Zwalnia to uczniów od żmudnej i kłopotliwej pracy obliczeniowej, nieistotnej dla zrozumienia problemu, pozwala natomiast skoncentrować uwagę na fizycznej treści analizowanych zjawisk. Wyniki otrzymane w kilku seriach pomiarowych można opracować w sposób statystyczny, co pozwala ocenić na ile określona metoda i przyrządy są dokładne. Najbardziej okazałym przykładem zalecanym podczas realizacji nauczania fizyki w Programie Dyplomowym Matury Międzynarodowej może być zestaw ćwiczeń z mechaniki podczas którego uczniowie sporządzają wykresy ruchów, rozwiązują kinematyczne równanie ruchu, czy też wyznaczają maksymalną wysokość na którą może wznieść się ciało. W tym przypadku szybkość prowadzenia obliczeń jest dość istotna, gdyż przeważnie dotychczas na lekcjach ograniczonych czasem na wykonanie kilku serii pomiarowych i dokonanie obliczeń, nie wspominając już o analizie i wyciągnięciu wniosków z doświadczenia często rezygnowano z możliwości programowania bezpośrednio przez ucznia ścieżki rozwiązania. Szybkość obliczeniową SageMath można wykorzystać do rozwiązywania problemów bardzo skomplikowanych np. ruch wirującego bąka czy też bardziej skomplikowanych obliczeń z zakresu fizykochemii ciała stałego, stanowiących część podstawy programowej w klasach z Programem Dyplomowym Matury Międzynarodowej. Wykonując doświadczenia przy pomocy SageMath uczniowie mogą sami ocenić, że stosunkowo szybko i sprawnie przeprowadzić nawet bardzo skomplikowane obliczenia.

Nadrzędnym celem modelowania procesów fizycznych jest wyrobienie poglądu na ich strukturę wewnętrzną (powiązania pomiędzy poszczególnymi elementami), poznanie praw rządzących ich przebiegiem, wyjaśnienie przyczyn występowania zjawisk towarzyszących określonemu procesowi, a także możliwość przewidywania odpowiedzi procesu na dowolne warunki. Osobnym zagadnieniem jest pojęcie modelu wymaganego w procesach poznawczych według zaleceń International Baccalaureate Organization, przez który rozumieć należy materialnie zrealizować układ, który, odzwierciedlając lub odtwarzając przedmiot badania, zdolny jest zastępować go tak, że jego badanie dostarcza nam nowej informacji o tym przedmiocie. Modele matematyczne procesów fizycznych konstruuje się na podstawie ogólnych zasad i praw fizyki poznawanych w cyklu nauczania. Przyjmuje się przy tym pewne założenia upraszczające strukturę wewnętrzną modelowanych procesów oraz ich powiązania z otoczeniem. Skonstruowany model może dostarczyć określonej wiedzy o modelowanym procesie, jeżeli jest modelem zasadnym dla danego procesu, to znaczy, jeśli odwzorowuje on przebieg procesu z zadowalającą dokładnością. Dysponując modelem zasadnym dla danego procesu fizycznego oraz wartościami wszystkich jego parametrów, można na tej podstawie przewidywać odpowiedz procesu na dowolne wymuszenie. Możliwość ta ma istotne znaczenie dla symulacji przebiegu procesów fizycznych. W nauczaniu fizyki w szkole korzysta się z gotowych, zasadnych modeli matematycznych dla analizowanych w czasie lekcji procesów fizycznych. Problemem, który pozwala zniwelować użycie SageMath jest poziom abstrakcji matematycznej, często niezrozumiały przez większość uczniów, na przykład zagadnienie warunków brzegowych, czy ograniczeń wynikających z zasięgu oddziaływań. Dostępne są więc wszystkie informacje, niezbędne do symulowania z wykorzystaniem komputera, przebiegu analizowanych procesów fizycznych.

Możliwość modelowania zjawisk fizycznych za SageMath, analizującego dane zjawisko metodą numeryczną ”krok po, kroku” należy do najcenniejszych z dydaktycznego punktu widzenia. W przeciwieństwie do opisu analitycznego model numeryczny kieruje uwagę bardziej na sposób rozwiązania problemu niż na formę rozwiązania. To pozwala łatwiej dostrzec związek między różnymi fenomenologicznie zjawiskami fizycznymi, ale opisywanymi przez podobne modele np. w takich zjawiskach jak: wymuszone drgania ciężarka na sprężynie, rozładowanie kondensatora, Modelowanie ukazuje związki pomiędzy procesem, a jego modelem matematycznym, natomiast symulacja dotyczy związków między modelem matematycznym procesu fizycznego i procesem mu równoważnym, przeprowadzonym na podstawie tego modelu na komputerze. Modelowanie ukazuje związki pomiędzy procesem, a jego modelem matematycznym, natomiast symulacja dotyczy związków między modelem matematycznym procesu fizycznego i procesem mu równoważnym, przeprowadzonym na podstawie tego modelu na komputerze. Modelowanie ukazuje związki pomiędzy procesem, a jego modelem matematycznym, natomiast symulacja dotyczy związków między modelem matematycznym procesu fizycznego i procesem mu równoważnym, przeprowadzonym na podstawie tego modelu na komputerze.d płynący w obwodzie RLC. Komputerowa symulacja procesów i doświadczeń fizycznych stanowi rozszerzenie zagadnienia modelowania. Modelowanie ukazuje związki pomiędzy procesem, a jego modelem matematycznym, natomiast symulacja dotyczy związków między modelem matematycznym procesu fizycznego i procesem mu równoważnym, przeprowadzonym na podstawie tego modelu na komputerze. Określenie związków pomiędzy procesem fizycznym, a procesem mu równoważnym nazywa się identyfikacją procesu fizycznego. Symulacja komputerowa procesu fizycznego ma więc na celu wytworzenie wiarygodnej odpowiedzi procesu na dane wymuszenie i w przypadku pisania samodzielnie przez ucznia komend języka programowania pozwala na zagłębienie się w istotę procesu a co za tym idzie jego głębsze zrozumienie.

W oddziałach realizujących Program Dyplomowy Matury Międzynarodowej w ramach określonej przez International Baccalaureate Organization 4-ej grupy przedmiotowej, istnieje bardzo wiele obszarów, w obrębie których zarówno ze strony podmiotu nauczania jakim jest uczeń jak i współtworzącego proces nauczania nauczyciela pełniącego rolę swoistego tutora, oprogramowanie jakim jest Python i SageMath jest wprost pożądane jako element zastosowania metod numerycznych i statystycznych w rozwiązywaniu problemów fizycznych. Poniższe zestawienie jest wypracowanych zbiorem opisanych powyżej zastosowań, które znalazły aplikację w trakcie realizacji zajęć z zakresu Physics Standard/Higher Level w XXXIII Liceum Ogólnokształcącym Dwujęzycznym im. Mikołaja Kopernika w Warszawie:

Tabela 2.3: Zakres potencjalnego zastosowania SageMath w fizyce z przykładami.

Przedmiot	godz.	Potencjalne użycie Python/SageMath
Measurements and uncertainties: • 1.1 - Measurements in physics • 1.2 - Uncertainties and errors • 1.3 - Vectors and scalars	5h	Obliczanie niepewności pomiarowych, operacje na wektorach. Przykładem może być scenariusz lekcji „Badanie ruchu przyśpieszonego" przyspieszony. Operacje na wektorach mogą być efektywnie realizowane w SageMath wykorzystując podsystem algebry liniowej. Do dyspozycji są funkcje wizualizujące `vector_plot` i `arrow`.
Mechanics: • 2.1 - Motion • 2.2 - Forces • 2.3 - Work, energy and power • 2.4 - Momentum and impulse	22h	Sporządzanie wykresów zależności drogi, szybkości i przyśpieszenia od czasu. Modelowanie torów ruchu - wykorzystanie funkcji `parametric_plot`. Rozwiązywanie kinematycznego równania ruchu metodami numerycznymi. Przykładem takiego podejścia może być modelowanie rzutu ukośnego: • http://visual.icse.us.edu.pl/szkola/rzut_ukosny.html.
Thermal physics: • 3.1 - Thermal concepts • 3.2 - Modelling a gas	11h	Modelowanie stanu gazu doskonałego. Sporządzanie wykresów w dowolnej przemianie gazowej. W dużą rolę odgrywa tu rysowanie wykresów funkcji np. rozdział *Wizualizacja*.
Waves: • 4.1 - Oscillations • 4.2 - Travelling waves • 4.3 - Wave characteristics • 4.4 - Wave behaviour • 4.5 - Standing	15h	Sporządzanie zależności wychylenia, prędkości i przyspieszenia danego ciała w ruchu drgającym, modelowanie fali poprzecznej i podłużnej. Superpozycja w ruchu harmonicznym. dobrym przykładem jest scenaiusz fale oraz przykład paczki falowej z *Wizualizacja.*.

Continued on next page

Tabela 2.3 – continued from previous page

Przedmiot	godz.	Potencjalne użycie Python/SageMath
Electricity and Magnetism: • 5.1 - Electric fields • 5.2 - Heating effect of electric currents • 5.3 - Electric cells • 5.4 - Magnetic effects of electric currents	15h	Rozwiązywanie równań opisujących obwód prądu za pomocą pierwszego i drugiego prawa Kirchhoffa. Równania takie można rozwiązywać za pomocą algebry liniowej lub algebry symbolicznen wykorzystując funkcję `solve`, zob.:. . • *O graficznym i algebraicznym rozwiązywaniu nierówności* • *O logarytmach i upraszczaniu* • *Solve and substitute* • *Czworokąt wpisany w okrąg z wykorzystaniem algebry komputerowej* Modelowanie pola wektorowego może być zilustrowane funkcjami `vector_plot` i `arrow`.
Circular motion and gravitation: • 6.1 - Circular motion • 6.2 - Newton's law of gravitation	5h	Numeryczne rozwiązywanie równania różniczkowego opisującego II Zasadę Dynamiki Newtona dla dowolnego punktu materialnego w danym polu wektorowym. Można rozszerzyć metody zaprezentowane w http://visual.icse.us.edu.pl/szkola/rzut_ukosny.html.
Atomic, nuclear and particle Physics: • 7.1 - Discrete energy and radioactivity • 7.2 - Nuclear reactions • 7.3 - The structure of matter	14h	Modelowanie krzywych zaniku promieniotwórczego. Obliczanie czasów półtrwania izotopów. Przydatnym narzędziem może okazać się zarówno numeryczne jak i algebraiczne rozwiązywanie równań różniczkowych: `desolve`. Pomocna może okazać się wizualizacja funkcji *Wizualizacja.* z rozdziału *SageMath w pigułce*.
Energy production: • 8.1 - Energy sources • 8.2 - Thermal energy transfer	8h	Sporządzanie wykresów zależności długości emitowanej fali elektromagnetyczne j od temperatury: prawo Viena. Prawo zaniku wykładniczego temperatury. Pomocna może okazać się wizualizacja funkcji *Wizualizacja.* z rozdziału *SageMath w pigułce*.
Wave phenomena: • 9.1 - Simple harmonic motion • 9.2 - Single-slit diffraction • 9.3 - Interference • 9.4 - Resolution • 9.5 - Doppler effect	17h	Model zależności natężenia światła od kąta ugięcia dla pojedynczej szczeliny, podwójnej szczelny (doświadczenie Younga) oraz dla układu szczelin. Można łatwo rozszerzyć przykład paczki falowej z *Wizualizacja.* do pokazania tych zjawisk.

Continued on next page

Tabela 2.3 – continued from previous page

Przedmiot	godz.	Potencjalne użycie Python/SageMath
Fields: • 10.1 - Describing fields • 10.2 - Fields at work	11h	Modelowanie pola wektorowego. W rozdziale *Wizualizacja.* znajduje się przykład wizualizacji pola wektorowego. Należy zwrócić uwagę, ze rozszerzenie tego przykładu na np. pole pochodzące od układu dipoli jest stosunkowo proste, stosując technikę analogiczną do tej przy rysowaniu paczki falowej.
Electromagnetic induction: • 11.1 - Electromagnetic induction • 11.2 - Power generation and transmission • 11.3 - Capacitance	16h	Sporządzanie zależności strumienie wektora indukcji magnetycznej od czasu i obliczanie pierwszej pochodnej (prawo indukcji Faradaya). Zob. przykład oblicznia pochodnej w zadaniu *Ławiej powiedzieć niż zrobić!*.
Quantum and nuclear Physics: • 12.1 - The interaction of matter with radiation • 12.2 - Nuclear physics	16h	Modelowanie kształtów orbitali atomowych. Możliwe jest wykorzystanie `implicit_plot3d` do narysowania orbitali atomowych.

2.4.3 Projekty międzyprzedmiotowe

Uwagi ogólne

Narzędzia SageMath, w szczególności wzbogacone skryptami w Python, dają ogromne możliwości do wykorzystania w realizacji projektów międzyprzedmiotowych. Wszelkie symulacje, badania teorii i zjawisk, obliczenia, analiza i prezentacja pomiarów, rozwiązywanie problemów eksperymentalnych i badawczych uczeń może wykonać za pomocą darmowego narzędzia, dostępnego on-line - wręcz na swoim smartfonie. Poza omówionymi wcześniej matematyką i fizyką, w zasadzie każdy przedmiot, którego uczący nauczyciel chciałby zaangażować uczniów do głębszej analizy, zainteresowania przedmiotem, wykorzystania ciekawości świata uczniów, może zaplanować problem, którym zaciekawi uczniów. Mogą to być projekty badawcze, koncepcyjne lub praktyczne.

Nauczyciele organizujący projekt międzyprzedmiotowy odpowiadają na pytania:

1. Jakie cele chcemy osiągnąć?
2. Jakiego typu to będzie projekt (badawczy, koncepcyjny, praktyczny)?
3. Jaki będzie temat projektu?
4. Jakie zagadnienia zostaną włączone do realizacji?

5. Na jakie pytania uczniowie mają
6. odpowiedzieć, jakie działania zaplanować?
7. Jakie mają być rezultaty projektu?
8. Jak uczniowie będą dokumentować wykonane działania?
9. Czy uczniowie mają postawić tezę i ją udowodnić (bądź obalić)?
10. Czy zaplanować eksperyment, symulacje, czy badania, co badać, mierzyć, obliczać?
11. Jakie będą ramy czasowe projektu?
12. Czy projekty będą indywidualne, czy zespołowe?
13. W jaki sposób uczniowie zostaną wybrani do zespołów?
14. Czego uczniowie nauczą się w trakcie realizacji projektu?
15. Co będzie w instrukcji do projektu?
16. Jak sprawdzimy wiedzę po realizacji projektu?
17. Jakie warunki muszą być zapewnione do realizacji projektu?
18. Kiedy będą konsultacje?
19. Jak będzie oceniany projekt?
20. Jak zorganizujemy prezentację rezultatów projektu?

Projekty badawcze

W ramach projektów badawczych uczniowie przeprowadzają badania teoretyczne lub praktyczne. Zbierają, porządkują i systematyzują informacje na zadany temat. Opracowują plan badawczy, może nowe koncepcje rozwiązań (badają możliwości wprowadzenia tych rozwiązań). Efektem takich projektów mogą być: publikacje wyników badań (np. strony internetowe, prezentacje), wystawy. Przykłady grup projektów:

Matematyka + informatyka

Projekt Badanie funkcji

Projekt polega na zbadaniu własności wybranej funkcji i zaprezentowaniu badań w postaci wykresów i tabel. Tego typu projekt może być zorganizowany w wielu wariantach. Nauczyciel może sam wybrać funkcje dla uczniów, np stosując zasadę im lepsza ocena z matematyki, tym trudniejsza funkcja do zbadania, może także przygotować koperty do rozlosowania, bądź wybór funkcji przez uczniów uczynić jednym z elementów projektu. Badanie może być wykonywane w grupach bądź indywidualnie. Może się zakończyć opublikowaniem wyników badań na serwerze SageMath lub wykonaniem dokumentu stanowiącego sprawozdania z badań, strony internetowej czy prezentacji.

Ten projekt został wykonany w pięciu oddziałach, zewaluowany i opisany w rozdziale 4 w części "Dobre praktyki:". Wypracowana w ten sposób metodyka prezentuje kolejne kroki przygotowania i organizacji projektu, wskazówki dla nauczyciela, etapy realizacji w postaci cyklu zajęć oraz rezultaty w postaci prac uczniów.

Jest to materiał gotowy do zastosowania. Instrukcja do projektu znajduje się w materiałach dla nauczyciela. W materiałach znajdują się także przykładowe prace uczniów, zarówno w postaci publikacji na serwerze, jak i w postaci skanów drukowanych dokumentów wykonanych przez uczniów z użyciem pakietu Latex.

Informatyka + geografia

Projekt Losowy punkt na Ziemi

Jedna grupa pisze skrypt (w Python), który losuje dwie liczby - szerokość i długość geograficzną. (Oczywiście losowanie można zorganizować inaczej). W ten sposób zostaje wylosowany punkt X na Ziemi. jeśli będzie to punkt na oceanie uczniowie losują jeszcze raz, (chyba, że projekt będzie dotyczył mórz i oceanów). Uzyskany w ten sposób punkt, lub grupa punktów stanowią dane do tematów:

- Zaplanuj trasę z punktu startowego S do docelowego (może to być punkt S) przez wylosowane punkty, tak, aby pokonać jak najkrótszą drogę i odwiedzić wszystkie punkty (np. żadną trasą nie podróżować dwa razy). Udowodnij, że to najlepszy wariant. Opisz trasę, środki transportu, czas podróży, koszt transportu. To algorytmiczny problem komiwojażera. Losujemy tyle punktów aby zadanie nie przerosło uczniów.
- Zaplanuj wyprawę z Warszawy do punktu X. Wykonaj symulację podróży. Sprawdź możliwości rezerwacji, Zaproponuj alternatywnie etapy, czas, rodzaje transportu, noclegów, wyżywienia, biorąc pod uwagę koszt, przygotowania, wyposażenie, zagrożenie i inne elementy i czynniki dla wyprawy. Sporządź wykresy pogodowe dla całego czasu wyprawy. wykres zmian temperatur. Dołącz galerię map i album zdjęć.
- Zaplanuj wyprawę z punktu A do B. Zaplanuj warianty podróży zależnie od tego, co warto zobaczyć/zwiedzić po drodze. Dołącz wykresy kosztów, temperatur, czasu podróży, galerię map i album zdjęć.

Ten projekt został zrealizowany w wielu oddziałach, uczniowie tworzyli strony internetowe ze sprawozdaniami z wirtualnych podróży.

