KRAKOWSKIE FORUM AI W EDUKACJI · OD BADAŃ DO PRAKTYKI SZKOLNEJ

UKEN, UL. INGARDENA 4, SALA 101N · 24 LISTOPADA 2026, 12:00

Jak przygotować *kompleksowe* zajęcia z AI?

Warsztat agentowy na przykładach czterech kompletnych kursów: **Claude Code** jako asystent techniczny prowadzącego



dr Bartosz Naskręcki

Wydział Matematyki i Informatyki

Uniwersytet im. Adama Mickiewicza w Poznaniu

Uniwersytet im. Adama Mickiewicza w Poznaniu

Slajdy, PDF, pełny tekst (PL i EN):

bnaskrecki.faculty.wmi.amu.edu.pl/uken2026

Prowadzący projektuje, agent buduje, *weryfikacja* rozstrzyga



Prowadzący

CEL · TREŚĆ · DECYZJE

- ścieżka kursu i ambicja merytoryczna
- wybór i głębia przykładów
- zasady oceniania, terminy
- co i kiedy publikujemy



Agent (Claude Code)

KOD · STRONY · FORMATY

- slajdy, książka, aplety, strona
- zadania Moodle z wariantami
- formatowanie, tłumaczenie, publikacja
- dokumentacja: plan, dziennik, pamięć



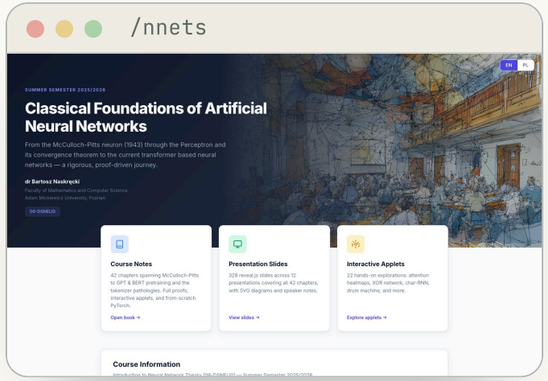
Weryfikacja

OBLICZENIA · TESTY · KONTROLE

- liczby liczone przy każdej budowie
- testy zadań i odrzucanie typowych błędów
- kontrola linków, układu, wersji językowych
- historia zmian w git

Agent zdejmuje z prowadzącego obowiązek eksperckiej wiedzy programistycznej. Czas, który zostaje, idzie w **plan kursu**, **wyrafinowane przykłady** i **szczegółowe omówienie treści**.

Ten sam warsztat, cztery kursy, *pięć wersji modelu*



WMI UAM · LUTY–MAJ 2026

Klasyczne podstawy sieci neuronowych

42 rozdziały-notatniki · **30** apletów · **12** talii slajdów · quiz i system oceniania w Moodle

76 commitów · Opus 4.6–4.7

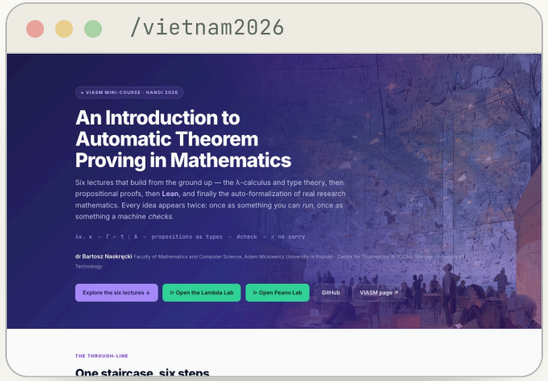


ADAPTACJA · MARZEC–CZERWIEC 2026

Elementy kryptoanalizy

45 rozdziałów w Pythonie + **24** w SageMath · **9** apletów (Enigma, Rejewski, Zygalski…) · szkielet w 1 dzień, potem audyt

Opus 4.6–4.8

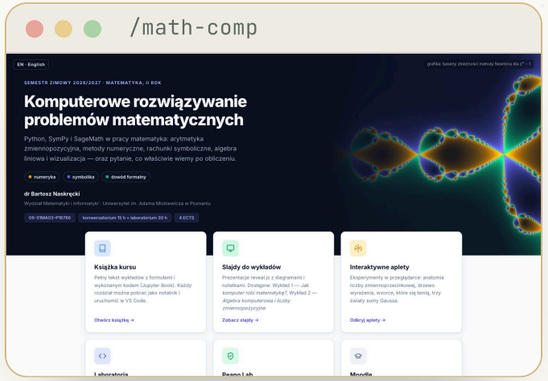


VIASM, HANOI · LIPIEC–SIERPIEŃ 2026

Automatyczne dowodzenie twierdzeń

6 wykładów · laboratoria w przeglądarce (Lambda Lab, Peano Lab) · **532** notatki w bazie wiedzy

325 commitów, 43 ze stopką Claude · Fable 5, Opus 4.8



WMI UAM · ZIMA 2026/27 · TRWA

Komputerowe rozwiązywanie problemów matematycznych

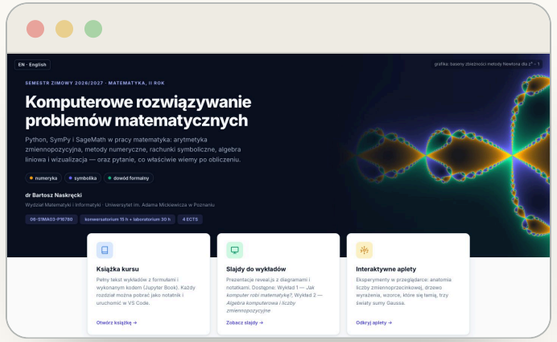
2 wykłady + szkice 6 · **4** pogłębienia · **2** moduły Moodle · wydanie angielskie

60 commitów · Opus 5.5

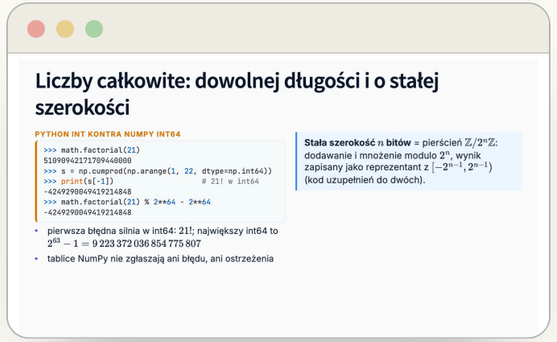
Opus 4.6 · luty	Opus 4.7	Opus 4.8	Fable 5	Opus 5.5 · listopad

Liczby policzone 24.11.2026 w repozytoriach kursów (`git log`, pliki); modele — ze stopek commitów `Co-Authored-By: Claude …`.