Fizyka, matematyka + język angielski

Wykorzystanie rezultatów pozwala na rozwój umiejętności leksykalnych w obrębie języka angielskiego inżynieryjno-technicznego. Współczesne formy wykorzystania w pracy ucznia przedmiotów matematyczno-przyrodniczych elementów technologii informacyjnej i komunikacyjnej nie stanowi tylko prawnego obowiązku, ale przede wszystkim ma za zadanie urozmaicać przygotowanie do konkursów/olimpiad i stanowić w jej odbiorze użyteczne narzędzie oraz element korygujący odbiór toku logicznego zajęć.

Może to być wykorzystywane jako forma pracy z uczniem zdolnym. Uczeń - olimpijczyk nie może wyobrażać sobie pracy, bez użycia współczesnych, co warte podkreślenia, aktualnie kształconym pokoleniom, technik informatycznych. Na bieżąco musi doskonalić swoje umiejętności obsługi programów komputerowych i języków programowania. Choć wiele realizowanych tematów będzie mu nieznanych, ponieważ dopiero w trakcie studiów magisterskich lub doktoranckich realizuje się zajęcia z języków programowania użytkowanych w chemii i fizyce, to jednak nabycie nowych umiejętności aplikacji komend językowych do rozwiązywania problemów natury rachunkowej, niedostępnych uczniom realizujących polską podstawę programową matematyki, nawet w zakresie rozszerzonym pozostaje wciąż w ich gestii.

W codziennej, szkolnej pracy, uczeń wykorzystuje technologię komputerową głównie do tworzenia różnorodnych testów użytkowych, diagnoz, narzędzi badawczych i ankiet. Własny warsztat pracy jest uporządkowany i na bieżąco udoskonalany. Wszystkie te programy i aplikacje charakteryzuje stosunkowo intuicyjna i oparta o proste relacje metoda ich konstrukcji, a co za tym brak możliwości zmian i wykonywania w nich innych niż umożliwione przez producenta operacji. Zarówno z punktu widzenia nauczyciela, jak i ucznia, będącego podmiotem procesu nauczania nie bez znaczenia pozostaje możliwość rozwiązywania problemów określonych w podstawie programowej oraz wykraczających poza jej ramy, w oparciu o szerokie spektrum metod znajdujących zastosowanie zarówno z punktu widzenia uczącego, jak i nauczanego. Olimpijczyk podejmujący analizę zagadnień obecnej oraz powstającej na skutek planowanej reformy systemu oświaty podstawy programowej przedmiotów takich jak fizyka i chemia i napotyka na wiele treści, których zrozumienie wymaga zastosowania złożonego rachunku z elementami analizy matematycznej. Zajęcia te zdecydowanie ułatwią mu pozyskiwanie wiedzy olimpijskiej z użyciem współczesnych metod rozwoju wiedzy.

Realizacja projektu pozwoliła też (co nie było pierwotnym zamierzeniem pomysłodawców całego przedsięwzięcia) na integrację między przedmiotami fizyka i język angielski. Uczniowie na lekcjach fizyki lub w domu wykonywali pomiary, których rezultaty należało później opracować w formie sprawozdania o z góry ustalonej strukturze. Pomiary dotyczyły następujących zagadnień:

- Badanie drgań struny
- Pomiar przyspieszenia w ruchu jednostajnie przyspieszonym
- Analiza zderzeń ślizgaczy na torze powietrznym.

Zasadniczo uczniowie na moich lekcjach wykonują dużo doświadczeń i mają już wprawę w pisaniu sprawozdań (wymagania co do formy i treści sprawozdania opublikowane są w formie PDF na stronie internetowej szkoły). Te trzy tematy włączyłem jednak do projektu, a uczniom poleciłem opracować sprawozdania nie w formie papierowej, tylko jako notatniki SageMath. Dzięki temu wszystkie obliczenia, wykresy i komentarze można było zrealizować za pomocą jednego wygodnego narzędzia.

Ponieważ projekt jest międzynarodowy i jego rezultaty muszą zostać przetłumaczone, poszedłem o krok dalej. Porozumiałem się z anglistami uczącymi w danych klasach i ogłosiłem, że najlepsze sprawozdania “przechodzą do finału” a ich autorzy zyskują przywilej przetłumaczenia swoich prac na język angielski, za co otrzymają dodatkowe punkty zarówno z fizyki, jak i z języka angielskiego. Warunkiem było uzyskanie pełnej akceptacji dokonanego tłumaczenia zarówno przez anglistę, jak i przeze mnie (sprawdzałem, czy tłumaczenie nie wypaczyło sensu merytorycznego opracowania). Faktycznie, uczniowie najwyżej ocenieni za polską wersję sprawozdania z radością przetłumaczyli swoje prace, a ich kopie opublikowałem na serwerze jako część końcowego raportu.

Wszyscy angliści byli bardzo zadowoleni ze współpracy. Oprócz standardowego języka angielskiego, prowadzą oni w klasach ścisłych przedmiot o nazwie język angielski techniczny. Udział w projekcie był dla nich okazją, by zastosować nauczane słownictwo i zwroty w żywym tekście naukowym nadzorowanym przez fizyka.

Projekty koncepcyjne

W ramach takich projektów uczniowie wykonują modele umożliwiające przeprowadzenie symulacji działania w rzeczywistej sytuacji. Symulacje w matematyce, fizyce, przedsiębiorczości, biologii, socjologii i innych dziedzinach.

Informatyka

Grafy: Zaplanuj model komunikacji lub zaopatrzenia w wodę w danej miejscowości.

Ten projekt został zrealizowany, uczniowie poprawiali drogi w województwie mazowieckim.

Przedsiębiorczość + informatyka

Projekt Symulacja inwestycji kapitałowych

Uczniowie losują wirtualny kapitał w kwocie X z kraju Y. Ich zadaniem jest wykonanie symulacji inwestycji zadanych części kapitału na różne sposoby. Np. 10% kapitału w lokaty bankowe, 20% w fundusze powiernicze, obligacje skarbu państwa, 30% giełda papierów wartościowych, 40% inwestycja własna (z wyłączeniem hazardu, loterii, nielegalnej działalności itp). Symulacja obejmuje np. jeden rok (warunki ustalone). Uczniowie sporządzają sprawozdania z poszczególnych inwestycji, wyjaśniają pojęcia, sporządzają wykresy i obliczenia, podsumowanie i wnioski.

Ten projekt był realizowany przez wiele lat, w klasach drugich i trzecich różnych profili. Instrukcja do projektu, przykładowe prace uczniów, wskazówki metodyczne dla nauczyciela znajdują się w materiałach.

Informatyka + biologia

Symulacja zmian w populacji ludzi przy zadanych warunkach lub wybranych gatunków zwierząt (lisy i króliki).

Projekty praktyczne

Uczniowie opracowują rozwiązania problemów praktycznych dla szkoły lub społeczności lokalnej, Przykłady tematów: przygotowanie pomocy dydaktycznych, plan wysadzenia krzewów na szkolnym terenie, zorganizowanie zawodów szachowych, zorganizowanie warsztatów "Przykłady wykorzystania SageMath w matematyce" dla młodszych kolegów.

Uwagi metodyczne do realizacji metody projektów

Najlepiej podjąć decyzję o wprowadzeniu metody projektów już na początku semestru czy roku szkolnego. Przy braku doświadczeń z wykorzystaniem tej metody, warto rozpocząć projekty od niewielkich przedsięwzięć, czyli jednej godziny lekcyjnej, potem cyklu zajęć, aby nabrać wprawy w stosowaniu metody.

Podział uczniów na zespoły można zaplanować na wiele sposobów, najlepiej zadbać o to, aby w każdym zespole znaleźli się uczniowie o różnym poziomie osiągnięć szkolnych. Grupy pracują sprawnie, gdy liczą 2, 4 osoby. Ważna jest pomoc nauczyciela przy wspólnym opracowaniu podziału ról i terminów. Kolejne kroki i ustalenia w zespole, nad którymi czuwa nauczyciel, prosząc uczniów o utworzenie notatki z każdego spotkania grupy podczas lekcji: W notatce powinny się znaleźć następujące ustalenia:

1. Wybór przewodniczącego zespołu.
2. Odczytanie instrukcji do projektu.
3. Omówienie poziomu trudności elementów projektu, propozycje podziału pracy.
4. Wybranie osoby, która będzie opracowywała dokument wynikowy.
5. Ustalenie terminów i autorów wykonania elementów projektu.
6. Ustalenie sposobu kontaktów w zespole.
7. Utworzenie przez przewodniczącego zespołu harmonogramu prac.
8. Ustalenie sposobu wymiany informacji bieżących.
9. Ustalenie standardów wykonawczych (np. utworzenie szablonu dokumentu, formatowanie, czarno-biały lub kolor, rozmiary wykresów, itp).
10. Uzupełnianie harmonogramu prac w ustalonych terminach spotkań zespołu.
11. Wymiana na bieżąco uwag odnośnie postępu prac, elastyczne modyfikowanie ustaleń.
12. Miła atmosfera spotkań, życzliwość, chęć współpracy, umiejętność rozwiązywania problemów.

Ważne jest monitorowanie pracy zespołów na wszystkich etapach realizacji projektu i prowadzenie notatek. Dobrym pomysłem jest poświęcenie jednej godziny na omówienie projektu, wybór grup i podział pracy. Potem co dwa tygodnie sprawdzanie postępów prac i notowanie ich w przygotowanej tabeli. Ustaleniu terminu konsultacji dla uczniów. Następnie przeznaczenie jednej godziny przed terminem oddania projektu na sprawdzenie postępu prac i ich omówienie. Oczywiście nauczyciel może przekładać termin oddania projektu w szczególnych przypadkach, jednak przy projektach długoterminowych należy unikać takich uprzejmości.

Kryteria oceny trzeba ustalić wspólnie z drugim nauczycielem, można skonsultować z uczniami. Każdy nauczyciel może zamieścić w kryteriach oceny swoją część. Najlepiej określić ilość punktów za poszczególne elementy projektu. Wówczas ocena nie nastręcza nauczycielowi żadnych trudności. Wszystkie oceniane elementy projektu muszą zostać wymienione w instrukcji, wyraźnie określone z podaną punktacją. Kryteria oceniania nie mogą ulec zmianie w trakcie realizacji projektu. Dlatego ta część instrukcji musi być przedyskutowana i dobrze przemyślana. Nauczyciele nie mogą omówić projektu dodając w nim elementy, których nie ma w instrukcji. Oczekiwania co do rezultatów projektu, czyli co ma zawierać i jak ma wyglądać publikacja, pokaz, film, dokument, prezentacja, dokumentacja zdjęciowa, materiał, strona internetowa lub inny rodzaj rezultatu muszą być wyraźnie określone. Na końcu każdego projektu powinny się znaleźć podsumowania i wnioski. Można także zachęcać uczniów do podsumowań nabytych doświadczeń w pracy grupowej i samooceny.

Prezentacje projektów przed grupą warto podzielić na części ze wskazaniem maksymalnego czasu na prezentację. Mogą to być: część teoretyczna, część praktyczna prezentowana kolejno przez uczestników zespołu, część podsumowująca. Całkowity czas prezentowania projektu najlepiej zaplanować na 10 minut, maksymalnie 20 minut. Także w czasie prezentacji wyników projektu nauczyciel powinien zadbać o życzliwą atmosferę, aby uczniowie wzajemnie się słuchali i nagradzali się brawami. Ocena końcowa może być jedna bądź kilka. To także musi być wyraźnie wskazane w instrukcji.

Projekty grupowe międzyprzedmiotowe są bardzo dobrze oceniane przez uczniów w badaniach ewaluacyjnych. Często odwiedzają mnie absolwenci, którzy zwracają uwagę, że nabyte podczas pracy nad projektem grupowym doświadczenia bardzo im pomogły w czasie studiów. To bardzo ważne, aby uczniowi zostawić miejsce na twórczość i samodzielność. Aby tak zaplanować realizację materiału, żeby znalazło się miejsce na projekt, Jest to metoda trudna do organizacji dla nauczyciela, ale obserwowanie twórczości uczniów zawsze przynosi nauczycielowi wiele satysfakcji.

Dobre praktyki

1. Projekt indywidualny Badanie funkcji

Projekt jest przeze mnie realizowany od wielu lat, ponieważ cieszy się dobrymi opiniami uczniów. Jest organizowany w klasach trzecich o profilu matematyczno-fizycznym. Od trzech lat polecam uczniom wykonanie projektu z użyciem SageMath. Wcześniej posługiwali się arkuszem kalkulacyjnym i portalem wolframalpha.com. W domu uczniowie wykorzystują portal https://cloud.sagemath.com. W szkole mogli sprawdzone wcześniej (w domu) polecenia przenieść na szkolny serwer SageMath i opublikować.

Instrukcja do projektu zakłada: wybór przez ucznia funkcji o ciekawych własnościach, przeprowadzenie badania wybranej funkcji zgodnie z przykładowym planem oraz udokumentowanie badania.

W ramach niniejszego projektu przeprowadziłam realizację tego uczniowskiego projektu indywidualnego w czterech grupach, z czterech oddziałów (zajęcia z informatyki są prowadzone w grupach nie przekraczających 16 uczniów). Każda z grup przeszła nieco inną ścieżkę celem sprawdzenia, która z metod będzie w ocenie uczniów, a także w moim podsumowaniu najlepsza.

- W pierwszej grupie uczniowie najpierw dostali projekt do zrealizowania. Na SageMath poświęciłam tylko jedną lekcję. Uczniowie nie znali SageMath i uczyli się ze źródeł Internetowych i nauczyciela tylko niezbędnych do zrealizowania projektu poleceń.
- W drugiej grupie uczniowie najpierw uczestniczyli w warsztatach z wykorzystania SageMath w matematyce a potem dostali zadanie wykonania projektu. Wśród ćwiczeń były też takie, które można wykorzystać w projekcie.

- W trzeciej grupie uczniowie najpierw badali funkcję na papierze z kalkulatorem, potem poznali różne narzędzia SageMath, do tych samych obliczeń, które wykonali na kartce.
- Czwarta grupa uczestniczyła najpierw w warsztatach z SageMath, potem warsztaty z Latex, potem przypomnienie obliczeń matematycznych niezbędnych w badaniu funkcji a potem projekt do wykonania z wykorzystaniem SageMath i Latex.

Każda z grup miała inne zadanie co do postaci końcowej projektu. Dla grup 2 i 3 w postaci wydruku pliku w formacie pdf, dla grupy 1 w postaci publikacji na portalu SageMath w języku angielskim a dla grupy 4 wydruk dokumentu utworzonego w Latex. Dokument pdf ze stroną tytułową, spisem treści, nagłówkiem/stopką, numeracją stron, wykorzystujący punktację, sformatowany z przestrzeganiem zasad formatowania dokumentów.

Każda z grup odpowiadała na standardowe pytania. Oprócz tego każda z grup odpowiadała na pytanie wiodące. Grupa pierwsza odpowiadała na pytanie Czy łatwo nauczyć się samodzielnie wybranych elementów SageMath, tych potrzebnych do projektu. Druga grupa odpowiadała na pytanie Czy ćwiczenia z SageMath poprzedzające wykonanie projektu okazały się pomocne. Trzecia grupa oceniała SageMath z punktu widzenia przydatności i oszczędności czasu. Czwarta grupa dodatkowo wypowiadała się na temat pakietu Latex.

2.4.4 Praca z uczniem zdolnym

Głównym celem wprowadzenia elementów języka programowania Python i SageMath w szkole średniej, jest kształtowanie umiejętności pisania przez nauczyciela jak i ucznia oprogramowania służącego do rozwiązywania skomplikowanych problemów matematycznych występujących w trakcie realizacji podstawy programowej przedmiotów ścisłych i przyrodniczych. W efekcie realizacji zajęć lekcyjnych z zastosowaniem tej metody zarówno nauczyciel jak i uczeń będzie mógł stosować poniższe rozwiązania na lekcjach chemii i fizyki tam, gdzie trudność matematyczna może zostać pokonana w stosunkowo łatwy sposób. Program wykorzystuje komendy, które pozwalają uzyskać rozwiązania problemów analizy i algebry takich jak równania różniczkowe, całki, pochodne i wykresy złożonych zależności nieliniowych. Komendy języka i produkty ich działania mogą zostać wypracowane wraz z uczniami, co służy wspólnemu pogłębieniu znajomość języka programowania i jego aplikacji. Przykładami wartymi zastosowania tego typu metody nauczania może być porównanie przemiany adiabatycznej i izotermicznej, otrzymanie rozwiązania w postaci wykresu zależności obu przemian i badanie zmiennej zależnej poprzez zmianę zmiennej zależnej in vivo. Analogicznie zaleca się analizę złożonego problemu prawa stygnięcia Izaaka Newtona warto rozważyć poprzez napisanie programu w języku Python i Sage, wraz ze sporządzeniem formuły, która kreśli uzyskaną zależność oraz podaje równanie opisujące to zjawisko. Nauczyciel chemii i fizyki jest zobowiązany wprowadzać na bieżąco i w miarę potrzeb sygnalizowanych przez uczniów, rodziców oraz pozostałych nauczycieli zespołu nauk przyrodniczych, innowacyjne metody podczas realizacji zajęć obejmujących podstawę programową oraz wykraczających poza ich ramy. Szczególnie, współczesny nauczyciel, powinien być autorem projektów wprowadzających nowoczesne technologie do pracy z uczniem o szczególnych uzdolnieniach. Najlepszym przykładem obrazującym podejście do pracy z uczniem o szczególnych zainteresowaniach interdyscyplinarnych jest problem dotyczący rozwiązywania skomplikowanych problemów matematycznych typu równanie różniczkowe o zmiennych nierozdzielonych występujące w klasycznych olimpijskich zadaniach z termodynamiki. Warto zwrócić uwagę na fakt, że niektóre z tego typu problemów nie objętych zakresem narodowej podstawy programowej, występują jako obowiązkowe w podstawie Programu Dyplomowego Matury Międzynarodowej i są rozwiązywane z użyciem kalkulatorów numerycznych. W tym przypadku nauczyciel może stosować metody programowania łącznie z technikami nauki współpracy w grupie. Reprezentatywnym przykładem użycia tego typu rozwiązań jest zagadnienie teorii słabych kwasów i zasad. Problem ten jest jednym z zagadnień podstawy programowej z przedmiotu chemia, uważanym za trudne. Napisanie z uczniami programu, wraz z instrukcją w języku angielskim pozwala w sposób prosty zrozumieć zasady rządzące dysocjacją elektrolitów słabych.

Nauczyciel, podczas pracy z uczniem o szczególnych uzdolnieniach powinien wdrażać na lekcjach chemii, fizyki i prowadzonych zajęciach pozalekcyjnych metody aktywizujące. Metody te kształtują umiejętności, stwarzają warunki ułatwiające uczenie się, aktywizują, są atrakcyjne dla uczniów i uwzględniają różne style uczenia się. Stosowanie metod aktywizujących w procesie dydaktycznym sprzyja pogłębieniu zdobytej wiedzy, jej operatywności i trwałości. Uczniowie myślą twórczo podczas wykonywania podjętych działań. Angażują się emocjonalnie, są aktywni w sferze percepcyjnej, ruchowej, werbalnej i emocjonalno-motywacyjnej. Zastosowanie języka programowania Python i SageMath pozwala uczniom szczególnie zainteresowanych udziałem w olimpiadach i konkursach przedmiotowych rozwiązywać złożone problemy arytmetyczne takie jak: sporządzanie wykresów zależności stopnia wyższego niż dwa, poszukiwanie miejsc zerowych złożonych funkcji, optymalizacja czy też obliczanie pochodnych lub całek skomplikowanych wyrażeń będących opisem matematycznym danych zjawisk fizykochemicznych.

2.4.5 Samodzielne zdobywanie wiedzy i umiejętności przez ucznia

Nauka matematyki dla większości uczniów nie jest łatwym zadaniem, ponieważ proces ten wymaga od nich znajomości abstrakcyjnych pojęć i ścisłego rozumowania w celu właściwego ich zastosowania. Nauczanie matematyki polega na osiągnięciu dwóch ważnych celów:

1. Opanowaniu przez uczniów wiedzy i zrozumienie koncepcji matematycznych.
2. Wykształceniu matematycznego myślenia w celu rozwiązywania nowych problemów.

Niektórzy eksperci od edukacji matematycznej wyróżniają pięć aspektów umiejętności, które uczniowie powinni rozwinąć w trakcie procesu uczenia się:

- zrozumienie koncepcji - rozumienie pojęć matematycznych, operacji i relacji,
- płynność w stosowaniu procedur - umiejętność elastycznego, precyzyjnego, sprawnego i właściwego zastosowania procedur
- umiejętności strategiczne - formułowanie, reprezentacja i rozwiązywanie problemów matematycznych,
- rozumowanie adaptacyjne - zdolność do logicznego myślenia, refleksji, wyjaśnień i uzasadniania,
- odpowiednie nastawienie - postrzeganie matematyki jako nauki uporządkowanej, użytecznej i wartej zgłębiania, połączone z jej wielorakim zastosowaniem i świadomością własnej skuteczności.

Ponieważ wielu uczniów nie posiada tych umiejętności, SageMath oferuje sposób na ich zdobycie poprzez zmianę metod nauczania i uczenia się matematyki. Istnieje jednak ryzyko, że używając SageMath uczeń rozwiąże zadania nie rozumiejąc związanych z nimi pojęć i algorytmów. Odpowiednie wykorzystanie przez nauczyciela środowiska SageMath podczas lekcji może ułatwić uczniom zrozumienie koncepcji i wyrobienie w nich pozytywnej motywacji do nabywania biegłości w stosowaniu procedur matematycznych.

Cechy SageMath, które są uważane za najbardziej istotne w procesie nauczania matematyki:

- umożliwienie przedstawienia różnych reprezentacji badanych obiektów.
- interpretacja graficzna skomplikowanych pojęć.
- przetwarzane przez komputer długich i żmudnych obliczeń, dzięki czemu uczniowie mogą skoncentrować się na koncepcjach i powiązaniach między nimi.

Systemy algebry komputerowej, takie jak SageMath, umożliwiają nauczanie matematyki z wykorzystaniem komputera. Zdobywanie przez uczniów wiedzy na lekcjach przy pomocy SageMath można ogólnie podzielić na dwa sposoby:

1. Przekazywanie wiedzy przez nauczyciela - komputer jest używany jako narzędzie do prowadzenia prezentacji przez nauczyciela. Na przykład nauczyciel omawia własności funkcji na podstawie jej wykresu lub przedstawia prezentację jakiegoś zagadnienia. Główny akcent położony jest tutaj na przygotowanie odpowiednich materiałów.
2. Samodzielne (lub pod kierunkiem nauczyciela) nabywanie wiedzy przez ucznia - uczniowie wykorzystują komputer jako narzędzie wspomagające ich procesy myślowe. Są oni w stanie znaleźć różne fakty matematyczne i zastosować je w praktyce. Bardzo skuteczne w takich sytuacjach jest wykorzystanie platformy SageMath jako narzędzia wspomagającego pracę uczniów.

Efektywne wykorzystanie SageMath w nauczaniu matematyki wprowadza do tradycyjnej edukacji innowacyjne i jakościowe zmiany, które obejmują weryfikację materiałów edukacyjnych i/lub zmianę sposobu nauczania. Pożądana zmiana w nauczaniu matematyki polega na przejściu od tradycyjnego wykładu przy tablicy do aktywności uczniów wspomaganej przez system algebry komputerowej.

Jedną z głównych zalet notatnika SageMath jest zintegrowane zarządzanie użytkownikami. Po zainstalowaniu serwera SageMath w szkole każdy uczeń może mieć indywidualne konto i dostęp do tego serwera z dowolnego komputera w szkolnej sieci lokalnej. Uczeń oraz nauczyciel mogą publikować swoje notatniki (mogą utworzyć publicznie dostępne ich kopie). Opcjonalnie, opublikowany notatnik może być automatycznie aktualizowany, w przypadku zapisania zmian w oryginalnym notatniku. Opublikowane notatniki mogą być przenoszone na inne serwery. Właściciel notatnika może udostępnić go innym użytkownikom w celu wspólnej pracy nad jego treścią. Daje to możliwość zorganizowania grupowej pracy uczniów nad rozwiązaniem konkretnego problemu.

Funkcje te umożliwiają następującą organizację pracy:

- Nauczyciel publikuje notatnik zawierający konkretne zadanie/problem do rozwiązania.
- Uczeń pracując z kopią tego notatnika rozwiązuje zadanie.
- Uczeń udostępnia swój notatnik nauczycielowi.

- Nauczyciel, w razie potrzeby, pisze komentarze i korekty.
- Nauczyciel publikuje wzorcowe arkusze uczniów.

Podczas pracy z własną kopią notatnika udostępnionego przez nauczyciela uczeń w swoim tempie i czasie nie ograniczonym do tradycyjnej lekcji samodzielnie poznaje zagadnienia, których powinien się nauczyć. Wykorzystując materiały opublikowane przez nauczyciela (np. przedstawienie teorii oraz kod umożliwiający rozwiązanie przykładowych zadań) może zmodyfikować kod tak, aby rozwiązać kolejne zadania zapisane przez nauczyciela w notatniku. W tym celu uczeń musi przeanalizować i zrozumieć kod aby móc go odpowiednio zmodyfikować i rozwiązać zadanie. Uczeń lepiej znający środowisko SageMath i język programowania Python ma możliwość napisania własnego kodu. Nauczyciel może tak sformułować zadania aby uczeń jednocześnie mógł poznać i utrwalić zagadnienia matematyczne oraz wykorzystać środowisko SageMath do ułatwienia i przyspieszenia swojej pracy.

Należy podkreślić, że nowe umiejętności oraz wiedza zdobywane przez ucznia samodzielnie, stopniowo, w jego własnym tempie, w czasie nie ograniczonym jedynie do lekcji w szkole są bardziej trwałe niż w gdyby zostały nabyte przez niego w tradycyjny sposób podczas lekcji.

ROZDZIAŁ 3

Zadania maturalne z Sagem

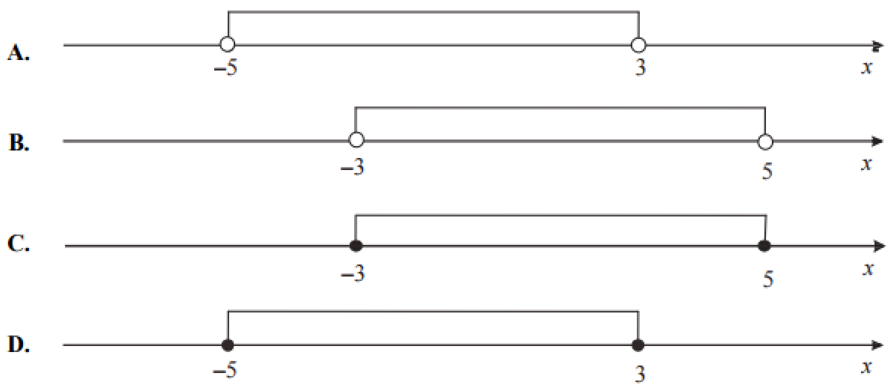
W tym rozdziale prezentujemy Państwu przykład zastosowania SageMath do rozwiązania kilkunastu zadań maturalnych z matematyki. Wiele zadań można by rozwiązać przez wpisanie `solve(..)` co demonstruje użyteczność oprogramowania, ale jednocześnie nie jest pouczające. Dlatego oprócz prezentacji rozwiązania, do zadań został dodany komentarz opisujący wybraną funkcjonalność SageMath a w niektórych przypadkach zadanie zostało wzbogacone o fakultatywne elementy. W efekcie poniższe materiały mogą służyć jako swoista książka przepisów „cookbook” w SageMath.

W trakcie trwania projektu iCSE4school zostało opracowanych wiele zadań z Bawarskiej matury z matematyki i są one w języku niemieckim i angielskim dostepne na stronie

http://visual.icse.us.edu.pl/school/abitur/.

3.1 O graficznym i algebraicznym rozwiązywaniu nierówności

Zadanie 1. (0–1)
Wskaż rysunek, na którym przedstawiono przedział, będący zbiorem wszystkich rozwiązań nierówności $-4 \le x - 1 \le 4$.



Mamy rozwiązań nierówność: $-4 \le x - 1 \le 4$. Jest to dość proste zadanie, ale możemy zobaczyć czy system algebry komputerowej też sobie z nim poradzi. Wpisujemy i od razu mamy wynik: b

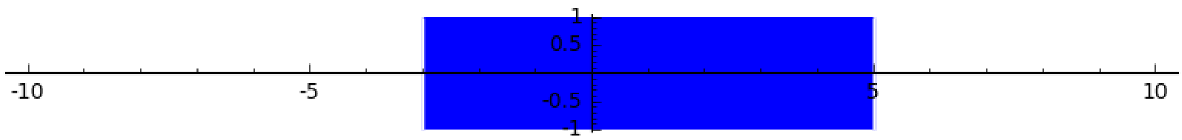
```
solve([-4<=x-1,x-1<=4],x)
```

$$\boxed{[[-3 < x, x < 5], [x == -3], [x == 5]]}$$

... ale czy nie jest on bardziej skomplikowany od zadania?

Możemy przekonać Sage to narysowania obszaru w którym x spełniają warunek z zadania. Zastosujemy, znacznie ogólniejszą, procedurę `region_plot`:

```
var('x,y')
region_plot([-4<=x-1,x-1<=4],(x,-10,10),(y,-1,1) )
```



== czy =?

We wszystkich językach programowania mamy dwa różne znaki równości. W Pythonie (który jest językiem Sage), mamy

- = znak przypisania do zmiennej.
- == znak porównania logicznego - czy to jest „matematyczne równa się”

Znaki <, >= itp. mają znaczenie takie jak w matematyce.

3.2 O logarytmach i upraszczaniu

Zadanie 2. (0–1)

Dane są liczby $a=-\frac{1}{27}$, $b=\log_{\frac{1}{4}} 64$, $c=\log_{\frac{1}{3}} 27$. Iloczyn abc jest równy

A. -9 **B.** $-\frac{1}{3}$ **C.** $\frac{1}{3}$ **D.** 3

Dane są liczby: $a = \frac{1}{27}$, $b = \log_{\frac{1}{4}}(64)$, $c = \log_{\frac{1}{3}}(27)$. Mamy policzyć iloczyn.

Wystarczy poprosić system algebry komputerowej i od razu mamy wynik:

```
a = 1/27
b = log(64,base=1/4)
c = log(27,base=1/3)
show(a*b*c)
```

Otrzymujemy wynik $\frac{1}{3}$.

Jak to działa?

Korzystamy tu z funkcji *log* z Sage - jest to algebraiczna implementacja logarytmu, która zna pewne zasady upraszczania. Na przykład logarytm z 64 o podstawie $\frac{1}{4}$ zostanie przedstawiony jako:

```
show(  log(64,base=1/4) )
```

3.3 O tym jak `solve` wykona pracę za nas i kiedy nas zawiedzie

Zadanie 4. (0–1)

Równość $\frac{m}{5-\sqrt{5}} = \frac{5+\sqrt{5}}{5}$ zachodzi dla

A. $m = 5$ **B.** $m = 4$ **C.** $m = 1$ **D.** $m = -5$

Czy równość:

$$-\frac{m}{\sqrt{5}-5} = \frac{\sqrt{5}+1}{5}$$

zachodzi dla: $m = 5$, $m = 4$, $m = 1$, czy może $m = -5$?

System algebry komputerowej z chęcią rozwiąże również i taką równość:

```
var('m')
solve( m/(5-sqrt(5))==(5+sqrt(5))/5,m)
```

Jak to działa: `solve`?

Funkcja *solve* jest bardzo potężnym narzędziem! Umożliwia ono nam algebraiczne rozwiązywanie równiań i układów równań. W powyższym przypadku zadziałało ono „magicznie” i otrzymaliśmy wynik. Nie zawsze jednak wynik jest możliwy do otrzymania, a czasem jest bardzo skomplikowany. Spróbuj sam, rozwiąż na przykład: $ax^2 + bx + c == 0$.

```
var('a,b,c,x')
show( solve(a*x^2+b*x+c==0,x) )
```

Zastąp wyraz z najwyższą potęgą x przez ax^3, ax^4,... Czy komputer znajdzie odpowiedź dla dowolnie dużej potęgi? Ograniczenia narzucone są przez Teorię Galois .

Zadanie daje nam jednak zbiór czterech wartości m do sprawdzenia, więc możemy też pokusić się o sprawdzenie każdej z opowiedzi:

```
for m in [5, 4, 1, -5]:
    print  bool(m/(5-sqrt(5))==(5+sqrt(5))/5)
```

Jak to działa: `bool`?

Funkcja *bool* próbuje, używając systemu algebry komputerowej sprawdzić algebraiczną poprawność równania. Pętla *for* zapewnia sprawdzenie dla każdej odpowiedzi z testu.

Możemy się posłużyć też przybliżeniem arytmetycznym wyrażeń po lewej i prawej stronie równości. Nie jest to dokładny wynik w sensie matematycznym, ale poniższy kod nie pozostawia złudzeń która odpowiedź jest poprawna:

```
for m in [5, 4, 1, -5]:
    print  (m/(5-sqrt(5))).n(),"=",((5+sqrt(5))/5).n()
```

3.4 Jak *implicit_plot* pomoże graficznie zinterpretować rozwiązania układu równań

Zadanie 5. (0–1)

Układ równań $\begin{cases} x - y = 3 \\ 2x + 0,5y = 4 \end{cases}$ opisuje w układzie współrzędnych na płaszczyźnie

A. zbiór pusty.
B. dokładnie jeden punkt.
C. dokładnie dwa różne punkty.
D. zbiór nieskończony.

Pytanie jest o liczbę rozwiązań układu równań. Ponownie możemy uzyskać odpowiedź od systemu rozkazując komputerowi rozwiązać nasz układ:

```
var('x,y,m')
solve( [x-y==3,2*x+0.5*y==4],[x,y])
```

[[x == (11/5), y == (-4/5)]]

Jednak ponieważ nie potrzebne są nam wartości pierwiastków tego układu równań, to możemy posłużyć się metodą graficzną! Rozwiązania układu równań to miejsca przecięć się wykresów obu krzywych. Narysujmy wykresy i zobaczmy czy udzielimy na podstawie ich oględzin poprawne odpowiedzi:

```
var('x,y')
p = implicit_plot(x-y==3,(x,-3,3),(y,-3,3))
p += implicit_plot(2*x+0.5*y==4,(x,-3,3),(y,-3,3),color='red')
p.show()
```

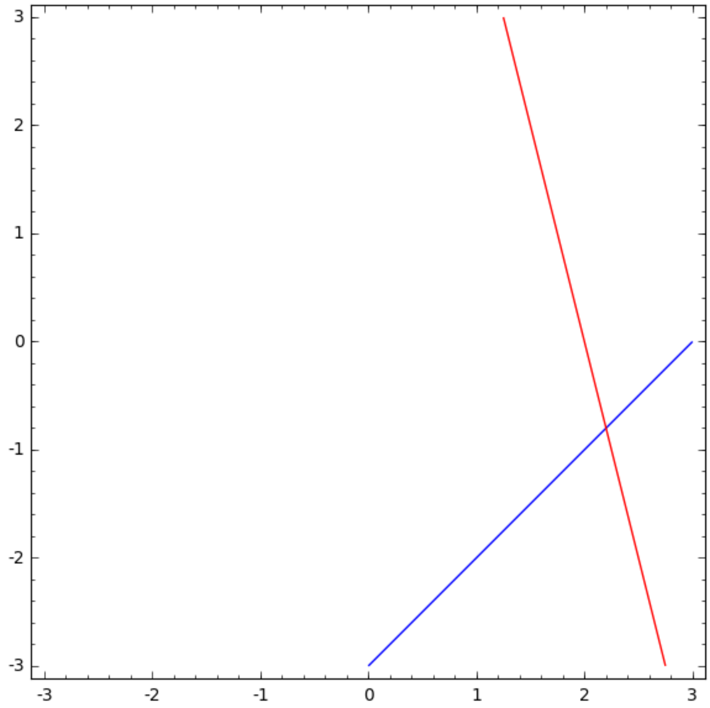
Dlaczego `implicit_plot`?

`implicit_plot` jest narzędziem do rysowania tak zwanych funkcji uwikłanych, czyli w postaci $F(x, y) == 0$. „Pod maską” jest to o wiele bardziej złożona machineria od zwykłego *plot*, ale dla nas ma on pewne zalety. Po pierwsze nie musimy rozwiązywać równania na y. Po drugie mamy pewność, że wykres nie wyjdzie z zadanego obszaru. Implicit plot potrafi narysować też krzywe które nie są wykresami funkcji. Sprawdź sam, na przykad rysując okrąg $x^2 + y^2 == 1$ lub prostą $x == 0$.