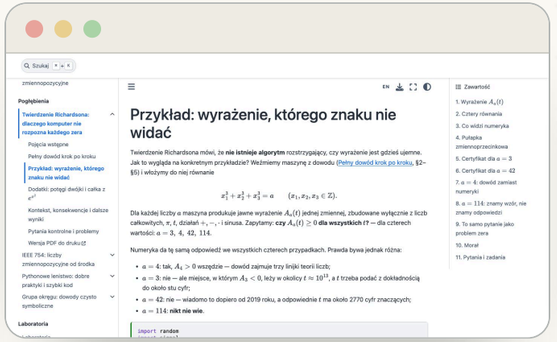
Kompletny kurs = *osiem warstw* z jednego źródła



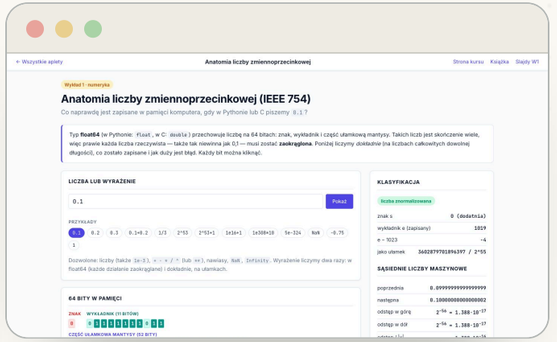
1 **Strona startowa** z grafiką liczoną programem



2 **Slajdy** HTML z notatkami i czasami



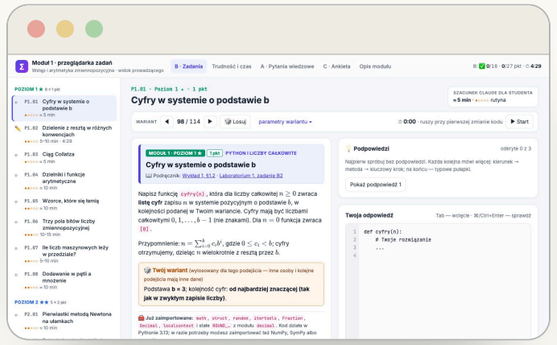
3 **Książka**: wzory, kod wykonywany przy budowie



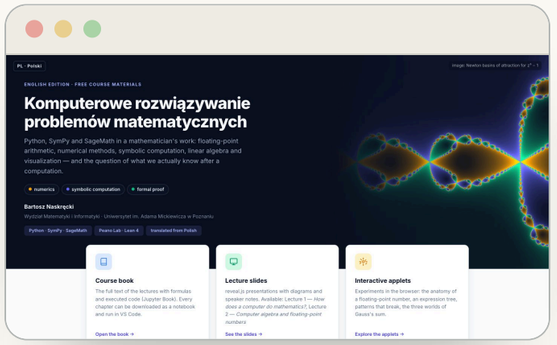
4 **Aplety** interaktywne w przeglądarce



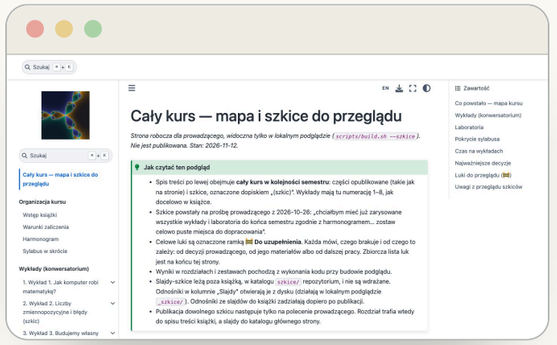
5 **Moodle**: moduły, zadania z kodem, ankiety



6 **Przegląd zadań** dla prowadzącego



7 **Wydanie angielskie** śledzące polskie



8 **Plan i szkice** całego semestru

Kurs „Komputerowe rozwiązywanie problemów matematycznych” (WMI UAM, 2026/27): bnaskrecki.faculty.wmi.amu.edu.pl/math-comp · Moodle i szkice — widok prowadzącego (niepubliczne).

Płatny model + *bezpłatne narzędzia* na własnym komputerze

PŁATNE · SUBSKRYPCJA MODELU

Claude Code
agent w terminalu lub VS Code: czyta pliki, uruchamia programy, pisze i testuje kod

Claude in Chrome
agent w przeglądarce — np. budowa zadań w zalogowanym Moodle

BEZPŁATNE · LOKALNIE

VS Code
edytor, podgląd plików

Python + uv
SymPy, NumPy, SageMath

Jupyter Book
książka ze wzorami i kodem

reveal.js + MathJax
slajdy HTML, PDF

Obsidian
baza wiedzy w Markdown

git + GitHub
historia, kopia, prywatne repo

UCZELNIANE

Moodle + CodeRunner
zadania z kodem sprawdzane automatycznie

Serwer WWW wydziału
statyczne strony, rsync

USOS, sylabus
jedyne źródło zasad

```
~/claude/math-comp — 26.10.2026
> ! scripts/deploy.sh
sprawdzanie wydania angielskiego: błędy 0,
uwagi 3
sprawdzono 4085 odnośników,
brakujących: 0, zagnieżdżonych <a>: 0
Transfer starting: 710 files
```

WSKAZÓWKA

Wybieraj **formaty tekstowe i otwarte** — Markdown, HTML, Python, LaTeX. Agent je czyta, pisze, porównuje i sprawdza; zamknięty edytor slajdów zamyka mu drogę.