`plot( f(x),(x,x1,x2))` rysujące wykres $f(x)$ w przedziale $x \in (x_1, x_2)$. Zakresy osi y ustalane są automatycznie.

Czy możemy użyć zwykłego `plot`?

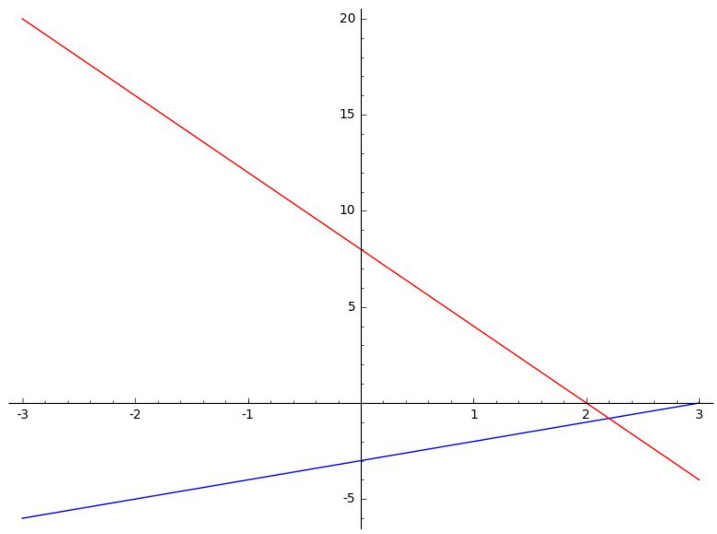
Oczywiście! Rozwiążmy oba równania ze względu na y:




```
var('x,y')
show( solve(x-y==3,y) )
show( solve(2*x+0.5*y==4 ,y) )
```

Teraz możemy narysować funkcję standardową procedurą rysującą wykresy funkcji jednej zmiennej:

```
p = plot(x-3,(x,-3,3))
p += plot(-4*x+8,(x,-3,3),color='red')
p.show()
```



Ale zaraz! Wyszły inne wykresy! A może się nam to wydaje? Zmieńmy zakres wartości funkcji na taki jak po pierwszym przypadku i bedzie to samo. Wystarczy zmienić przedostatnią linijke na:

```
p += plot(-4*x+8,(x,-3,3),color='red',ymax=3,ymin=-3,aspect_ratio=1)
```

3.5 O tym jak nakłonić funkcję `plot` do współpracy

Zadanie 7. (0–1)

Równanie $\frac{x-1}{x+1} = x - 1$

A. ma dokładnie jedno rozwiązanie: $x = 1$.
B. ma dokładnie jedno rozwiązanie: $x = 0$.
C. ma dokładnie jedno rozwiązanie: $x = -1$.
D. ma dokładnie dwa rozwiązania: $x = 0$, $x = 1$.

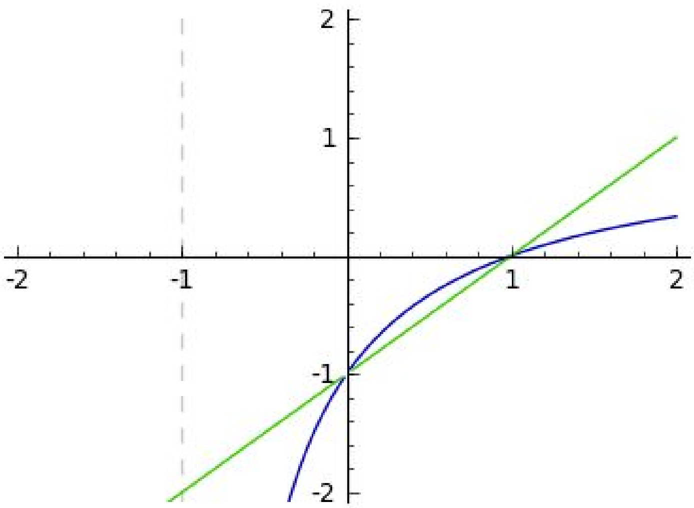
Równanie $\frac{x-1}{x+1} == x - 1$ aż się prosi by je rozwiązać za pomocą algebry komputerowej. Rzeczywiście, wynik otrzymujemy natychmiast:

```
var('x')
solve( (x-1)/(x+1)==x-1,x)
```

$$\boxed{[x == 0, x == 1]}$$

Ale przyjrzyjmy się w ilu miejscach przecinają się wykresy prawej i lewej strony powyższej równości:

```
plot([(x-1)/(x+1),x-1],(x,-2,2),detect_poles='show',\
 ymin=-2,ymax=2,figsize=4)
```



Po co tyle dodatkowych parametrów?

Spróbuj do poprzedniej komórki wpisać „surową” komendę:

```
plot([(x-1)/(x+1),x-1],(x,-2,2))
```

Co się stało z wykresem? Okazuje się, że mamy osobliwość, która nieco szkodzi automatycznemu skalowaniu osi. Dlatego najlepiej będzie ręcznie ustawić zakres wartości rzędnych. Dodatkowo warto zasugerować Sage'owi żeby wykrył osobliwości i je nam pokazał rysując pionowe asymptoty.

3.6 O tym jak technologia `@interact` pomoże rozwiązać równanie z parametrem

Zadanie 9. (0–1)

Na wykresie funkcji liniowej określonej wzorem $f(x)=(m-1)x+3$ leży punkt $S=(5,-2)$. Zatem

A. $m=-1$ **B.** $m=0$ **C.** $m=1$ **D.** $m=2$

Korzystając z algebry komputerowej wynik otrzymujemy natychmiast:

```
var('x,y,m')
rownanie = y == (m-1)*x+3
show(rownanie)
show(rownanie.subs({x:5,y:-2}).solve(m))
```

Otrzymujemy $m = 1$

Ponieważ mamy do sprawdzenia tylko cztery możliwości można też pokusić się o narysowanie graficznej reprezentacji w każdym z przypadków:

```
var('x,y,m')
rownanie = y == (m-1)*x+3
@interact
def rysuj(m_=[1,2,-1,0]):
    p = point((5,-2),size=20,color='red')
    p+=implicit_plot(rownanie.subs({m:m_}),(x,-10,10),(y,-10,10))
    p.show(figsize=4,aspect_ratio=1)
```

Jak to działa? - *@interact*

W systemie Sage polecenie to jest niezwykle wygodnym narzędziem do szybkiego tworzenia interaktywnych wykresów lub obliczeń. `@interact` jest tak zwanym dekoratorem funkcji `rysuj`. Dekorator uruchamia funkcję `rysuj` jako argument innej funkcji, która generuje dla nas wygodne przyciski i po ich naciśnięciu wywołuje funkcje `rysuj` z wybraną wartością parametru.

3.7 O konstrukcji złożonych poleceń z pomocą metod

Zadanie 10. (0–1)

Funkcja liniowa f określona wzorem $f(x) = 2x + b$ ma takie samo miejsce zerowe, jakie ma funkcja liniowa $g(x) = -3x + 4$. Stąd wynika, że

A. $b = 4$ **B.** $b = -\frac{3}{2}$ **C.** $b = -\frac{8}{3}$ **D.** $b = \frac{4}{3}$

Korzystając z algebry komputerowej możemy praktycznie w jednej linijce otrzymać wynik:

```
var('x,b')
f =  2*x + b
g = -3*x + 4
f.subs(solve(g,x)[0]).solve(b)[0].show()
```

Otrzymujemy $b = -\frac{8}{3}$.

Co oznacza wyrażenie `f.subs(solve(g,x)[0]).solve(b)[0].show()`? W bezpośrednim przekładzie z Pythona na Polski:

> Podstaw do wyrażenia `f` pierwsze rozwiązanie równania $g(x) = 0$, rozwiąż ze względu na b i pokaż ładnie na ekranie pierwsze rozwiązanie.

Informacja: Warto zauważyć, że dla Sage `solve(g,x)` to jest to samo co `solve(g==0, x)` oznacza: rozwiąż $g(x) = 0$. Wynikiem rozwiązywania jest zawsze lista, może być pusta jeśli nie ma rozwiązań lub jednoelementowa.

Jak to działa? - metody i funkcje.

Językiem systemu algebry komputerowej Sage jest Python. W Python-ie wszystkie zmienne są objektami i posiadają, prócz danych takich jak np. wartość zmiennej, tak zwane metody. Metodami są funkcje, które wywołujemy taką składnią: `nazwa_objektu.funkcja()`.

Wypróbuj sam działanie metod. Weź na przykład wyrażenie algebraiczne Sage, które jest bardzo złożonym objektem i posiada np. rozmaite metody do manipulacji na wyrażeniach. Na przykład w komórce:

```
var('x,a')
wyrazenie = (x+a)^2
wyrazenie
```

dopisz do statniej liniki:

1. `wyrazenie.show()` - pokaże nam ładną postać matematyczną wzoru.

2. `wyrazenie.expand()` - rozwinie wzór, spróboj połączyć z poprzednim: `wyrażenie.expand().show()`
3. `wyrazenie.subs(a==2)`

Informacja: W Sage jest system pomocy, spróbuj napisać `wyrazenie.expand?`. Lista dostępnych metod jest olbrzymia: `dir(wyrazenie)`.

W notatniku (sagenb lub jupyter) działa klawisz uzupełnienia `TAB` (tabulator) po `wyrazenie.`

3.8 Pojedynek funkcji kwadratowej z `solve`

Zadanie 11. (0–1)

Funkcja kwadratowa określona jest wzorem $f(x) = x^2 + x + c$. Jeżeli $f(3) = 4$, to

A. $f(1) = -6$ **B.** $f(1) = 0$ **C.** $f(1) = 6$ **D.** $f(1) = 18$

Korzystając z algebry komputerowej mamy wynik od razu:

```
var('x,c')
f(x) = x^2 + x + c
f(1).subs(solve(f(3)==4,c)[0])
```

Otrzymujemy -6.

Pozostaje jeszcze przetłumaczyć `f(1).subs(solve(f(3)==4,c)[0])` z Pythona na Polski:

„Podstaw do wyrażenia $f(1)$ pierwsze rozwiązanie równania $f(3) = 4$, ze względu na c."

Podpowiedź: W Pythonie numerujemy listy od zera. Oznacza to, że pierwszy element listy `l` to `l[0]`.

3.9 O brutalnej metodzie i pętli `for` przyrządzonej na dwa sposoby

Zadanie 12. (0–1)

Ile liczb całkowitych x spełnia nierówność $\frac{2}{7} < \frac{x}{14} < \frac{4}{3}$?

A. 14 **B.** 15 **C.** 16 **D.** 17

Ile liczb całkowitych spełnia nierówność $\frac{2}{7} < \frac{x}{14} < \frac{4}{3}$?

Zadanie może się wydawać zbyt łatwym by traktować je komputerem, ale możemy je wykorzystać by poćwiczyć sobie pętlę `for`.

I tak możemy sprawdzić („dla pewności”) wszystkie liczby z przedziału $x \in (-1000, 1000)$ wykorzystując system Sage, a właściwie sam język Python. Takie podejście czasem jest zwane metodą *brute force* - czyli brutalną zob. link . Zmuszamy bowiem komputer do brutalnie dużego wysiłku - przynajmniej w stosunku to złożoności postawionego problemu.

Uczyńmy to więc:

```
liczby = [x for x in range(-1000,1000) if 2/7<x/14<4/3]
print len(liczby),":",liczby
```

14 : [5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18]

Oczywiście każdy matematyk zaprotestuje, nie mamy pewności czy nie ma liczb całkowytych poza przedziałem, które spełniają te nierówności. W tym przypadku mie ma problemu by rozwiązać w dziedzinie liczb rzeczywistych:

```
var('x')
solve([2/7<x/14,x/14<4/3],x)
```

[[4 < x, x < (56/3)]]

Ponieważ wykonaliśmy sprawdzenie każdej liczby z osobna, można również oszacować zakres. Skoro $\frac{x}{14}$ jest większe od $\frac{2}{7}$ to na pewno będzie większe od 0. Z drugiej strony jest mniejsze od $\frac{4}{3}$ to będzie mniejsze też od np. 2. Czyli wychodzi ze x będzie większe od 0 i mniejsze od $2 \times 14 = 28$. Okazało się, że poprzedni przedział nie zawęził poszukiwania!

Ja to działa? - „list comprehension” - produktowanie list

Wyrażenie `[x for x in range(-1000,1000) if 2/7<x/14<4/3]` zawiera konstrukcję która wykona następującę polecenie: *podaj mi wszystkie x od -1000 do 999, które* *spełniają dany warunek*. Jest to de facto pętla for, tylko tak sprytnie zapisana, że od razu generuje listę. Przypomina to nieco zapis matematyczny:

$$S = \{x \mid x \in \mathbb{N},\ \frac{2}{7} < \frac{x}{14} < \frac{4}{3}\}$$

Możemy również wykonać to samo zadanie bardziej klasycznie wyglądającą pętlą `for`:

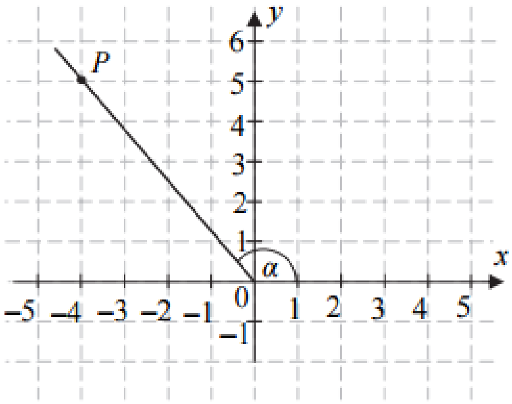
```
i = 0
for x in range(-1000,1000):
    if 2/7<x/14<4/3:
        i+=1
print i
```

Otrzymujemy 14.

3.10 Jak skomplikować proste i popaść w kłopoty z funkcją tangens

Zadanie 14. (0–1)

Tangens kąta α zaznaczonego na rysunku jest równy



$P = (-4, 5)$

A. $-\frac{\sqrt{3}}{3}$

B. $-\frac{4}{5}$

C. -1

D. $-\frac{5}{4}$

Wynik otrzymujemy natychmiast bez komputera metodą *przez „ogląd”:*, łatwo widać, że ostatnia odpowiedz jest prawdziwa. Mówiąc inaczej korzystamy z definicji tangensa: $tan(\alpha) = \frac{y}{x}$ dla $x = -4$ i $y = 5$.

Wyobraźmy sobie jednak, że zapomnielismy co to tangens, ale pamiętamy, że Sage posiada funkcję odwrotną `arctan`, która pozwoli nam rozwiązać zadanie „od końca” przez wyliczenie kątów i narysowanie wszystkie odpowiedzi w teście. Będziemy do tego potrzebowali jeszcze odległości od początku układu współrzędnych, którą wyliczymy z Twierdzenie Pitagorasa: $r = \sqrt{x^2 + y^2} = \sqrt{5^2 + 4^2}$. Z definicji *sinusa* i *cosinusa* wiemy też, że współrzędne spełniają:

$$x = r\cos(\phi)$$
$$y = r\sin(\phi)$$

Wystarczy więc dla każdej odpowiedzi z testu obliczyć kąt i narysować odpowiadający punkt na wykresie (np. funkcją `point((x,y))`. Właściwą odpowiedź rozpoznamy graficznie.

Nasuwa się pytanie: czy można jeszcze bardziej skomplikować to zadanie?

Napiszmy więc:

```
r = sqrt( 5^2+4^2 )
tanphi=-sqrt(3)/3 # inne odpowiedzi: -4/5,-1,-5/4
phi = arctan(tanphi )
point ( (r*cos(phi),r*sin(phi) ),size=20)
```

Jednak wykres nie jest zbyt piękny, fajnie by było ustawić takie same zakresy osi jak w zadaniu. Aby precyzyjnie odczytać położenie punktu z wykresu dorysujemy też siatkę z lini pionowych i poziomych `gridlines`.

```
r = sqrt( 5^2+4^2 )

tanphi=-sqrt(3)/3
phi = arctan(tanphi )
point ( (r*cos(phi),r*sin(phi) ),size=20,\
  ymin=-6,ymax=6,xmin=-6,xmax=6,\
  gridlines=[range(-5,6),range(-5,6)])
```

No to mamy już nasz program gotowy, dodajemy jeszcze `@interact` by sprawdzać odpowiedzi kliknięciem w odpowiedni klawisz:

```
r = sqrt( 5^2+4^2 )
@interact
def _(tanphi=[-sqrt(3)/3,-4/5,-1,-5/4]):
    phi = arctan(tanphi )
    plt = point ( (r*cos(phi),r*sin(phi) ),size=20,\
      ymin=-6,ymax=6,xmin=-6,xmax=6,\
      gridlines=[range(-5,6),range(-5,6)])
    plt.show()
```

Co się okazuje, **żadna** odpowiedź daje punktu w tym samym miejscu jak w zadaniu? Co poszło nie tak?

Okazuje się że zawiniła włanosć funkcji $\tan$. Jej okres to π - a nie 2π jak u poczciwego *sinusa*. Oznacza to, że jeżeli:

$$a = \tan(\phi),$$

to także zachodzi:

$$a = \tan(\phi + \pi)$$

Czyli działając funkcją odwrotną:

$$\arctan(a) = \phi + \pi!$$

Dodajmy więc do naszego rysunku drugi punkt dlą kąta $\phi + \pi$:

```
r = sqrt( 5^2+4^2 )
@interact
def _(tanphi=[-sqrt(3)/3,-4/5,-1,-5/4]):
    phi = arctan(tanphi )
    plt = point ( (r*cos(phi),r*sin(phi) ),size=20,\
      ymin=-6,ymax=6,xmin=-6,xmax=6,\
      gridlines=[range(-5,6),range(-5,6)])
    phi += pi
    plt += point ( (r*cos(phi),r*sin(phi) ),size=20 )
    plt.show()
```

I zgadza się! W ostatnim przypadku otrzymujemy wszystkie punkty dla których $\tan(\phi) = -\frac{5}{4}$ i $r = \sqrt{41}$. Jednym z tych punktów jest ten z rysunku w zadaniu.