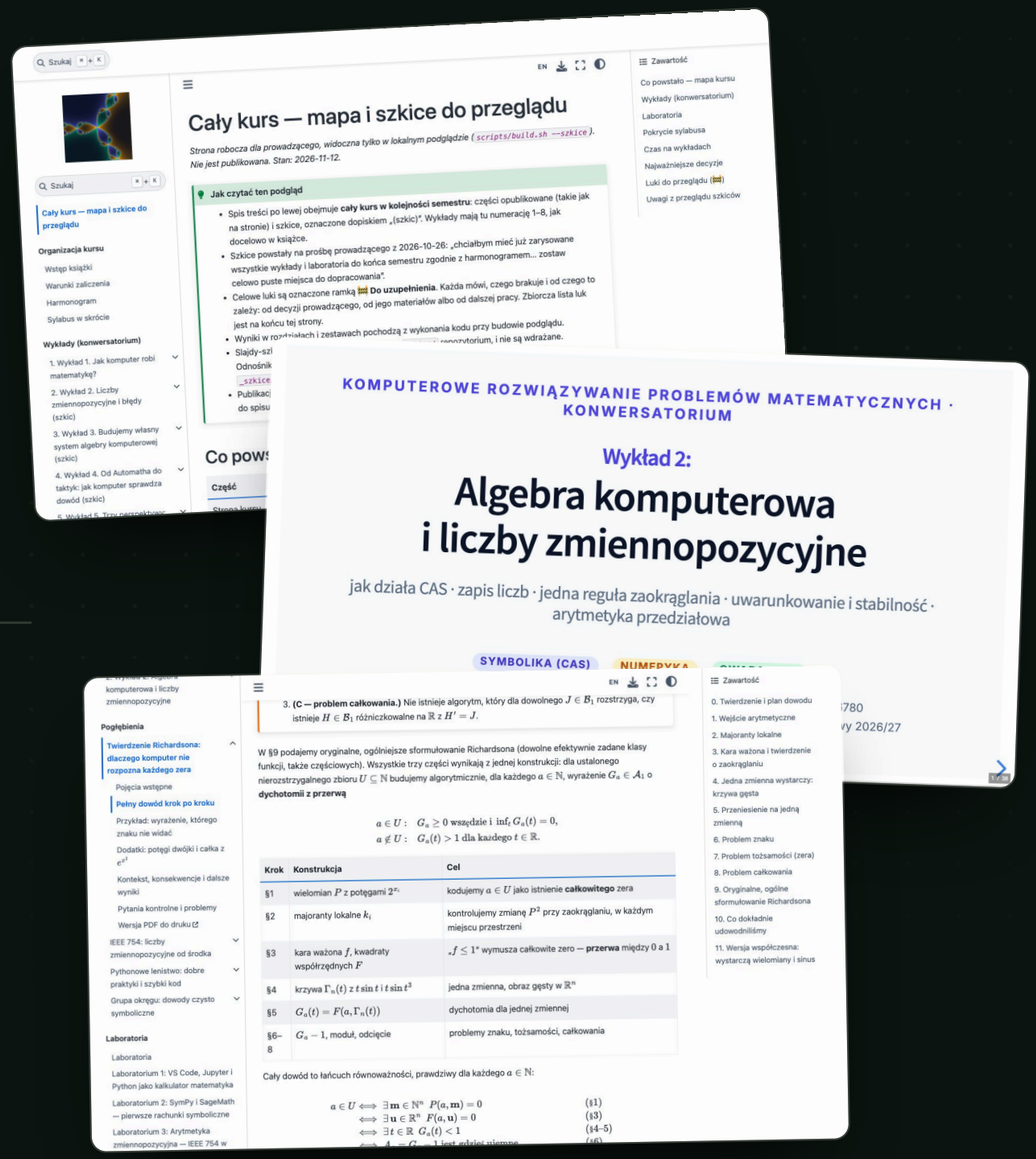
Terminal: rzeczywisty przebieg publikacji kursu math-comp (wynik skryptu z dziennika sesji, 26.10.2026).

I

CZĘŚĆ I WARSZTAT

Od pustego folderu do *planu kursu*

Prompt startowy, pamięć projektu, rodzaje poleceń i etapy pracy.



Pierwsze polecenie: *kontekst, produkty, granice*

PROWADZĄCY → CLAUDE CODE · MATH-COMP

Projekt jest portalem kursu Komputerowe rozwiązywanie problemów matematycznych, który prowadzę dla studentów drugiego roku kierunku matematyka [...]. Załączam szczegółowy dokument sylabusa [...] 8 spotkań (po 90 minut) na konwersatorium [...] ćwiczenia (15 spotkań po 90 minut). ①
Istotne jest dla mnie przygotowanie landing page (Grafikę do landing page wygeneruj), slajdów na konwersatorium oraz zestawów ćwiczeń [...] oraz stworzenie apletów. ②
Będziemy się wzorować na szablonie do wykładu github.com/nasqret/classical-foundations-ann ③ [...] Chcę do materiałów posiłkować się zawartością mojego kursu [...] ④
Istotnym elementem będą zadania interaktywne w Moodle wymagające kodowania i sprawdzania tego kodu. ⑤
Chciałbym, aby w pierwszym wykładzie pojawiły się informacje o tym, jak taki Computer Algebra System właściwie działa [...] i czym się to różni od pełnego dowodu formalnego, np. kompilowanego w Lean [...] ⑥
Na razie udostępnisz tylko pierwszy wykład. Kolejnych na razie nie generuj. Muszę pomyśleć. ⑦
The documentation should be: memory file · journal file · plan file · Obsidian vault · Jupyter Book · local git repository · private GitHub repository ⑧

1. **Kontekst:** kto, dla kogo, sylabus, terminy z USOS
2. **Produkty:** strona, slajdy, laboratoria, aplety
3. **Wzorzec:** wcześniejszy kurs jako szablon stylu i struktury
4. **Źródła:** własne materiały z poprzednich lat
5. **Ocenianie:** zadania z kodem w Moodle
6. **Ambicja merytoryczna** pierwszego wykładu
7. **Granica:** tylko wykład 1 — „Muszę pomyśleć”
8. **Dokumentacja:** siedem artefaktów — ta sama lista co przy tym wykładzie

Cytat z dziennika sesji Claude Code (skróty [...]; uzupełnione polskie znaki). Pełny tekst: książka, rozdz. „Biblioteka promptów”.

Agent zaczyna każdą sesję *od lektury* planu i dziennika

📘 **CLAUDE.md**
instrukcja: zasady twarde, układ, komendy, pułapki

🧠 **MEMORY.md**
fakty i decyzje trwałe: terminy, zasady, adresy

📓 **JOURNAL.md**
dziennik z datami — gdzie skończyliśmy (1258 wierszy)

🗺️ **PLAN.md**
statusy i **otwarte decyzje prowadzącego**

🕸️ **vault/**
Obsidian: pojęcia, decyzje, infrastruktura (81 notatek)

📖 **book/**
Jupyter Book: pełny tekst z wzorami i kodem

🌿 **git + prywatne GitHub**
każda zmiana opisana i odwracalna — 60 commitów w sześć tygodni

PROWADZĄCY · 25.10.2026

Zrób 1 i 2. Chcę zachować bliskość twierdzenia Richardsona. Nie robimy didaskaliów.

PAMIĘĆ AGENTA · NOTATKA „FEEDBACK”

Richardson: stay on the theorem — bez dygresji i przykładów spoza konstrukcji dowodu. How to apply: nowe pomysły tylko jako krótka propozycja, nigdy wpisane do tekstu bez prośby.

EFEKT

Raz wypowiedziana decyzja **obowiązuje w kolejnych tygodniach** — także po wyczerpaniu kontekstu i w nowej sesji.

Pliki projektu `math-comp` (stan 24.11.2026); notatka pamięci `richardson-stay-on-theorem.md` (pamięć Claude Code projektu).

60 wypowiedzi w sześć tygodni: *sześć rodzajów* poleceń

1 · STARTOWY

> …Na razie udostępnisz tylko pierwszy wykład. Kolejnych na razie nie generuj. Muszę pomyśleć.

raz: kontekst, produkty, granice

2 · TREŚCIOWY

> …standard IEEE 754 i szczegóły jego implementacji, triki, subtelności. To ma być perełka tematyczna, genialne przykłady (relatywnie proste).

cel dydaktyczny i poziom, nie technika

3 · ROZMOWA MERYTORYCZNA

> Czy ten przykład nie jest nieco wydumany? Jeśli nie znamy trójki dla n = 114, to właściwie nie znamy tej funkcji, którą ona koduje, prawda?

krytyka i dopytywanie

4 · RUNDA DECYZJI

> – Nazwy w kodzie: zostają polskie czy tłumaczymy?
> **>>> Nie tłumacz kodów ani nazw.**

agent pyta listą, prowadzący odpowiada „>>>"

5 · STERUJĄCY

> Wypchnij i wdróż.
> Slajdy 20 do końca przenieś do wykładu 2.

krótko; publikacja tylko na polecenie

6 · ZGŁOSZENIE Z SALI

> W pytaniu 11 z modułu 1 jeden ze studentów dostał błąd w zadaniu: `TypeError…` Czy błąd jest po stronie implementacji czy zadania?

pętla zwrotna od studentów

Co piąta z 60 wypowiedzi (**13**) ma najwyżej trzy słowa: „Tak.", „Kontynuuj.", „Wypchnij." — resztę pracy agent wykonuje sam, między pytaniami.

Dziennik sesji kursu math-comp, 5.10–19.11.2026 (uzupełnione polskie znaki; pełna lista: książka, „Biblioteka promptów").

Od koncepcji do utrzymania: *siedem etapów*

1 **Koncepcja** — tytuł, abstrakt, sylabus, terminy

→ 2 **Szkielet** — portal, wzorzec, dokumentacja

→ 3 **Wykład po wykładzie** — publikacja tylko na polecenie

→ 4 **Szkic całości** — cały semestr z celowymi lukami 🚧

→ 5 **Pogłębienia** — perełki tematyczne 🐏

→ 6 **Moodle** — moduły, warianty, testy

→ 7 **Utrzymanie** — poprawki, wersja EN, redakcja



5.10 — portal, wykład 1 (52 slajdy), 5 apletów, laboratoria 1–2

6.10 — pełny dowód tw. Richardsona: rozdział, PDF, aplet

25.10 — Moodle moduł 1, przeglądarka zadań, **wydanie angielskie**

26.10 — szkice wykładów 2–8 i laboratoriów 4–9

31.10 — „Pythonowe lenistwo”: 9 części, czasy mierzone

12.11 — grupa okręgu: dowody symboliczne + aplet

18.11 — wykład 2 na 90 min, Moodle moduł 2

19.11 — rozdział wykładu 2 w książce, PL i EN

Tempo wyznacza **prowadzący**: szkic całego semestru jest gotowy po trzech tygodniach, ale na stronę trafia tylko to, co zostało omówione i zatwierdzone.

Daty z historii git repozytorium `nasqret/math-comp` (60 commitów, 5.10–19.11.2026).

II

CZĘŚĆ II TREŚĆ

Ambitna ścieżka i *wyrafinowane przykłady*

Na to idzie czas, który agent oddaje prowadzącemu: plan kursu, perełki tematyczne, auto-research w rozmowie.



Cel: doprowadzić drugi rok *do frontu badań* — małymi krokami

PROWADZĄCY · 26.10.2026, 00:05

Oryginalnym pomysłem tego przedmiotu jest to, aby pokazać studentom, jak można dotrenować lub wytrenować modele językowe, aby rozwiązywały problemy matematyczne. […] Wiem, że to dość ambitny plan, ale pomysł jest taki, aby studentów szybko doprowadzić do frontu badań naukowych w tej tematyce, a jednocześnie pokazać im na miarę ich możliwości małe projekty.



W1 · opublikowany — Jak komputer robi matematykę?

W2 · opublikowany — Algebra komputerowa i liczby zmiennopozycyjne

W3 · szkic — MiniCAS od środka

W4 · szkic — Od Automatha do taktyk

W5 · szkic — Metody numeryczne: obraz → CAS → dowód

W6 · szkic — CAS w dowodach: bazy Gröbnera, geometria

W7 · szkic — Geometria w stylu AlphaGeometry

W8 · szkic — Modele językowe zgadują taktyki

Agent pisze też **narzędzia dydaktyczne**, których prowadzący nie musiałby umieć programować: sprawdzarkę typów λΠ, mały silnik geometrii DD+AR (16 testów), predyktory taktyk dla dowodów z Peano Lab.

Prompt z 26.10.2026 (fragment); plan wykładów W3–W8 to propozycja z `PLAN.md` kursu — szkice niepublikowane (6 talii po 39–46 slajdów, celowe luki 🚧).

Eksploracja źródeł: agent proponuje, *prowadzący* zawęża

6.10 **PROWADZĄCY**

Mam w folderze src notatki oraz oryginalny artykuł Richardsona. Chciałbym przygotować solidnie dowód krok po kroku ze wszystkimi szczegółami. [...]

AGENT

rozdział z pełnym dowodem (pojęcia, dowód, kontekst, zadania) + wersja LaTeX/PDF z tych samych źródeł + aplet

22.10 **PROWADZĄCY**

Czy możemy zbudować przykład wyrażenia, o którego zerowaniu lub ujemności będzie bardzo trudno rozstrzygnąć? [...] numeryka może być bardzo zwodnicza.