Jak to działa: `gridlines`

Opcja *gridlines* pozwala na narysowanie pionowych i poziomych lini na dowolnym wykresie w Sage. Składnia wygląda tak:

```
gridlines = [ [x1,x2,...,xn], [y1,y2,...,yn] ]
```

gdzie *x1,x2,...* to położenia prostych pionowych a *y1,y2,...* poziomych. Jest to zagnieżdżona lista, dwuelementowa, na której znajdują się dwie listy ze współrzędnymi. Przykłady:

- `gridlines=[[1,2,5],[]]` - trzy linie pionowe
- `gridlines=[ range(1,20),[3]]` - 19 lini pionowych i jedna pozioma

3.11 O tożsamości trygonometrycznej i założeniach

Zadanie 15. (0–1)

Jeżeli $0° < \alpha < 90°$ oraz $\mathrm{tg}\alpha = 2\sin\alpha$, to

A. $\cos\alpha = \frac{1}{2}$ **B.** $\cos\alpha = \frac{\sqrt{2}}{2}$ **C.** $\cos\alpha = \frac{\sqrt{3}}{2}$ **D.** $\cos\alpha = 1$

Sprawdźmy czy Sage potrafi rozwiązać równanie $\tan(x) = 2\sin(x)$?

```
var('x')
print solve(tan(x)==2*sin(x),x)
```

Otrzymujemy dwa rozwiązania. Które jest poprawne? Zauważmy, że mamy podany w zadaniu warunek dla kąta, co wyklucza jedno z nich. W Sage możemy spróbowac podać ten warunek używając funkcji `assume()`:

```
var('x')
assume(x>0)
assume(x<pi)
show( cos(solve(tan(x)==2*sin(x),x)[0].rhs()) )
```

Warto też przyjrzeć się wykresom lewej i prowej strony równania. Nie pomoże to w tym przypadku otrzymać rozwiązanie, ale zasugeruje, że rozwiązanie istnieje i pokaże mniej więcej jego wartość:

```
p = plot([tan(x),2*sin(x)],(x,0,pi),\
    ymax=5,ymin=-5,detect_poles='show')
p.show()
```

Jak to działa? - `rhs()`

W Sage funkcja `solve` standardowo zwraca listę wyrażeń w postaci:

```
[ x==2, x==3 ]
```

Czasem chcemy podstawić - bez ręcznego przepisywania wyrażenia - wartość rozwiązania po lewej stronie równości np. do innego wyrażenie. W Sage służy do tego celu metoda `.rhs()` (z ang. *right hand side*), która zwraca prawą strone wyrażenia zawierającego logiczne (matematycznie) równa się ==. Na przykład `(a==1).rhs()`, zwróci 1.

Istnieje też metoda `.lhs()` (z ang. *left hand side*), która zwraca lewą stronę wyrażenia.

3.12 O kątach i rysowaniu

Zadanie 16. (0–1)

Miara kąta wpisanego w okrąg jest o 20° mniejsza od miary kąta środkowego opartego na tym samym łuku. Wynika stąd, że miara kąta wpisanego jest równa

A. 5° **B.** 10° **C.** 20° **D.** 30°

Wiemy, że kąt wpisany jest dokładnie dwa razy mniejszy od środkowego więc zadanie sprowadza się do rozwiązania równania: $2x = x + 20$, gdzie x oznacza wartość kąta wpisanego:

```
solve( 2*x == x + 20,x)
```

I mamy nasze rozwiązanie!

Ale możemy pokusić się o geometryczną konstrukcje, wykorzystująca w Sage możliwość rysowania lini przechodzącej przez zadane punkty `line` i okręgów `circle`. Załóżmy, że okrąg ma promien jeden to do narysowania kąta środkowego potrzebujemy wzorów na współrzędne punktu na okręgu. Z pomocą przychodzą definicje sinusa i cosinusa i mamy:

$$\cos(\phi) = \frac{x}{r}$$
$$\sin(\phi) = \frac{y}{r}$$

Czyli współrzędne punktu na okręgu jednostkowym będącego pod kątem ϕ względem osi x wynoszą: $(\cos(\phi), \sin(\phi))$

```
var('x')
def dwa_katy(phi0= 1.23):

    p = line( [(0,0),(cos(phi0),sin(phi0))])
    p += line( [(-1,0),(cos(phi0),sin(phi0))])
    p += circle((0,0),1,color='gray',fill=True,alpha=0.2)
    return p

@interact
def _(phi0=slider(0,180,1)):
    dwa_katy(phi0/180*pi).show(figsize=5)
```

Hmmm, a co z drugą połową wykresu? Oczywiście mamy symetrię $y \to -y$ - lub $\phi \to -\phi$ więc wystarczy odbic względem osi x nasz rysunek. Moglibysmy dopisać odpowiednie komendy w funkcji `dwa_katy`, ale możemy postąpić chytrzej!. Zauważmy, że w funkcji `dwa_katy` możemy podać ujemną wartość kąta. Gdybyśmy mogli nałożyć na rysunek dla kąta ϕ jego symetryczny odpowiednik dla $-\phi$ do otrzymalibyśmy kompletną ilustracje. W Sage jest to niezwykle proste - objekty graficzne można do siebie dodawać, a nasza funkcja własnie zwraca objekt graficzny. Poeksperymentujmy sami zastępując ostatnią linię kodu przez:

Rys. 3.1: Wynik jest interaktywny, najlepiej samemu wykonać kod:

```
(dwa_katy(phi0/180*pi)+dwa_katy(-phi0/180*pi)).show(figsize=5)
```

3.13 Kąty, radiany i szacowanie funkcji sinus

Zadanie 17. (0–1)

Pole rombu o obwodzie 8 jest równe 1. Kąt ostry tego rombu ma miarę α. Wtedy

A. $14° < \alpha < 15°$ **B.** $29° < \alpha < 30°$ **C.** $60° < \alpha < 61°$ **D.** $75° < \alpha < 76°$

Jeżeli obwód rombu wynosi 8 to jego bok jest cztery razy mniejszy czyli ma ma długość $a = 2$. Z drugiej strony pole rombu to $P = a^2 \sin(\alpha)$. Możemy więc wyznaczyć kąt α z równania:

$$a^2 \sin(\alpha) = 1$$

czyli:

$$4 \sin(\alpha) = 1$$

Mając do dyspozycji Sage rozwiązanie otrzymujemy natychmiast:

```
solve(4*sin(x)==1,x)[0].rhs().n()*180/pi.n()
```

Otrzymujemy 14.4775121859299.

Gotowe!

Radiany vs. stopnie

Argumentem funkcji trygonometrycznych są kąty w radianach natomiast w zadaniu (i życiu codziennym) posługujemy się stopniami. Przeliczanie radianów w stopnie i na odwrót jest bardzo proste, wystarczy pamiętać, że kąt pełny $360°$ to 2π rad. Wynika z tego, że jeżeli mamy wynik w radianach, to aby przeliczyć go na stopnie należy pomnożyć przez $\frac{180}{\pi}$. Z drugiej strony mając stopnie możemy otrzymać radiany mnożąc przez $\frac{\pi}{180}$

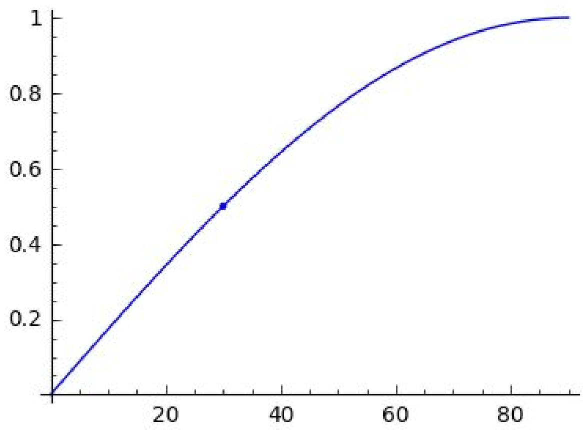
Przypuśćmy, że nie potrafimy oszacować numeryczne wartości $\arcsin(\frac{1}{4})$. Czy możemy rozwiązać to zadanie?

Wiemy przecież, że $\arcsin(\frac{1}{2}) = \frac{pi}{6}\text{rad} = 30°$ Narysujmy wykres funkcji $\sin$ dla kątów ostrych.

```
p = plot( sin(x/180*pi), (x,0,90), figsize=4)
p
```

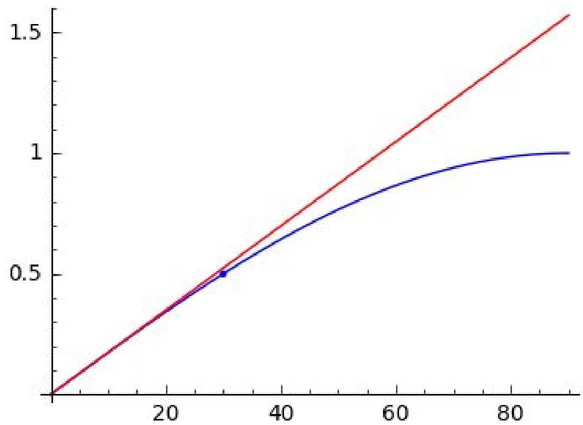
Punkt $(30°, \frac{1}{2})$ należy do wykresu funkcji sinus:

```
p = plot( sin(x/180*pi), (x,0,90), figsize=4)
p += point( [30,1/2])
p
```



Zauważmy, że dla kątów mniejszych od $30°$ funkcja sinus może być całkiem nieźle przybliżona przez liniową:

```
p = plot( sin(x/180*pi), (x,0,90), figsize=4)
p += point( [30,1/2])
p += plot( x/180*pi, (x,0,90),color='red')
p
```



Ponadto widać, że wykres sinusa leży pod prostą:

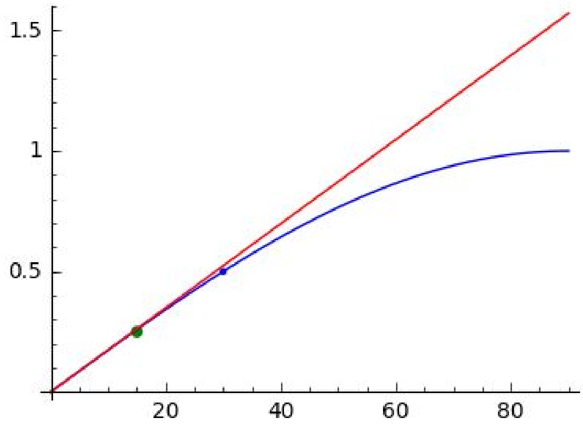
$$\sin(x) \lesssim x,$$

lub

$$\sin(\frac{\pi}{180}x) \lesssim \frac{\pi}{180}x.$$

Możemy to zinterpretować następująco - dla małych kątów sinus rośnie proporcjonalnie z kątem. Jeżeli dla $30°$ wynosił $1/2$ to będzie $\sin 15° \simeq \frac{1}{4}$. Zobaczmy graficznie jak przedstawia się nasze przybliżenie:

```
p = plot( sin(x/180*pi), (x,0,90), figsize=4)
p += plot( x/180*pi, (x,0,90),color='red')
p += point( [30,1/2])
p += point( [15,1/4],color='green',size=30)
p
```



3.14 O prostopadłości i równoległości prostych

Zadanie 18. (0–1)

Prosta l o równaniu $y=m^2x+3$ jest równoległa do prostej k o równaniu $y=(4m-4)x-3$. Zatem

A. $m=2$ **B.** $m=-2$ **C.** $m=-2-2\sqrt{2}$ **D.** $m=2+2\sqrt{2}$

Zadanie 19. (0–1)

Proste o równaniach: $y=2mx-m^2-1$ oraz $y=4m^2x+m^2+1$ są prostopadłe dla

A. $m=-\frac{1}{2}$ **B.** $m=\frac{1}{2}$ **C.** $m=1$ **D.** $m=2$

Sa to dwa bardzo podobne zadania dotyczą one wzajemnego położenia prostych na płaszczyźnie. Postąpimy w następujący sposób. Narysujemy sobie każdą z odpowiedzi i spróbujemy wizualnie sprawdzić czy jest poprawnym rozwiązaniem. Z drugiej strony, wypiszemy jawnie postać prostych i spróbujemy „odgadnąć” regułę.

---

aspect_ratio

Będziemy używać funkcji `implicit_plot` z opcją `acpect_ratio=1`. W innym przypadku kąty na rysunkach nie odpowiadały by faktycznym!

Zadanie 18:

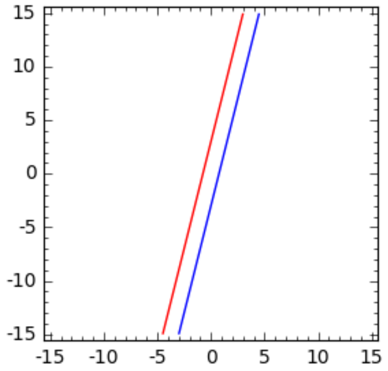
```
var('x,y,m')
rownanie1 = y==m^2*x+3
rownanie2 = y==(4*m-4)*x-3
@interact
def _(m_=[-2,2,2+2*sqrt(2),-2-2*sqrt(2)]):
    p = implicit_plot(rownanie1.subs({m:m_}),
     ↪  (x,-15,15),(y,-15,15),color='red')
    p+= implicit_plot(rownanie2.subs({m:m_}),(x,-15,15),(y,-15,15))
    show([rownanie1.subs({m:m_}),rownanie2.subs({m:m_})])
    p.show(figsize=4,aspect_ratio=1)
```

Zadanie 19:

```
var('x,y,m')
rownanie1 = y==2*m*x-m^2-1
rownanie2 = y==4*m^2*x+m^2+1
@interact
def _(m_=[2,1,1/2,-1/2]):
    p = implicit_plot(rownanie1.subs({m:m_}),
     ↪  (x,-15,15),(y,-15,15),color='red')
    p+=implicit_plot(rownanie2.subs({m:m_}),(x,-15,15),(y,-15,15))
```

m_ -2 2 2*sqrt(2) + 2 -2*sqrt(2) - 2

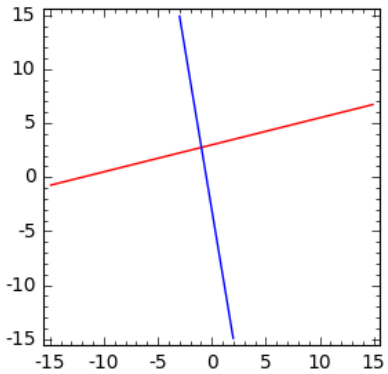
$$[y = 4\,x + 3, y = 4\,x - 3]$$



```
    show([rownanie1.subs({m:m_}),rownanie2.subs({m:m_})])
    p.show(figsize=4,aspect_ratio=1)
```

m_ 2 1 1/2 -1/2

$$\left[y = \frac{1}{4}\,x + 3, y = -6\,x - 3\right]$$



Czy wykonując powyższy kod możemy sformułować stwierdzenie na temat prostopodłości i równoległości prostych?

3.15 Punkty odbicia i działania na wektorach

Zadanie 20. (0–1)

Dane są punkty $M=(-2,1)$ i $N=(-1,3)$. Punkt K jest środkiem odcinka MN. Obrazem punktu K w symetrii względem początku układu współrzędnych jest punkt

A. $K'=\left(2,-\frac{3}{2}\right)$ **B.** $K'=\left(2,\frac{3}{2}\right)$ **C.** $K'=\left(\frac{3}{2},2\right)$ **D.** $K'=\left(\frac{3}{2},-2\right)$

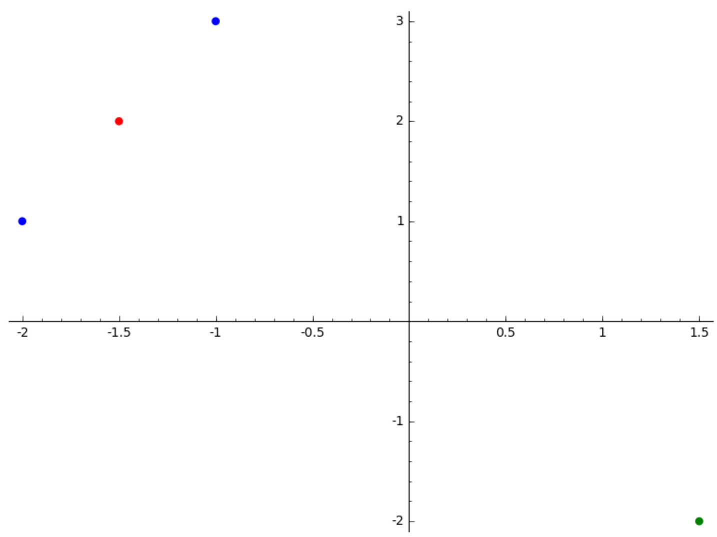
Zadanie to można łatwo sformułować w języku operacji na wektorach swobodnych. W systemie Sage istnieje objekt `vector`, który z listy współrzędnych tworzy wektor w przestrzeni Eulidesowej (dowolnego wymiaru!). W naszym przypadku mamy przestrzeń dwuwymiarową i parę współrzędnych dla każdegu punktu.

Wektory możemy dodawać i mnożyć przez liczbę. Z drugiej strony, możemy każdy wektor narysować jako punkt używając polecenia `point`.

- Środek odcinka na płaszczyźnie to średnia arytmetyczną wektorów reprezentujących jego końce.
- Odbicie względem początku układu współrzędnych to pomnożenie przez -1.

Poniższy kod rozwiązuje i ilustruje nasze zadanie:

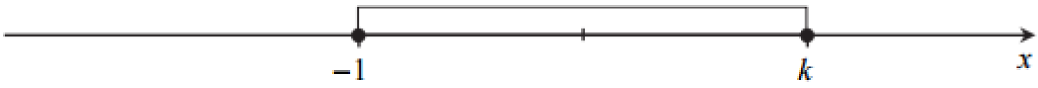
```
p1 = vector((-2,1))
p2 = vector((-1,3))
show(-1/2*(p1+p2) )

plt = point( p1,size=40)+point(p2 ,size=40)
plt += point(1/2*(p1+p2),size=40,color='red')
plt += point(-1/2*(p1+p2),size=40,color='green')
plt.show()
```



3.16 Nierówności - graficznie!

Zadanie 1. (0–1)

Na rysunku przedstawiony jest zbiór wszystkich liczb rzeczywistych spełniających nierówność $|2x-8|\le 10$.



Stąd wynika, że

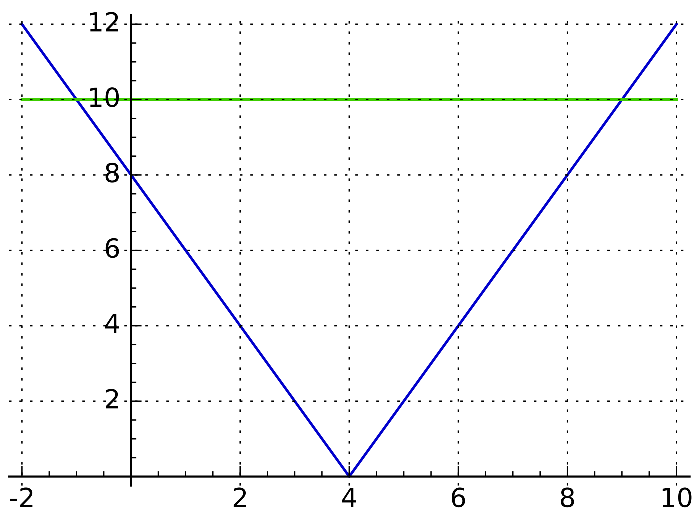
A. $k=2$ **B.** $k=4$ **C.** $k=5$ **D.** $k=9$

Poniższy kod poda nam odpowiedź:

```
show( solve(abs(2*x-8)<=10,x))
```

Dla bliższego zrozumienia dlaczego, możemy narysować prawą i lewą stronę nierówności. Kiedy będzie $|2x-8|$ mniejsze od 10?

```
plot([abs(2*x-8),10],(x,-2,10),figsize=4,gridlines=True)
```



Z wykresu łatwo odczytać wynik badanej nierówności.

3.17 Pomocna ręka wykresu

Zadanie 2. (0–1)

Dana jest funkcja f określona wzorem $f(x)=\begin{cases} x-2 & \text{dla } x\le 0 \\ \left||x+3|-4\right| & \text{dla } x>0 \end{cases}$

Równanie $f(x)=1$ ma dokładnie

A. jedno rozwiązanie.

B. dwa rozwiązania.

C. cztery rozwiązania.

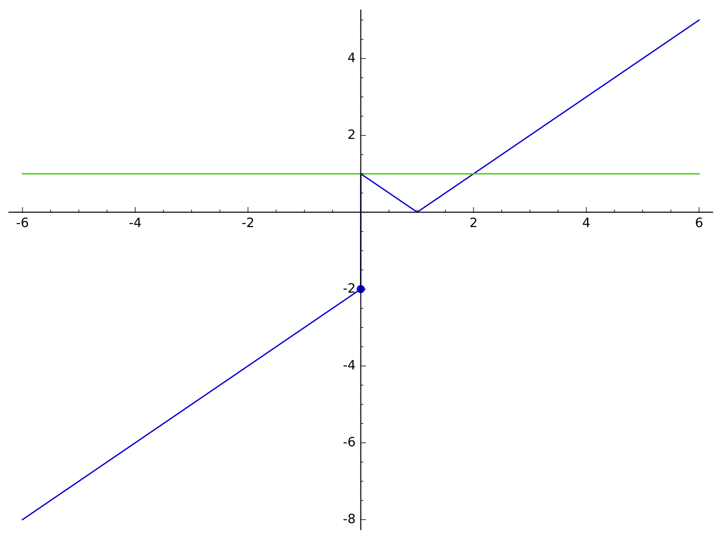
D. pięć rozwiązań.

Zadanie to można rozwiązać rysując wykres powyższych funkcji. Narysowanie wykresu może też być pomocne w sprawdzeniu rozwiązania. SageMath mamy do dyspozycji funkcję pozwalającą zapisać i operawać na funkcjach określonych na przedziałach: `piecewise`.

```
f = piecewise([
 ↪  [RealSet.open_closed(-10,0),x-2],[(0,10),abs(abs(x+3)-4)]])
plot([f(x),1],(x,-6,6)) + point((0,f(0)),size=40)
```

Przedziały domknięte i otwarte a funkcja `piecewise`

Zwróćmy uwagę, że korzystając z funkcji code:*piecewise* możemy podać przedział otwarty `(1,2)` lub domknięty `[1,2]`. Jednak przedział jednostronnie domknięty, np. `(1,2]` stanowił by zapis niepoprawny z punktu widzenia języka Python. Dlatego w takim przypadku należy zastosować konstuktor `RealSet.open_closed(1,2)`.



Z wykresu łatwo odczytać liczbę rozwiązań równania.

3.18 Ławiej powiedzieć niż zrobić!

Zadanie 8. (0–3)

Udowodnij, że dla każdej liczby rzeczywistej x prawdziwa jest nierówność

$$x^4 - x^2 - 2x + 3 > 0.$$

Wystarczy rozwiązać rówanie. Nierówność będzie spełniona dla każdego x, jeżeli równanie nie ma pierwiastków rzeczywistych i jest spełnione da jakiegokolwiek x. To ostatnie widać np. dla $x = 0$

Nie potrafimy w pamięci rozwiązywać równań czwartego stopnia, więc stosujemy Sage:

```
f(x) = x^4 -x^2 -2*x +3
for s in f(x).solve(x):
    print s.rhs().n()
```

Otrzymujemy:

```
-1.07095068169267 + 1.05537210011239*I
-1.07095068169267 - 1.05537210011239*I
1.07095068169267 - 0.424335310268147*I
1.07095068169267 + 0.424335310268147*I
```

I koniec, wszystkie pierwiastki są zespolone!

Można by się jeszcze zastanowić, dlaczego przybliżyliśmy rozwiązanie? Skasujmy `.n()` na końcu i zobaczmy co wyjdzie.

Niestety okazuje się, że część urojona jest przedstawiona wyrażeniami zawierającymi i co nie daje gwarancji jest urojona. Spróbujmy dopisać więc `.full_simplify()`. Wzory mają ok 0.5m długości, ale przynajmniej mamy matematycznie dokładną odpowiedź.

Hmm - a czy nie dało by się inaczej? Oczywiście. Ponieważ wielomian jest stopnia parzystego, to wystarczy sprawdzić, że we wszystkich minimach funkcja ma wartość dodatnią. Z Sage zajmuje to sprawdzenie tylko chwilkę:

```
f(x).diff().solve(x)
```

Otrzymujemy jeden pierwiastek rzeczywisty $x = 1$. Funkcja jest więc większa zera w zerze i posiada jedno miejsce w którym znika pochodna. Druga pochodna w tym punktcie jest dodatnia więc mamy minimum. Funkcja w minimum ma wartość $f(1) = 1$ więc

```
f(x).diff(2)(1),f(1)
```

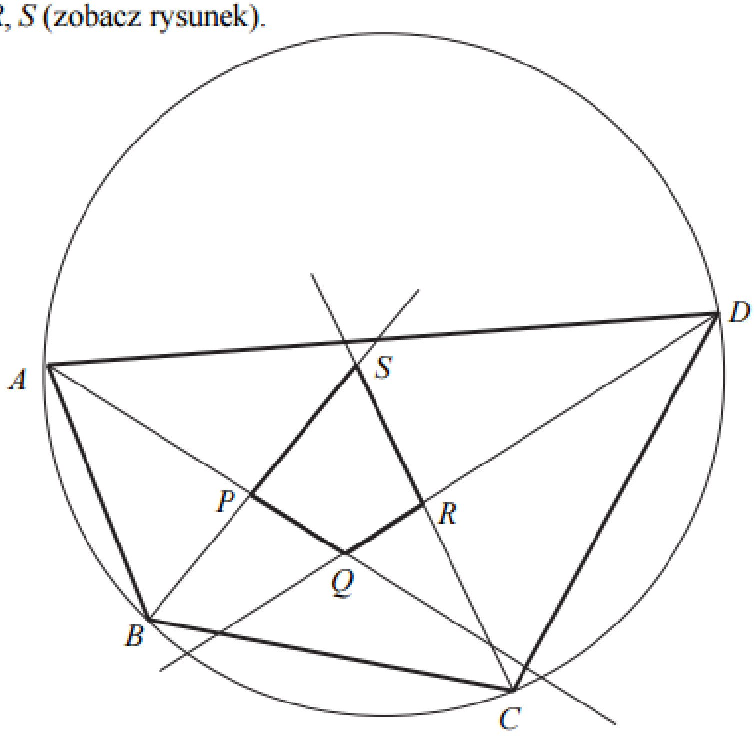
Na zakończenie narysujmy wykres by przekonać się, że nase rozważania sa poprawne:

```
plot(f(x),(x,-2,2))
```

3.19 Solve and substitute

Zadanie 9. (0–3)

Dwusieczne czworokąta $ABCD$ wpisanego w okrąg przecinają się w czterech różnych punktach: P, Q, R, S (zobacz rysunek).



Wykaż, że na czworokącie $PQRS$ można opisać okrąg.

Zadanie to nie jest trudne, ale występuje wiele zmiennych i łatwo o pomyłkę. Dlatego warto zastosować system algebry komputerowej do „czarnej roboty" i zająć się jedynie wypisaniem formuł. Pozostawiamy czytelnikowy odnalezienie w poniższym kodzie twierdzeń o sumie kątów w trójkącie i czworoboku.

```
var('a,b,c,d,p,s,r,q')
eqs = [ a/2 + d/2 + q == 180, \
        d/2 + c/2 + r == 180, \
        b/2 + c/2 + s == 180,\
        b/2 + a/2 + p ==180 ]
sol = solve(eqs,[p,s,r,q],solution_dict=True)
print (q+s).subs(sol).subs( a==(360-b-c-d) )
print (p+r).subs(sol).subs( a==(360-b-c-d) )
```

`solution_dict=True`

Zostosowanie parametru `solution_dict=True` w funkcji `solve` pozwala na otrzymanie rozwiązania w postaci Pythonowego słownika. Taka postać jest bardziej uniwersalna do dalszych przekształceń.

3.20 Czworokąt wpisany w okrąg z wykorzystaniem algebry komputerowej

Zadanie 10. (0–4)

Długości boków czworokąta $ABCD$ są równe: $|AB|=2$, $|BC|=3$, $|CD|=4$, $|DA|=5$. Na czworokącie $ABCD$ opisano okrąg. Oblicz długość przekątnej AC tego czworokąta.

3.20.1 Rozwiązanie klasyczne

Szkolny sposób rozwiązania tego zadania korzysta z dwóch twierdzeń. Po pierwsze suma kątów naprzeciwległych w czworokącie opisanym na okręgu wynosi π. Po drugie można zastosować twierdzenie cosinusów dla trójkątów ABC i ACD.

Oznaczając dla zwięzłości literami długości boków: $|AB| = a, |BC| = b, |CD| = c, |DA| = d, |AC| = x$ oraz kąty $\sphericalangle ABC = \alpha$ i $\sphericalangle CDA = \pi - \alpha$ możemy w Sage napisać dla powyższych trójkątów dwa równania:

```
var('a,b,alpha,c,d,x')
eq1 = a^2 + b^2 - 2*a*b*cos(alpha) == x^2
eq2 = c^2 + d^2 - 2*c*d*cos(pi-alpha) == x^2
table([[eq1],[eq2]])
```

Otrzymujemy:

- $-2\,ab\cos\left(\alpha\right) + a^2 + b^2 = x^2$
- $-2\,cd\cos\left(\pi - \alpha\right) + c^2 + d^2 = x^2$

Drugie równanie można uprościć i otrzymamy układ równań z dwoma niewiadomymi:

```
eq2 = eq2.simplify()
table([[eq1],[eq2]])
```

Otrzymujemy:

- $-2\,ab\cos\left(\alpha\right) + a^2 + b^2 = x^2$
- $2\,cd\cos\left(\alpha\right) + c^2 + d^2 = x^2$

Ponieważ interesuje nas tylko wartość długości przekątnej x możemy wyeliminować z powyższych równań $\cos\alpha$. Wynik otrzymamy natychmiast:

```
(eq1*c*d + eq2*a*b).solve(x)[1].rhs().subs({a:2,b:3,c:4,d:5})
```

Otrzymujemy: *sqrt(253/13)*

Jak to działa?

Wykonaj powyższe polecenie obcinając kolejne wyrażenia po „kropkach” od lewej by przekonać się jak to działa krok po kroku:

```
(eq1*c*d + eq2*a*b)
(eq1*c*d + eq2*a*b).solve(x)
(eq1*c*d + eq2*a*b).solve(x)[1]
(eq1*c*d + eq2*a*b).solve(x)[1].rhs()
```

3.20.2 Rozwiązanie alternatywne

Przypuśćmy jednak, że nie pamiętamy ani twierdzenia cosinusów ani nie znamy własności czworokątów wpisanych w okrąg. Można by pokusić się o napisanie układu równań spełnionych przez współrzędne wszystkich punktów oraz promień okręgu, który jest też nieznany!. W sumie mamy $8 + 1 = 9$ niewiadomych! Wynika z tego, że będziemy potrzebowali dziewięciu równań. Współrzędne każdego z punków spełniają równanie okręgu, co daje nam już cztery zależności. Następnie, ponieważ znamy odleglości pomiędzy kolejnymi wspólrzędnymi to mamy znowu cztery równości. Brakuje jeszcze jednej. Zauważmy, że nasz czworokąt wpisany w okrag możemy obracać o dowolny kąt względem środka okręgu. Wybierzmy tylko jedną orientację - na przykład taką w której pierwszy punkt leży na osi X - co nam da brakujące równanie $y_0 = 0$.

Wszyskie te rówania zapiszemy od razu w Sage:

```
var('r,x0,y0,x1,y1,x2,y2,x3,y3')
eqs=[x0^2+y0^2==r^2,\
 x1^2+y1^2==r^2,\
 x2^2+y2^2==r^2,\
 x3^2+y3^2==r^2,\
  (x1-x0)^2+(y1-y0)^2==2^2,\
  (x2-x1)^2+(y2-y1)^2==3^2,\
  (x3-x2)^2+(y3-y2)^2==4^2,\
  (x0-x3)^2+(y0-y3)^2==5^2,\
 y0==0]

table(list(enumerate(eqs)))
```

Otrzymujemy:

- :math:‘ x_{0}^{2} + y_{0}^{2} = r^{2} ‘,
- :math:‘ x_{1}^{2} + y_{1}^{2} = r^{2} ‘,
- :math:‘ x_{2}^{2} + y_{2}^{2} = r^{2} ‘,
- :math:‘ x_{3}^{2} + y_{3}^{2} = r^{2} ‘,

- :math:‘ {left(x_{0} - x_{1}right)}^{2} + {left(y_{0} - y_{1}right)}^{2} = 4 ‘,
- :math:‘ {left(x_{1} - x_{2}right)}^{2} + {left(y_{1} - y_{2}right)}^{2} = 9 ‘,
- :math:‘ {left(x_{2} - x_{3}right)}^{2} + {left(y_{2} - y_{3}right)}^{2} = 16 ‘,
- :math:‘ {left(x_{0} - x_{3}right)}^{2} + {left(y_{0} - y_{3}right)}^{2} = 25 ‘,
- :math:‘ y_{0} = 0 ‘

Pozostaje rozwiązać układ dziewięciu równań wielomianowych i otrzymamy rozwiązanie zadania. Bez pomocy algebry komputerowej powyższy układ równań nie wygląda zachęcająco. Okazuje się, że nawet dla komputera jest on problemem i wymaga dość wyroafinowanych technik. Jednak po chwili otrzymamy wynik:

```
sols = solve(eqs,[x0,y0,x1,y1,x2,y2,x3,y3,r],solution_dict=True)
print ((x0-x2)^2+(y0-y2)^2).subs(sols[0]).canonicalize_radical()
```

Otrzymujemy: *253/13*

Udało nam się otrzymać rozwiązanie (ściśle mówiąc kwadrat rozwiązania)!

ROZDZIAŁ 4

Zakończenie

4.1 Warunki wdrożenia produktu, czyli jak odnieść sukces

Zdobyte podczas realizacji prezentowanego projektu doświadczenia, konsultacja ze środowiskiem akademickim a także przeprowadzona ewaluacja pozwoliła nam wysnuć wnioski dotyczące uwarunkowań wdrożeniowych opracowanej w projekcie metodologii. W celu określenia mocnych i słabych stron a także szans i zagrożeń dotyczących możliwości wdrożenia projektu do praktyki szkolnej przeprowadzona została analiza SWOT powodzenia przedsięwzięcia. Została ona sformułowana na podstawie sumy doświadczeń uczestników projektu, analizy uwarunkowań pochodzących od lokalnego środowiska szkolnego i zewnętrznego środowiska oświatowego. Wyróżniliśmy bardzo wiele czynników wpływających na potencjalny sukces przedsięwzięcia, rozumiany jako wdrożenie opracowanej w ramach projektu metodologii w taki sposób, aby stała się w szkole stałym elementem wykorzystywanym podczas zajęć przedmiotów ścisłych.

Choć SageMath jest bardzo przydatnym narzędziem w szkole średniej, to jednak bardzo słabo spopularyzowanym. Z ankiet ewaluacyjnych wynika, że uczniowie wcześniej o nim nie słyszeli. Nie słyszeli o SageMath także nauczyciele. Nasze działania w ramach zrealizowanego projektu mają na celu tę sytuację zmienić.

Zainteresować ucznia nie będzie trudno ponieważ uczniowie są osadzeni w świecie technologii komputerowej i informacyjnej, nie mają w tym zakresie obaw, łatwo poznają nowe aplikacje. Pokazały to wywiady, ankiety ewaluacyjne i doświadczenia nauczycieli realizujących projekt.

Z badań ewaluacyjnych wynika, że 80% uczniów bardzo pozytywnie oceniło SageMath. Uczniowie uznali narzędzia SageMath za bardzo przydatne, podkreślali dostępność SageMath on-line, intuicyjne polecenia, dużo źródeł internetowych, pomoc w rozwiązywaniu prac domowych, nauki na poziomie zaawansowanym czy rozwijania zainteresowań.

Jednak aby uczeń posiadł umiejętność posługiwania się narzędziami SageMath, powinien je znać nauczyciel, który ucznia zaciekawi i zmotywuje do stosowania SageMath. Z kolei aby nauczyciel posiadł umiejętność korzystania z narzędzi SageMath, także powinien zostać zaciekawiony, zmotywowany i przekonany. Bazując na doświadczeniu nauczycieli, którzy wzięli udział w projekcie, a którzy wcześniej o SageMath nie słyszeli wiemy, że ocenili bardzo wysoko przydatność SageMath do nauczania swojego przedmiotu. W wywiadach podkreślali takie zalety SageMath jak: oprogramowanie darmowe, dostępne on-line, z możliwością pracy na własnym serwerze, na którym można zapisywać prace,publikować materiały i oceniać prace uczniów. Dostrzegli potencjał SageMath do zastosowania w edukacji szkolnej i pozaszkolnej czyli możliwości: szybkiego wykonywania obliczeń dla wielu parametrów, graficznego prezentowania wyników, ułatwiającego zrozumienie tematu, dokonania pogłębionej analizy problemów, poznawania treści wykraczających poza program nauczania, czyli narzędzia także dla ucznia do samokształcenia, rozwijania zainteresowań, przygotowywania do olimpiad i konkursów przedmiotowych. Nauczyciele uznali SageMath także za bardzo dobre narzędzie do formułowania uogólnień na podstawie szybkiego badania zagadnienia dla wielu danych wejściowych, rozwiązywania problemów dla wielu przypadków, szybkiej weryfikacji prawidłowości rozwiązań, możliwości przygotowywania prac klasowych i ćwiczeń na zajęcia, warsztaty i konkursy.