AGENT

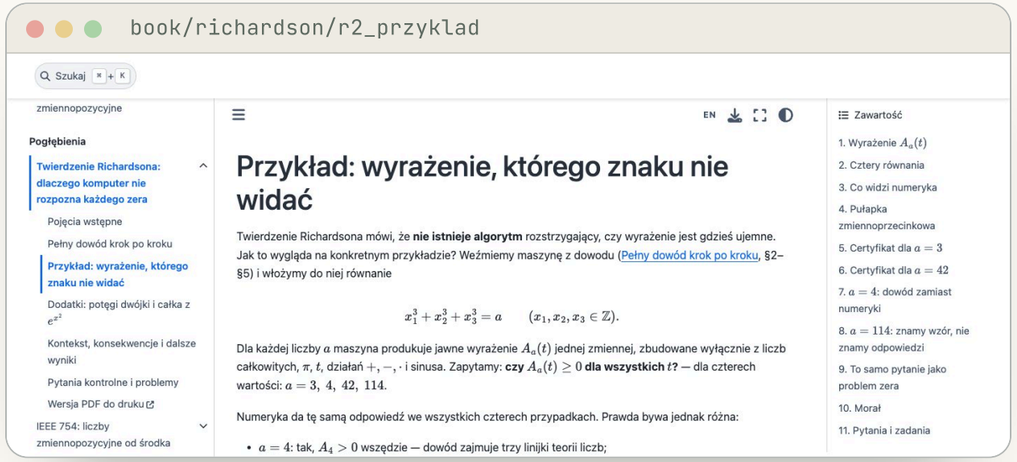
„wyrażenie, którego znaku nie widać" — z równania $x^3 + y^3 + z^3 = a$; certyfikat dla $a = 42$ liczony przy budowie książki

25.10 **PROWADZĄCY**

Czy ten przykład nie jest nieco wydumany? [...] **Nie robimy didaskaliów.**

TWIERDZENIE (D. RICHARDSON, 1968)

Dla wyrażeń zbudowanych z x, liczb wymiernych, π, $\ln 2$ oraz funkcji $\sin$, $\exp$, $|\cdot|$ za pomocą dodawania, mnożenia i składania **nie istnieje algorytm**, który rozstrzyga, czy dane wyrażenie jest tożsamościowo równe 0.



Prompty z dziennika sesji (fragmenty); rozdział „Twierdzenie Richardsona" i jego wersja PDF: bnaskrecki.faculty.wmi.amu.edu.pl/math-comp/book/richardson/.
Twierdzenie: J. Symbolic Logic 33 (1968).

Jeden prosty obiekt, *pięć części* matematyki

PROWADZĄCY · 12.11.2026, 21:08

Istota jest [...] pokazanie studentom, jak elegancko sprawdzać za pomocą metod czysto symbolicznych dowody matematyczne. [...] Fajnie będzie wycisnąć jak najwięcej z tego prostego przykładu fajnej grupy algebraicznej.

PROWADZĄCY · 21:48

Siła matematyki jest zauważanie, że takie zorganizowane abstrakcje w naturalny sposób skracają i konceptualizują dowody, a także wskazują całkowicie nowe fenomeny [...]

OKRĄG $a^2+b^2=1$ JAKO GRUPA

$$(a,b)\cdot(c,d)=(ac-bd,\ ad+bc)$$

POTĘGI PUNKTU → CZEBYSZEW

$$(\cos\theta,\sin\theta)^n=\big(T_n(\cos\theta),\ \sin\theta\,U_{n-1}(\cos\theta)\big)$$

$$T_m\circ T_n=T_{mn}$$

GRUPA LIEGO, FORMALNIE

$$\varepsilon^2=0\ \Rightarrow\ T_n'=n\,U_{n-1}$$



Pięć części: aksjomaty grupy jako certyfikaty wielomianowe · Czebyszew z potęg punktu · siedemnastokąt Gaussa · okrąg nad $\mathbb{F}_p$ i test Lucasa–Lehmera · grupa Liego i liczby dualne. Od prośby (21:08) do publikacji PL i EN (22:03) i apletu (22:12): **nieco ponad godzina**.

Prompty z 12.11.2026 (fragmenty); pogłębienie „Grupa okręgu: dowody czysto symboliczne”: math-comp/book/okrag/ i aplet „Grupa okręgu” (commity 12.11.2026).

Każda liczba pokazana studentom jest *obliczona*, nie zapamiętana

> *Every number or output shown to students must be computed, not recalled.*
>
> CLAUDE.MD KURSU MATH-COMP · ZASADA TWARDA NR 3

```
# Python 3.12: pułapki zapisane w instrukcji agenta
>>> sum([0.1] * 10) == 1.0
True      # sum() kompensuje błędy zaokrągleń
>>> s = 0.0
>>> for _ in range(10): s += 0.1
>>> s
0.9999999999999999
>>> int(math.exp(math.pi * math.sqrt(163)))
262537412640768256    # prawdziwa: …743,99…
```

MECHANIZM

Notatniki książki **wykonują się przy każdej budowie**. Zdanie o czasie działania ma ukrytego „strażnika”: jeśli świeży pomiar mu przeczy, budowa się zatrzymuje.

```
upewnij(t_timeit < 0.8 * czasy_zakresu[...],
        "%%timeit mierzy pętlę jak w funkcji,
         a nie jak w komórce")
```



Wyniki powyżej sprawdzone 24.11.2026 (Python 3.12.11); $e^{\pi\sqrt{163}} = 262\,537\,412\,640\,768\,743{,}99999999999925\ldots$, odstęp liczb maszynowych w tym miejscu = 32.
Strażnik: pogłębienie „Pythonowe lenistwo”, cz. 1.

Za dużo materiału? *Pogłębienia* zamiast cięć

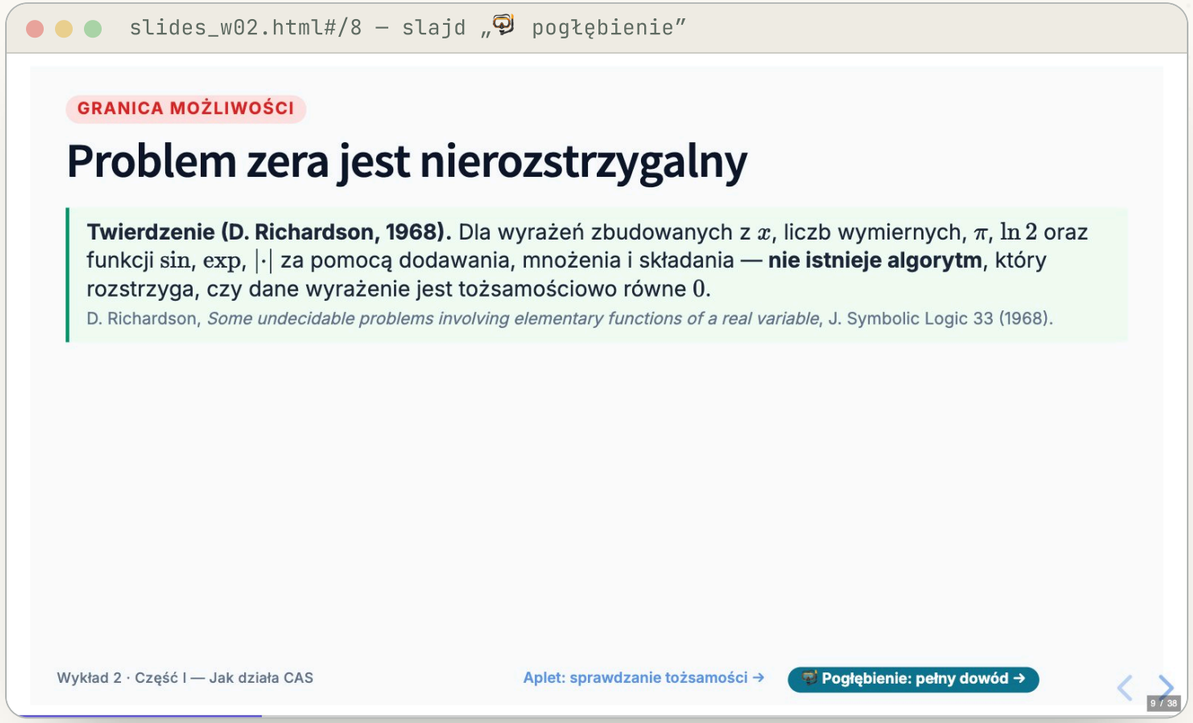
PROWADZĄCY · 18.11.2026, 02:31

Część tematów zaawansowanych z numeryki możemy przenieść do pogłębień tematycznych. Chciałbym jutro omówić podstawy CAS i numeryki, jak planowałem, ale odchudź liczbę przykładów, bo mam tego za dużo. Usuń robocze znaki, ale oznacz część slajdów ze znaczkiem pogłębienia (wymyśl fajną ikonkę). Te tematy będą tylko zaanonsowane i przekierowane do pogłębienia. Zmniejsz liczbę slajdów obowiązkowych do takiej, abym spokojnie zmieścił się w 90 minutach.

38
slajdów w talii

≈ 73
min obowiązkowych (31 slajdów) z 90

7 🤿
slajdów-zapowiedzi z linkiem do książki



Czasy części na slajdzie planu to **sumy czasów z notatek prelegenta** — liczone, nie szacowane.

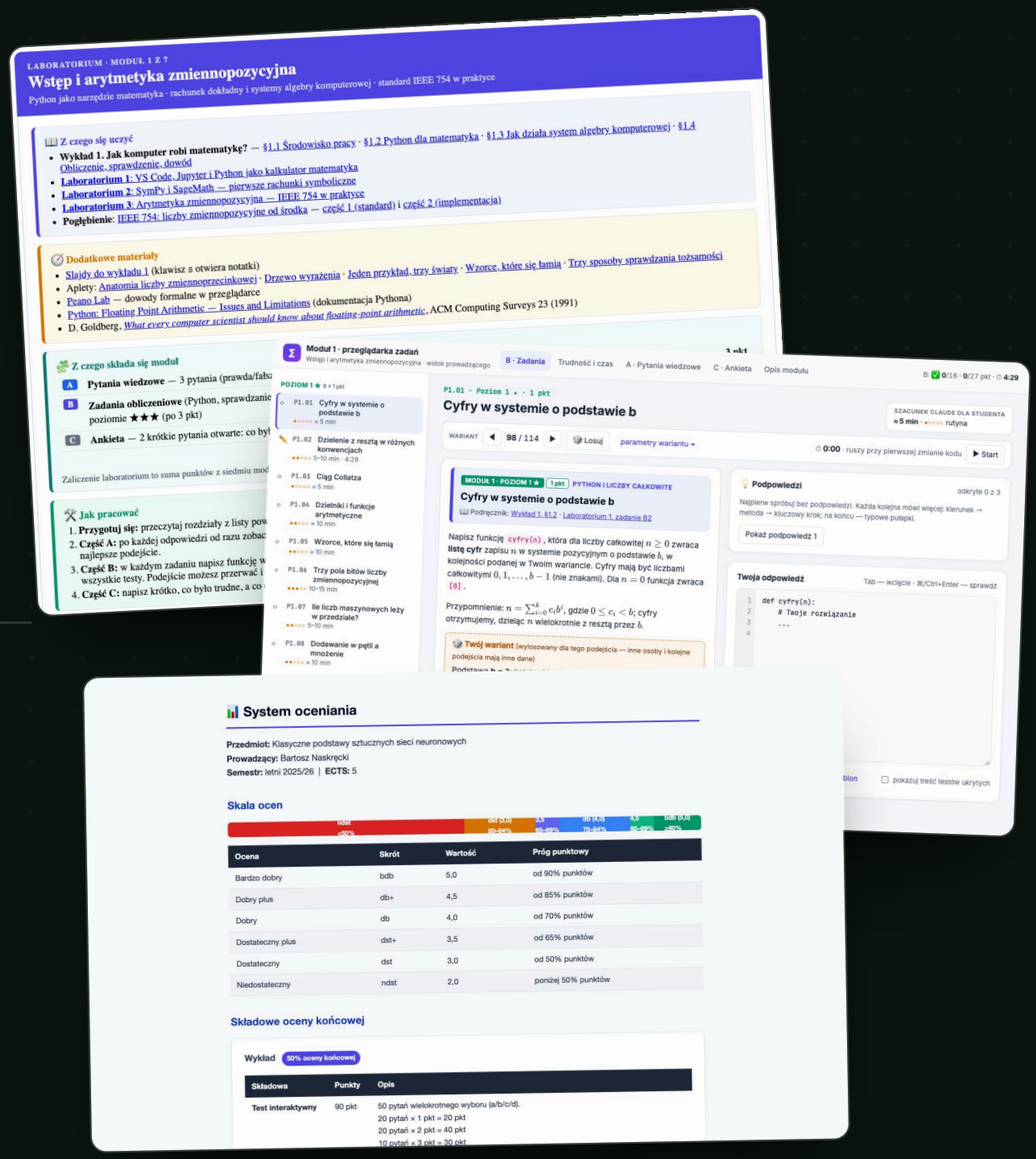
Prompt z dziennika sesji (uzupełnione polskie znaki); talia `slides_w02.html`: math-comp/slides_w02.html.

III

CZĘŚĆ III MOODLE I PRZEPŁYW INFORMACJI

Zadania, które *sprawdzają się same*

Od specyfikacji oceniania do modułów w Moodle, przeglądu zadań i poprawek po zgłoszeniach studentów.



Prowadzący pisze zasady, agent buduje moduły

PROWADZĄCY → CLAUDE CODE

Zaliczenie laboratoriów będzie składać się z sumy punktów uzyskanych za rozwiązanie zadań z 7 modułów. […]
1. Pytania wiedzowe: TAK/NIE, jednokrotny wybór (ok. 15% zadań) […]
2. Pytania obliczeniowe w CodeRunner (ok. 80% zadań) […] na 3 poziomach (z rosnącą punktacją). […]
3. Pytania otwarte (krótkie odpowiedzi) jako element ankietowania zajęć (ok. 5%). […]
Studenci mogą rozwiązywać zadania do skutku. […] Zadania obliczeniowe są zindywidualizowane (liczbowo i pod względem drobnych różnic technicznych) do każdego studenta. […]
Chciałbym, abyś przygotował te zadania, sterując moim komputerem bezpośrednio w moim zalogowanym serwisie Moodle, budując wszystkie zadania i testując je. […] Warto dla urozmaicenia dodać estetyczny system kolorowych anotacji, ramek itd.