Ale aby tak powiedzieć o SageMath, najpierw nauczyciele sami musieli przejść drogę poznawania nowego oprogramowania. Wiadomo, że “początki są trudne”- nauczyciele wielokrotnie powtarzają to uczniom. Ostatecznie czas potrzebny na podstawy SageMath, został przez nauczycieli oceniony na około 10 godzin i nie uznali oni tego oprogramowania za bardzo trudne w poznawaniu. Te stwierdzenia, badane podczas ewaluacji, nasuwają wniosek, że nauczyciele, którzy poznali SageMath wyrazili bardzo pozytywne opinie łącznie z przekonaniem, że będą SageMath stosować zarówno na swoje potrzeby jak i na lekcjach i zajęciach pozalekcyjnych.

A co z nauczycielami, którzy sami nie dotknęli tego oprogramowania, a jedynie o nim słyszeli (na przykład bezpośrednio przy prowadzeniu wywiadu)?. Tu sprawa nie wygląda tak pozytywnie, ponieważ nieznane nie budzi zaufania. Choć przeważały postawy zaciekawienia - około 60% i postawy sceptyczne - około 20%, była jednak także spora grupa przeciwników - około 20%. Ci ostatni nie chcą tracić czasu na nowe technologie, są zwolennikami obliczeń na kartce, uważają, że uczeń straci zainteresowanie zagadnieniem, skoro od razu może mieć wynik.

Wśród nauczycieli zaciekawionych SageMath znalazły się osoby zainteresowane innowacjami, awansem zawodowym, aktywnymi metodami nauczania, osoby obawiające się o utratę godzin w związku z reformą, osoby chcące zmienić szkołę na stopień wyższą a przede wszystkim osoby młodsze. Zatem to jest grupa docelowa nauczycieli, do której staramy się dotrzeć w naszych publikacjach, konferencjach i propagowaniu rezultatów projektu. I dlatego ogromnie ważnym elementem wpływającym na wdrożenie projektu do praktyki szkolnej jest rozpowszechnianie jego rezultatów. Przygotowaliśmy metodykę nauczania matematyki, fizyki i informatyki z wykorzystaniem SageMath. Przygotowaliśmy materiały w postaci scenariuszy lekcji wraz z opisem teoretycznym i ćwiczeniami do warsztatów, uwagi metodyczne, instrukcje do projektów wraz z systemem oceniania, i przykładowymi pracami uczniów, relacje z zajęć, doświadczenia nauczycieli oraz wyniki ewaluacji. Wszystkie materiały są opublikowane i są gotowe do bezpośredniego użycia przez nauczycieli.

Ogromnie ważną grupą z punktu widzenia warunków wdrożenia rezultatów projektu jest także otoczenie edukacyjne zewnętrzne nauczyciela. Czyli po pierwsze inni nauczyciele pracujący w tej samej szkole. A wśród nich nauczyciele pokrewnych przedmiotów ścisłych, nauczyciele informatyki, czy nauczyciele, którzy dzięki swojej aktywności w postaci udziału w szkoleniach, konferencjach czy samokształceniu, mogą zachęcić swoich kolegów do zastosowania nowych technologii.

Kolejny krąg otoczenia to dyrekcja szkoły, która ma ogromny wpływ na motywowanie nauczycieli do rozwoju zawodowego np. opracowania autorskiego programu nauczania, opracowania innowacji, udziału w szkoleniach, zastosowania aktywnych i innowacyjnych metod nauczania - z wykorzystaniem nowych technologii, organizacji koła olimpijskiego czy indywidualnej pracy z uczniem zdolnym.

Następne bardzo ważne ogniwo to przedmiotowi doradcy metodyczni. W ramach swoich zadań organizują dla nauczycieli spotkania metodyczne, warsztaty, konsultacje czyli są nośnikami informacji i z pewnością będą zainteresowani metodami wykorzystywania nowych technologii, z których SageMath jest doskonałym przykładem. Zadbaliśmy, aby zaprosić doradców przedmiotów ścisłych na organizowane w ramach projektu konferencje.

Idąc dalej, powiadomiliśmy o naszych konferencjach ośrodki szkolące nauczycieli. Z pewnością warto, aby SageMath znalazł się w ofercie szkoleniowej i będziemy o to zabiegać. Wielu nauczycieli będzie zainteresowanych takimi szkoleniami. Nasze badania ewaluacyjne to pokazały. Nauczyciele słysząc o SageMath byli zainteresowani tym narzędziem, dopytywali czy są kursy z tego zakresu i przyznawali, że wolą ukończyć kurs, niż samodzielnie się nowego oprogramowania uczyć.

Sięgając jeszcze dalej w głąb struktury ośrodków edukacji mamy poziom programów nauczania i innowacje. Zgodnie z przepisami Prawa Oświatowego, nauczyciel, czy grupa nauczycieli może opracować innowację pedagogiczną, przekazać jej założenia na posiedzeniu szkolnej Rady Pedagogicznej, a następnie po uzyskaniu zgody tejże rady, zarejestrować innowację w odpowiednim Kuratorium Oświaty. Takie innowacje po ewaluacji i ewentualnej modyfikacji mogą zostać wprowadzone na stałe do programu szkolnego i dzięki temu stać się elementem wykorzystywanym przez wszystkich nauczycieli danego przedmiotu.

Bardzo dobrym momentem na wdrażanie naszego projektu jest czas reformy, kiedy nauczyciele już "nie mogą spać spokojnie", niektórzy przejdą do innej szkoły inni zmierzą się z nowymi warunkami zatrudnienia, a wszyscy zmierzą się z nowym programem. Zatem trzeba być konkurencyjnym. Do dobra okazja do odkurzenia starych nawyków, np. w postaci wprowadzenia nowych metod nauczania, wprowadzenia innowacji, stworzenia kółka "dla zdolnych" czy innej aktywności. Czyli znajdą się nauczyciele zainteresowani, będzie także bogata oferta szkoleń dla nauczycieli informatyki, obiecanych przez MEN, będzie też regulacja zwiększająca godziny na przedmioty ścisłe, (na informatykę w szczególności), zarówno w szkole podstawowej jak i liceum. To bardzo dobra perspektywa do wdrożenia rezultatów prezentowanego projektu.

Formułując odpowiedź na pytanie "Jakie warunki muszą być spełnione aby odnieść sukces", prześledziliśmy drogę, jaką musi przebyć informacja, jakimi drogami może dotrzeć do nauczyciela, i jakie czynniki będą wpływały na powodzenie wdrożenia rezultatów naszego projektu. Zadaliśmy sobie następujące pytania:

Nauczyciel Skąd się dowie o SageMath? -> popularyzacja metodyki w mediach, konferencje, inny nauczyciel, oferta ośrodka szkoleniowego, metodyk przedmiotowy, Jak się przekona, że warto? -> pokazy możliwości SageMath, warsztaty, motywacja nauczyciela do aktywności, otwartość, zachęta ze strony innego nauczyciela lub metodyka przedmiotowego, dyrekcji szkoły, reforma i związana z nią zmiana pracy, reforma i związana z nią zmiana programów nauczania. Jak się nauczy? samokształcenie, pokaz np nauczyciela informatyki, kurs, konferencja, publikacja Gdzie przeprowadzi zajęcia? -> czy ma dostęp do pracowni komputerowej, może wykorzysta smartfony uczniów? Jak, do czego wykorzysta? -> wybierze z programu nauczania treści, które współpraca z nauczycielami przedmiotu, innowacja, projekty, konkursy, kółko, praca z uczniem zdolnym, przygotowanie uczniów do olimpiady przedmiotowej,

Dyrektor -> Zachęci, Pomoże, Wyśle na kurs?

Metodycy przedmiotów ścisłych -> czy przyjdą na konferencję, będą popularyzowali?

Ośrodki szkolące nauczycieli -> czy coś wie o SageMath, czy przyjmie zaproszenie na konferencje, czy zorganizuje kurs?

Ministerstwo -> Czy zapewni w siatce godzin odpowiednią rezerwę na wdrażanie SageMath? - tak, nauczyciel ma do zrealizowania obowiązkowo podstawę programową, a rozszerzenia planuje sam (ewentualnie w porozumieniu z zespołem przedmiotowym) i ma na to godziny, bo podstawa programowa ma charakter ogólny, nauczyciel sam opracowuje treści i tematy.

Czy programy nauczania będą odpowiednio elastyczne? Program nauczania nauczyciel wybiera sam (ewentualnie w porozumieniu z zespołem przedmiotowym). Wybiera (dla danego przedmiotu) z zestawu gotowych programów , zatwierdzonych przez MEN lub opracowuje sam jako program autorski (tylko do zatwierdzenia przez Radę Pedagogiczną w szkole). Zatem wszystko zależy od nauczyciela !!

4.2 Analiza SWOT

Mocne strony Strengths

1. Upowszechnianie stosowania techniki komputerowej i informacyjnej podczas zajęć szkolnych.
2. Promocja otwartego oprogramowania inżynierskiego.
3. Możliwość zastosowania przez nauczyciela mieszanych narzędzi dydaktycznych (z metodami aktywnymi) gwarantujących większe zainteresowanie ucznia lekcją, więcej samodzielności dla ucznia.
4. Duże możliwości SageMath i Python do wykorzystania podczas zajęć wielu przedmiotów (matematyka, fizyka, informatyka, chemia, biologia, geografia)
5. Łatwa w zastosowaniu wizualizacja obliczeń w postaci różnorodnych wykresów i animacji,
6. Dostępność SageMath on-line.
7. Możliwość wykorzystania urządzeń mobilnych podczas lekcji (nie trzeba rezerwować pracowni komputerowej).
8. Łatwa instalacja dedykowanego serwera SageMath na potrzeby szkoły, niemal bezobsługowe administrowanie tym serwerem.
9. Dostępność SageMath w chmurze - darmowa strona sagecloud.
10. Przewaga operacyjna w stosunku do innych narzędzi (np. wolframalpha), w których można wykonać jedną operację i nie można wykonać skryptów Python.
11. Możliwość publikowania na serwerze materiałów dla uczniów.
12. Dostęp nauczyciela na bieżąco do prac uczniów na serwerze, co zapewnia monitorowanie postępów ucznia i łatwe sprawdzanie prac.
13. Możliwości wykorzystania SageMath i Python na wielu poziomach zaawansowania podczas zajęć szkolnych i pozaszkolnych (np. koła przedmiotowe, przygotowanie do konkursów).
14. SageMath i Python to dobra propozycja dla uczniów do rozwijania zainteresowań i samokształcenia.
15. Wygodne narzędzie dla nauczyciela podczas przygotowywania się do zajęć szkolnych (sprawdzanie wyników obliczeń proponowanych zadań na lekcje bądź prace klasowe).
16. Z punktu widzenia nauczyciela, który nie zetknął się z SageMath wystarczy poświęcenie około 10 godzin na poznanie podstaw i przygotowanie podstawowych zajęć.
17. Nauczyciel nie musi znać języka Python aby stosować SageMath, ale zainteresowanie SageMath może być motywacją do poznawania języka programowania Python.
18. Możliwość tworzenia dokumentacji: zbiorów zadań, skryptów, sprawozdań, raportów z wykorzystaniem systemu składu komputerowego Latex.

19. Możliwość eksportowania obliczeń i wykresów do innych aplikacji np. edytora tekstu.
20. Dużo materiałów w sieci na temat Python i SageMath, nauczyciel może prowadzić łatwo samokształcenie.
21. Przygotowanie w ramach niniejszego projektu sprawdzonych materiałów dla nauczyciela do bezpośredniego wykorzystania podczas lekcji.
22. Prezentacja metodyki, zewaluowanej, skonsultowanej ze środowiskami akademickimi gwarantującej nauczycielowi osiąganie założonych celów edukacyjnych.

Szanse w otoczeniu Opportunities

1. Rozwój techniki komputerowej i informacyjnej.
2. Nacisk na przedmioty ścisłe (matematyka, informatyka) w szkole podstawowej i 4. letnim liceum.
3. Wymiana kadr: młodsi nauczyciele bardziej zainteresowani wykorzystaniem oprogramowania komp. podczas zajęć.
4. Awanse zawodowe - nauczyciele poszukują innowacyjnych rozwiązań.
5. Python jako środowisko do wyboru na egzamin maturalny z informatyki.
6. Więcej godzin informatyki, w tym programowania w szkole podstawowej.
7. Zapowiedź szkoleń nauczycieli informatyki w związku z reformą systemu edukacji.
8. Zainteresowanie dyrekcji szkół innowacyjnymi metodami nauczania.
9. Instalacja szkolnego serwera SageMath zapewnia dostęp do SageMath wielu nauczycielom w szkole.
10. Pozytywne doświadczenia nauczycieli wykorzystujących SageMath zostaną przekazane dyrekcji szkoły i rozpropagowane.
11. Aktywny nauczyciel może zainteresować innych np. organizując w szkole warsztaty z pokazem możliwości SageMath dla nauczycieli.
12. Możliwość podejmowania współpracy nauczycieli różnych przedmiotów w realizowaniu wspólnej partii materiału.
13. Możliwość sprawnego realizowania projektów o szerszym charakterze niż zajęcia szkolne, np. udział w konkursach dla młodych naukowców, realizacja projektów międzyprzedmiotowych wymagających np. wykonania symulacji, badań i eksperymentów z pomiarami i obliczeniami.
14. Pokazanie uczniom, że ich telefony komórkowe mogą być użyte do obliczeń i nauki (nie tylko do portali społecznościowych).
15. Zainteresowanie ze strony uczniów (pomoc przy pracach domowych, samokształceniu, przygotowaniu do olimpiad i konkursów przedmiotowych)
16. Dotarcie akcją promocyjną (konferencje, publikacje) do wielu aktywnych nauczycieli, dyrektorów szkół, metodyków przedmiotowych, ośrodków szkolących nauczycieli.

17. Możliwość kontaktu zainteresowanych dyrektorów szkół, nauczycieli czy ośrodków szkolących, zarówno z nauczycielami uczestniczącymi w projekcie jak i nauczycielami akademickimi z ICSE.
18. Możliwość korzystania z opublikowanych materiałów przygotowanych dla nauczycieli i uczniów w ramach niniejszego projektu.
19. Możliwość korzystania z opublikowanych materiałów na stronach ICSE - bardzo bogate materiały o zróżnicowanym poziomie.

Słabe strony Weaknesses

1. Nauczyciel musi poświęcić czas na samokształcenie zanim zacznie stosować SageMath.
2. Jeśli nauczyciel nie programuje w żadnym języku, zastosowanie Python będzie wymagało dużo pracy.
3. Nauczyciele z długim stażem pracy niechętnie wykorzystują nowe technologie i niechętnie się szkolą.
4. Stwarzanie trudnej sytuacji dla ucznia, który nie ma smartfona czy tabletu, jeśli nauczyciel zaplanuje lekcję i dyskretnie wcześniej nie sprawdzi, czy wszyscy uczniowie mają urządzenia mobilne, (a jeśli nie, musi zaplanować np. pracę w grupach bądź w pracowni komputerowej).
5. Obawa nauczyciela, że uczniowie nie będą chcieli w zeszycie wykonywać zadań, które mogą wykonać szybciej, nie angażując swojego zainteresowania etapami obliczeń.
6. Obawa nauczycieli, że uczeń korzystający z gotowych wyników obliczeń, nie uczący się na swoich błędach będzie mniej sprawny w obliczeniach.

Zagrożenia w otoczeniu Threats

1. Ogniskowanie zainteresowania nauczycieli na podstawę programową.
2. Zmiana programów nauczania i planu realizacji materiału przez nauczyciela w 4. letnim liceum.
3. Brak dostępnych szkoleń SageMath.
4. Trudność w uzyskaniu dla nauczyciela dostępu z domu do serwera SageMath.
5. Szkoła nie refunduje nauczycielowi zakupu komputera.
6. Nie wszyscy uczniowie mają urządzenia mobilne.
7. Nie zawsze dostępna pracownia komputerowa.
8. Częste awarie darmowego serwera on-line sagecloud
9. Konieczność inwestycji w szkolny serwer SageMath, (który daje pewność, że uda się przeprowadzić planowane zajęcia).
10. Konieczność administrowania serwerem SageMath.

11. Skarżenie się nauczycieli na zbyt dużą ilość obowiązków szkolnych i w związku z tym brak czasu na wprowadzanie nowych elementów czy samokształcenie.

4.3 Ewaluacja projektu

Celem ewaluacji była:

- ocena skuteczności i efektywności działań przeprowadzonych w ramach projektu,
- ocena stopnia realizacji założonych celów
- analiza i ocena rzeczywistych efektów przeprowadzonych działań
- sprawdzenie oczekiwań grupy docelowej. Określony poziom oczekiwań grup pozwala określić metody pracy w projekcie
- sprawdzenie prawidłowości realizacji projektu zgodnie z założeniami
- badania uczestników projektu, zadowolenia z udziału w projekcie
- trwałości rezultatów.