97
pytań wiedzowych w puli modułu (losowane 3)

16
zadań CodeRunner: 8 × ★, 5 × ★★, 3 × ★★★

1662
warianty danych w module 2 — nowe przy każdym podejściu

Moodle · opis modułu 1 (HTML z generatora)

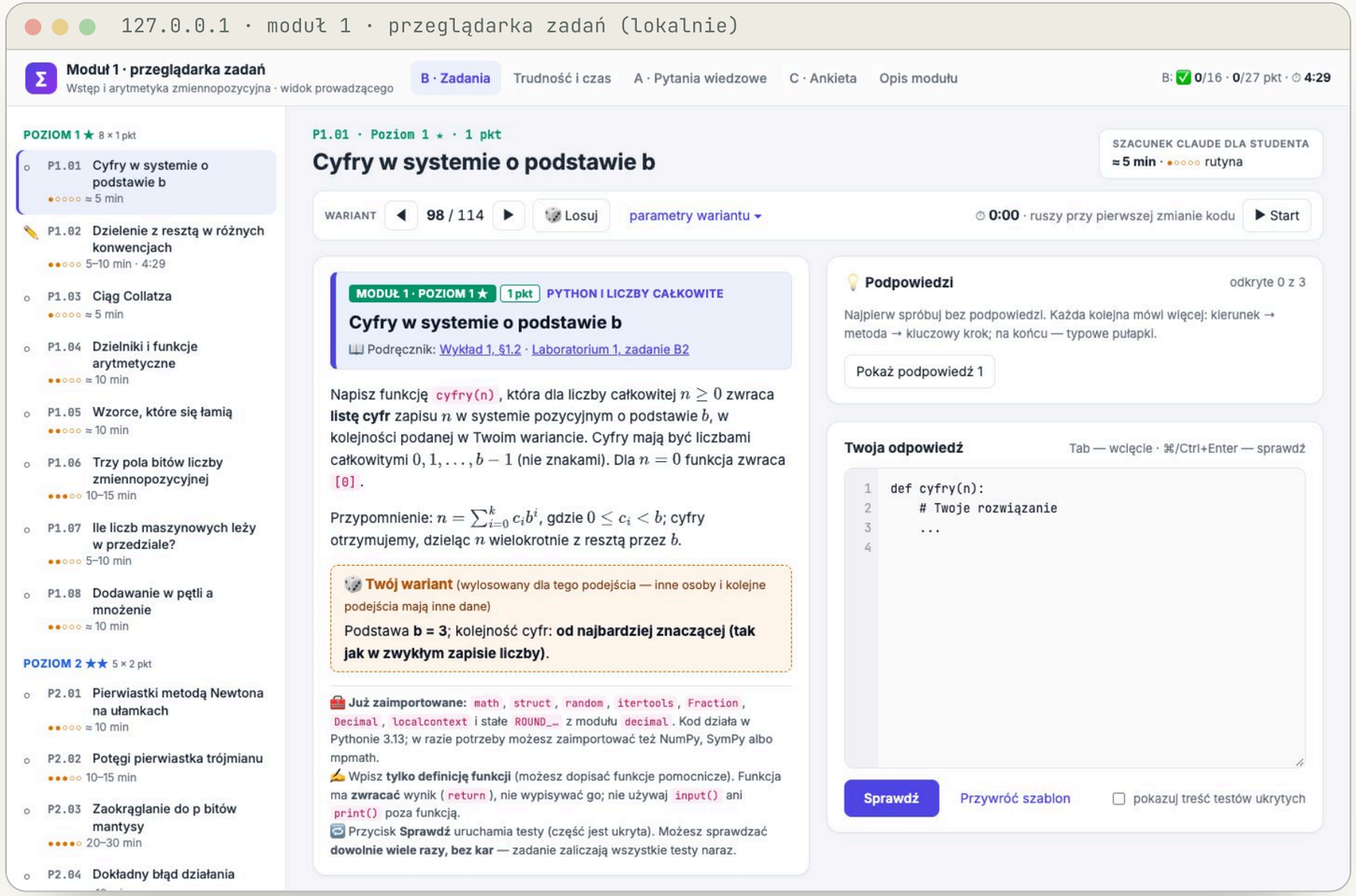


AGENT

≈ 10,4 tys. wierszy generatora w Pythonie: warianty, testy jawne i ukryte, **mutanty** (typowe błędy studentów muszą zostać odrzucone), budowa w Moodle przez rozszerzenie Chrome.

Prompt z 25.10.2026 (fragmenty, uzupełnione polskie znaki); liczby z plików `moodle/modul01`, `moodle/modul02` repozytorium kursu (24.11.2026). Moduł 1 udostępniony 25.10, moduł 2 zbudowany 18.11.

Prowadzący *rozwiązuje zadania*, zanim zrobią to studenci



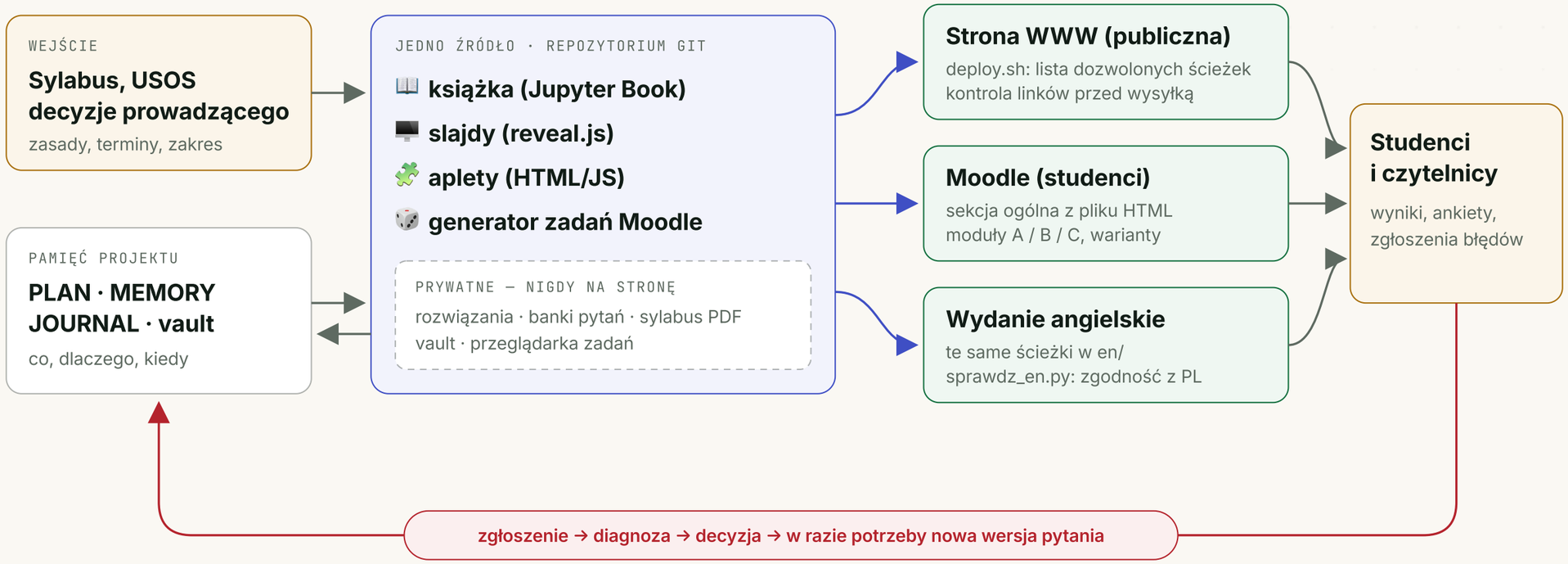
Narzędzie `moodle/przegladarka.py` (lokalny serwer; prywatne — nie trafia na stronę kursu). Zrzut: 24.11.2026.

PROWADZĄCY · 25.10.2026

Czy mogę w jakiś prosty sposób podejrzeć wszystkie zadania bez ingerowania w Moodle'a? [...] Chciałbym, abyś przygotował mi wersję z podpowiedziami na zadanie, ale tak, abym mógł sam zaproponować implementację. Chcę wiedzieć, jak trudne są te zadania.

- te same testy co w Moodle
- każdy wariant: przewijanie i losowanie
- podpowiedzi stopniowane: kierunek → metoda → kluczowy krok → pułapki
- szacunek czasu i trudności, licznik czasu prowadzącego

Jedno źródło, wiele kanałów, *pętla zwrotna* od studentów



12.11: `TypeError` u studenta w zadaniu 11 modułu 1 → „Czy błąd jest po stronie implementacji czy zadania?" → diagnoza: błąd w kodzie studenta (`2**(p-1-e)` daje `float`), zadanie poprawne; podpowiedź — decyzja prowadzącego.

Kurs math-comp: `deploy.sh`, `moodle/kurs/sekcja_ogolna.py`, `sprawdz_en.py`. Zgłoszenie z 12.11.2026 bez danych studenta; bez zmian w Moodle (dziennik kursu).

Skill: zapisany warsztat, który *agent wywołuje sam*

```
.claude/skills/tlumacz-wyklad/SKILL.md
---
name: tlumacz-wyklad
description: Tłumaczy slajdy i stronę
  wykładu PL → EN: struktura, liczby,
  terminologia, niuanse języka…
---
# Tłumaczenie wykładu
1. Polski jest źródłem; EN w site/en/
2. Kodu, nazw plików, poleceń — nie tłumacz
3. Cytaty promptów: wiernie + „(transl.)”
4. Słowniczek terminów, uwagi tłumacza
5. Kontrola: scripts/sprawdz_en.py
```

Skill to **katalog z instrukcją** (`SKILL.md`) i materiałami. Agent ładuje go, gdy zadanie pasuje do opisu, albo na polecenie `/nazwa`.

seminar-deep-study WŁASNY · PUBLICZNY

referat jako głębokie studium jednej pracy: wybór tematu, digest, rachunki w CAS, Lean, notatki, slajdy, vault

labyrinth-exploration WŁASNY · PUBLICZNY

otwarte problemy badawcze jako labirynt: mapa tego, co udowodnione, obalone i nieznane

readback WŁASNY

dwóch agentów odczytuje sformalizowany dowód z powrotem na matematykę

agent-mailbox WŁASNY

skrzynka w plikach: dwóch agentów wymienia zadania i werdykty

chrome-browser

agent w przeglądarce — tak powstały moduły w zalogowanym Moodle

tlumacz-patent

ten sam mechanizm w innym zawodzie: pakiet skilli dla rzeczników patentowych

tlumacz-wyklad NOWY · TEN WYKŁAD

wersja angielska tych slajdów i strony powstała według tego skilla — na wzór skilla patentowego i zasad tłumaczenia kursu math-comp (słownik 66 haseł)

Skille w `~/.claude/skills/`; publiczne: github.com/nasqret/seminar-deep-study, github.com/nasqret/labyrinth-exploration.

Osiem zasad *na pierwszy kurs* z agentem

1 **Zacznij od folderu, nie od czatu**
projekt to katalog plików; agent pracuje na nich, git pamięta historię

2 **Napisz porządny prompt startowy**
kontekst, produkty, wzorzec, źródła, granice, dokumentacja

3 **Wymagaj pamięci projektu**
plan, dziennik, decyzje — każda sesja zaczyna się od lektury

4 **Mów o celu, nie o technice**
„perełka tematyczna, proste przykłady”; formaty i kod to zadanie agenta

5 **Decyzje zostaw sobie**
zasady oceniania, terminy, zakres publikacji — odpowiadaj na listy „do decyzji”

6 **Żądaj dowodu sprawdzenia**
liczby liczone przy budowie, zadania z testami, kontrola linków i układu

7 **Oddziel prywatne od publicznego**
rozwiązania i banki pytań nigdy na stronę; publikacja tylko na polecenie

8 **Celuj wysoko**
czas odzyskany na technice wydaj na ścieżkę kursu i głębię przykładów

Dziękuję

Prowadzący **projektuje**, agent **buduje**, **weryfikacja** rozstrzyga.

TEN WYKŁAD (PL · EN)
slajdy, PDF, pełny tekst (Jupyter Book), biblioteka promptów

SIECI NEURONOWE
bnaskrecki.faculty.wmi.amu.edu.pl/nnets

VIASM 2026
bnaskrecki.faculty.wmi.amu.edu.pl/vietnam2026

KURS NA ŻYWO
bnaskrecki.faculty.wmi.amu.edu.pl/math-comp

KRYPTOANALIZA
bnaskrecki.faculty.wmi.amu.edu.pl/crypto

SKILL DO REFERATÓW
github.com/nasqret/seminar-deep-study

bnaskrecki.faculty.wmi.amu.edu.pl/**uken2026**

Uniwersytet
im. Adama Mickiewicza
w Poznaniu