Na potrzeby badania zostały zastosowane następujące kryteria ewaluacyjne:

1. Kryterium trafności - relevance (cele a potrzeby) - rozumiane jako zbadanie, czy działania odpowiadają na rzeczywiste potrzeby grupy docelowej
2. Kryterium efektywności - (effectiveness) (nakłady a efekty) - zbadanie, czy działania są bardziej efektywny, wydajne finansowo od dotychczas stosowanych rozwiązań,
3. Kryterium skuteczności - (utility) (plan a wykonanie) - zbadanie, czy podejmowane działania są skuteczne
4. Kryterium użyteczności - (efficiency) - (efekty a potrzeby) - zbadanie, czy istnieje możliwość praktycznego wykorzystania zdobytej wiedzy / umiejętności
5. Kryterium trwałości - (durability) - zbadanie, w jakim stopniu oraz pod jakimi warunkami prawdopodobne jest utrzymanie rezultatów po zakończeniu projektu

Badanie ewaluacyjne realizowane było przy wykorzystaniu istniejących danych i dokumentów (dane zastane), a także poprzez badania empiryczne, w trakcie których gromadzone są dodatkowe dane ilościowe i jakościowe (dane wywołane). Pozwoliło to na uzyskanie komplementarności danych, ich wzajemne uzupełnianie się i wewnętrzną weryfikację uzyskiwanych odpowiedzi. Zastosowana technika triangulacji metodologicznej umożliwiło uzyskanie bardziej obiektywnego i pełnego obrazu badanych zjawisk. W badaniu ewaluacyjnym stosowano:

- triangulacja źródeł danych - analizowano zarówno dane zastane, jak i dane wywołane,
- triangulacja metod badawczych - łączenie różnych metod badawczych (badania ilościowe oraz jakościowe) po to, by wyeliminować ograniczenia każdej z nich.

W niniejszej ewaluacji wykorzystywano różne źródła danych:

- dane zastane - analizie desk research poddane zostają dokumenty programowe, określające warunki zewnętrzne realizacji projektu, dokumentacja wewnętrzna zgromadzona przez zespół projektowy w trakcie realizacji poszczególnych działań, wyniki monitoringu,
- dane wywołane

- dane pozyskane
- wyniki ankiet i wywiadów standaryzowanych. Wykorzystano m.in. następujące narzędzia:
- tabela ewaluacji projektu
- ankiety ex ante i ex post dla uczniów (w polskiej i angielskiej wersji językowej).
- ankiety sprawdzające przyrost wiedzy
- ankiety dotyczące udziału w konferencjach, warsztatach upowszechniających (dot. m.in. ogólnej oceny konferencji, stopnia spełnienia oczekiwań, zainteresowania wykorzystaniem rezultatów projektu).

Każdy z partnerów dostosował narzędzia ewaluacyjne do swoich potrzeb. Szczegółowy związane z wynikami ewaluacji zawarte są w raporcie z ewaluacji. Poniżej zawarto najważniejsze wnioski związane z wpływem projektu na proces dydaktyczny.

Badania przeprowadzone wśród uczniów 2 AZSO w Chorzowie

Ankieta ex post została przeprowadzona na wszystkich uczestnikach ostatniego rocznika projektu iCSE4school, realizujących fizykę rozszerzoną na poziomie rozszerzonym (16 osób). W ostatnich miesiącach trwania projektu (lato 2017) byli uczniami drugiej klasy o profilu matematycznym. Cały oddział liczy 34 osoby - pozostali wybrali rozszerzenia z innych przedmiotów przyrodniczych. W klasie pierwszej wszyscy zostali przeszkoleni z podstaw programowania w języku Python. W klasie drugiej uczestniczyli w dodatkowych zajęciach dotyczących zastosowań języka Python na matematyce i fizyce.

Na pytanie: Jakiego narzędzia użyłbyś do rozwiązania złożonego problemu matematycznego lub fizycznego? najwięcej osób wybrało Sage, Python i Geogebra (po 9 głosów) oraz C/C++ (7 osób). Oznacza to, że że ponad połowa ankietowanych przekonała się do języka Python w środowisku SageMath.

Na pytanie Czy wykorzystujesz oprogramowanie otwarte (Open Source)? 8 osób odpowiedziało, że tak, jedna nie, a pozostali przyznali, że nie wiedzą, co to jest otwarte oprogramowanie.

Kolejne pytanie brzmiało “Do czego można zastosować komputer podczas lekcji matematyki lub fizyki?” 16 osób wskazało narysowanie funkcji, 15 - graficzne przedstawienie danych pomiarowych, 13 - wykonanie obliczeń arytmetycznych na podstawie wzoru, 11 - do udowodnienia twierdzenia, 8 - do wyprowadzenia wzoru. Nikt nie wybrał odpowiedzi do niczego. Wynika z tego, że uczniowie widzą w SageMath dobre narzędzie do rozwiązywania wielorakich problemów.

Na pytanie „Czy napisanie algorytmu może się przydać na lekcji fizyki?” kilkanaście osób wskazało przydatność algorytmów w rozwiązywaniu równań ruchu, w upraszczaniu wyrażeń oraz w automatyzowaniu zbierania danych eksperymentalnych. Nikt nie zaznaczyła żadnej z dwóch możliwych odpowiedzi negatywnych. Niemal wszyscy prawidłowo rozumieją pojęcie system algebry komputerowej. Większość uczniów prawidłowo definiuje metody numeryczne.

Niemal wszyscy wiedzą, jaki język programowania jest podstawą systemu SageMath.

Posługiwanie się środowiskiem SageMath uczniowie określili jako średnio trudne.

Czy komputer (z Sage) był dla Ciebie przydatny na lekcji matematyki w ...? W rysowaniu funkcji, rysowaniu wykresów danych i algorytmicznym badaniu zjawisk nierozwiązywalnych klasycznie większość uczniów zaznaczało Bardzo przydatny. W przekształceniach algebraicznych, obliczeniach numerycznych i podstawianiu liczb do wzorów najwięcej głosów zebrała opcja Trochę przydatny.

Na analogiczne pytanie dotyczące fizyki, uczniowie we wszystkich kategoriach zaznaczali najczęściej opcję Bardzo przydatny. Wszyscy korzystali ze źródeł internetowych podczas posługiwania się SageMath, przy czym 13 osób korzystało ze źródeł polskich i angielskich, a 3 tylko z polskich.

Prawie wszyscy odkryli samodzielnie jakieś nowe możliwości SageMath. Prawie wszyscy korzystają z SageMath w domu. Za najważniejsze cechy SageMath ankietowani uważają: darmowość (16 głosów), możliwość tworzenia wykresów (14), dostępność przez przeglądarkę internetową (13), otwartość (Open Source) (12), dostępność algebry komputerowej (11), możliwość programowania w Python (10). Najmniej osób wybrało łatwość tworzenia @interact (7). 11 osób deklaruje, że będzie korzystać z SageMath przy rozwiązywaniu zadań domowych z matematyki i fizyki, 5 osób jest przeciwnego zdania. Na pytanie Czy Twoim zdaniem Sage powinien być powszechnie stosowany w szkołach na lekcjach przedmiotów ścisłych? 14 osób odpowiedziało tak, 1 nie i 1 nie wiem. Ostatnie pytanie brzmiało: Oceń w skali 1-6 przydatność Sage dla Ciebie teraz i w przyszłości. Większość osób (11) zaznaczyło cyfrę 4, 2 osoby cyfrę 5 a po jednej: cyfry 1, 3 i 6.

Z przedstawionych danych wynika, że uczniowie poznali wszechstronne narzędzie, z którego potrafią zrobić wieloraki użytek i doceniają jego wartość w wielu aspektach.

Zdecydowana większość uczniów sądzi, że będzie korzystała z oprogramowania SageMath do rozwiązywania zadań z prac domowych z matematyki i fizyki oraz, że narzędzie to powinno być powszechnie stosowane w szkołach na lekcjach przedmiotów ścisłych. O korzyściach, jakie uczniowie wynieśli z realizacji projektu, można się też dowiedzieć z ich wypowiedzi przed kamerą podczas kręcenia filmu podsumowującego projekt (Youtube: https://www.youtube.com/embed/BAUCbMXWceI). Uczniowie wskazywali, że dzięki programowaniu, nauczanie przedmiotów ścisłych było mniej "suche" i abstrakcyjne, bardziej namacalne, łatwiej było pewne rzeczy zobaczyć lub sobie je uzmysłowić. SageMath zdejmował z uczniów żmudny obowiązek wykonywania czynności mało twórczych i powtarzalnych (takich jak obróbka danych i rysowanie wykresów) na rzecz myślenia koncepcyjnego, szybkiego przeprowadzania symulacji poprzez odpowiedni dobór parametrów i sposobu działania algorytmu. Dzięki Pythonowi matematyka i fizyka "ożyły", pojawiła się na przykład możliwość usłyszenia funkcji jako wykresu fali dźwiękowej, a czynności, które tradycyjnie zajmowałyby dużo czasu, udało się zredukować do napisania kilku linijek prostego kodu programu.

Podsumowując, realizacja projektu przyniosła korzyści zaangażowanym w projekt nauczycielom. Wzbogacili oni swój warsztat pracy, nauczyli się nowego języka programowania i z pewnością będą wykorzystywać nabyte umiejętności w kolejnych latach. Dzięki projektowi wzrasta prestiż szkoły w środowisku lokalnym. Zakupione pomoce naukowe będą służyć długie lata. Filmy nakręcone w ramach projektu i opublikowane na Youtube stanowią trwały dorobek intelektualny szkoły, udostępniony do wykorzystania ogółowi zainteresowanych.

Badania przeprowadzone wśród uczniów XXXIII LO im. Mikołaja Kopernika

W ramach projektu icse4school zostały przeprowadzone lekcje pilotażowe, pokazy, warsztaty i zajęcia problemowe.

W celu opracowania metodyki prowadzenia zajęć z wykorzystaniem metody projektów zostały zaplanowane i przeprowadzone badania w czterech grupach uczniów z klas trzecich matematyczno-fizycznych. Projekt obejmował cykl zajęć podczas zajęć pozalekcyjnych oraz pracę własną (pracę domową). Wybrane grupy były objęte w klasie drugiej nauką algorytmiki i programowania w języku C++. Były to grupy o zrównoważonym poziomie wiedzy i zainteresowania w zakresie matematyki i informatyki. Każda z grup – A, B, C, D przeszła nieco inny rodzaj zajęć i otrzymała nieco inne zadanie.To pozwoliło przeprowadzić ewaluację i zbadać wpływ zmienionych warunków na rezultaty zajęć czyli umiejętności uczniów i poziom wykonanych przez nich prac. Ewaluacja pozwoliła także zmodyfikować materiały na podstawie uwag uczniów. Nauczyciele wnikliwie obserwowali zajęcia, formułowali wnioski i spostrzeżenia, obserwowali pracę uczniów i przeprowadzali z nimi wywiady. Na początku i na końcu kolejnego cyklu zajęć były prowadzone anonimowe ankiety, które posłużyły do opracowania kolejnych wersji materiałów i metod prowadzenia zajęć. Na podstawie wywiadów z uczniami, oceny ich zaangażowania podczas zajęć, oceny przyrostu umiejętności uczniów posługiwania się funkcjami SAGE, kreatywności i pomysłowości uczniów, radzenia sobie z postawionymi problemami, poziomu wykonanych prac, ankiet anonimowych wypełnionych przez uczniów przed rozpoczęciem zajęć i po zakończeniu zajęć wprowadzano modyfikację do wytworzonych rezultatów pracy intelektualnej.

Obserwacje podczas ewaluacji zajęć dotyczyły m.in.: stopnia zainteresowania i zaciekawienia uczniów tematem, pytań, jakie zadawali uczniowie, formy pracy wybranej przez nauczyciela, na jakie pytania nauczyciela uczniowie nie potrafili odpowiedzieć, czy uczniowie byli kreatywni, jakie mieli pomysły rozwiązania postawionych problemów, czy uczniowie sami potrafili odnaleźć w źródłach internetowych czy materiałach od nauczyciela elementy pomocne do rozwiązania problemów, czy poziom trudności wybranych zagadnień był odpowiedni, czy uczniowie wysnuli dobre wnioski z badania i rozwiązywania postawionych problemów, jakie było tempo wzrostu poziomu umiejętności uczniów po przeprowadzeniu cyklu zajęć według scenariusza, czy ilość zajęć była wystarczająca wg zaplanowanego przez nauczyciela materiału do realizacji, jakie narzędzia SageMath okazały się najczęściej wykorzystywane, o jakie warto rozszerzyć materiały dla ucznia, czy uczniowie dokonali „odkryć” jak i do czego wykorzystać inne polecenia SageMath, jeśli były zróżnicowane zajęcia dla zdolnych i mniej zdolnych uczniów, to czy były odpowiednie, czy uczniowie dzielili się wiedzą z kolegami, czy uczniowie dyskutowali między sobą na temat wyników rozwiązań zadań.

Z kolei wywiady z uczniami podczas ewaluacji zajęć dotyczyły m.in. : opinii uczniów co do przydatności dla nich narzędzi SAGE i Python (do rozwiązywania prac domowych, samokształcenia, przygotowania do konkursów przedmiotowych, rozwijania zainteresowań, przygotowania do matury); co było najbardziej interesujące, co zmieniliby, co można dodać, stopnia trudności ćwiczeń; intuicyjności w posługiwaniu się narzędziem SageMath, łatwości znalezienia materiałów i pomocy w Internecie, korzystania np. ze smartfonów do wyszukiwania informacji czy wykonywania obliczeń w SAGE, korzystania ze zdobytej wiedzy na innych przedmiotach, czy użyliby SAGE jako pierwszego wyboru do obliczeń w matematyce, fizyce czy chemii.

Ważniejsze wnioski:

Przed rozpoczęciem zajęć uczniowie w ankiecie anonimowej “przed” pisali, że posługują się jedynie arkuszem kalkulacyjnym do wykonywania wykresów i obliczeń a także, że wcześniej nie słyszeli o SageMath. Tylko kilku uczniów w każdej z grup korzystało ze strony wolframalpha.com.

Wszyscy uczniowie wykazali zainteresowanie nowym narzędziem podczas pokazów możliwości SageMath, zainteresowały ich inżynierskie zastosowania SageMath, zastosowania w fizyce, do wykonywania symulacji i animacji. Zaciekawili się projektem, który został omówiony, dostrzegli potencjał do wykorzystania SageMath w pracach domowych i projektach, na studiach. Uczniowie z grupy A, w której zostały omówione i zastosowane podczas warsztatów funkcje SAGE, takie, które mogą zostać użyte w projekcie wykazali najmniejsze zainteresowanie innymi możliwościami SAGE. Ich praca była najmniej twórcza, podeszli do projektu zadaniowo.

Uczniowie z grupy B i C nie otrzymali wsparcia w postaci omówienia niezbędnych do wykorzystania w projekcie funkcji. Samodzielnie musieli ich poszukać w internetowej dokumentacji i przykładach. Ich praca była najbardziej kreatywna, nauczyli się najwięcej. Grupa C przygotowała opisy projektu z języku angielskim.

Uczniowie z grupy D tylko częściowo otrzymali wsparcie w postaci omówienia i prezentacji potrzebnych do projektu funkcji, dodatkowo zostali objęci cyklem zajęć z tworzenia dokumentów w formacie PDF z wykorzystaniem pakietów Latex (darmowy program, system Tex do składu komputerowego).

Poziom prac uczniów obrazował ich zaangażowanie. Wszystkie grupy poradziły sobie z projektami. Część uczniów podczas badania funkcji zmieniała jej postać z powodu zbyt skomplikowanej postaci pochodnej. Ocena prac: grupa A - mało opisów lub ich brak, nie wykorzystane parametry niektórych funkcji np. plot, limit. Prace najmniej kreatywne.

Grupa B i C prace na podobnym wysokim poziomie, nie wszyscy uczniowie dokonali odkryć, ale w większości prac pojawiło się dużo elementów unikalnych. Uczniowie znacznie lepiej posługiwali się SAGE, niż pozostali. Było więcej zabawy i odkryć podczas zajęć. Opisanie badania funkcji w języku angielskim nie sprawiło uczniom trudności.

W ankiecie ewaluacyjnej “po” wszyscy uczniowie byli zadowoleni, że poznali nowe “fajne” darmowe narzędzia do codziennego wykorzystywania. Uznali, że wiele jest do odkrycia, “na pewno przyda się na studiach”. Uznali, że czują się bezpiecznie, bo zawsze można sprawdzić wyniki. Niektórzy poddawali w wątpliwość, czy nauczyciel matematyki będzie zadowolony z tego, że uczeń rozwiąże prawie każdą pracę domową z pomocą komputera.

Wszyscy uczniowie wyrazili zadowolenie, że poznali SageMath że mogą z telefonu komórkowego szybko rozwiać swoje wątpliwości podczas wykonywania obliczeń i rozwiązywania zadań. Wszyscy uczniowie docenili potencjalne możliwości zastosowania SageMath do wizualizacji rozwiązań, wykonywania symulacji eksperymentów.

Badania przeprowadzone wśród nauczycieli biorących udział w warsztatach

1. Jaka jest Pani/a zdaniem użyteczność narzędzi wypracowanych w projekcie „Zintegrowane nauczanie przedmiotów ścisłych z perspektywą obliczeniową - iCSE4school” dla ucznia od 16. do 19. roku życia zainteresowanego przedmiotami ścisłymi? Ocenę wyraź w punktach na skali 1 - 10, przy czym 10 oznacza ocenę najwyższą. średnia arytmetyczna 8,3

2. Prosimy o uzasadnienie oceny:

- Język Python nie jest znanym językiem programowania wśród nauczycieli.
- W warsztatach uczestniczyła bardzo mała liczba osób (mając do wyboru więcej opcji uczestnicy konferencji wybierali inne warsztaty). Pełne obłożenie miały w Warszawie (ogółem ponad 500 uczestników) warsztaty dotyczące podstaw programu Scratch.
- Propozycję nauki programowania i wykorzystania języka Python w nauczaniu przedmiotów ścisłych uznano za pozytywną, jednak widać że nauczyciele mają pewne obawy (nie spotkali się, więc trzeba się Pythona nauczyć; Geogebra jest lepiej znana więc uważana za bardziej przydatną; trzeba włożyć wiele pracy w przygotowanie się do zajęć).
- W warsztatach uczestniczyli w większości nauczyciele szkół ponadgimnazjalnych.
- Według stażu pracy nauczycieli zainteresowanie programowaniem w szkole rozkłada się mniej więcej równomiernie.

Podsumowując: warto zwrócić uwagę na fakt, że zdaniem badanych w działaniach iCSE4school tkwi dwutorowy potencjał - generowania współpracy nauczycieli matematyki, fizyki, informatyki oraz odniesienia się do licealnych zainteresowań (np. klas matematyczno-informatycznych). Oba te czynniki stanowią jednak swego rodzaju trudność: działania międzyprzedmiotowe uchodzą za wymagające dużego zaangażowania, a nawiązanie do uczniowskiego hobby może nauczyciela postawić w sytuacji niewiedzy. Nikt z badanych nie uczestniczył w tego typu projekcie - istotne jest zatem upowszechnianie jego idei. Zgodnie z zasadą, że zajmowanie się problemami (pozornie?) trudnymi daje satysfakcję z pracy: i uczniowskiej, i nauczycielskiej.