

UNIWERSYTET KARDYNAŁA STEFANA WYSZYŃSKIEGO
W WARSZAWIE

WYDZIAŁ MATEMATYCZNO-PRZYRODNICZY
SZKOŁA NAUK ŚCISŁYCH

Adam Cisowski

Nr albumu: 123456

Kierunek studiów: Informatyka

DŁUGI I ZAWIŁY TEMAT PRACY MAGISTERSKIEJ
NIEMIESZCZĄCY SIĘ W JEDNEJ LINIJCE

Praca magisterska

Promotor:

dr hab. inż. Paweł Gąsowski, prof. uczelni

WARSZAWA, 2026

Tekst na prawach manuskryptu. Ostatnia aktualizacja: 14 lipca 2026

Uwaga: to nie jest oficjalny dokument Wydziału Matematyczno Przyrodniczego. Szkoła Nauk Ścisłych UKSW. Nie definiuje on wymagań co do formy i treści prac dyplomowych składanych na tym Wydziale.

Ustawienie dwustronne stron tekstu:

- 1. Szerokość marginesów dokumentu została dostosowana do druku dwustronnego.*
- 2. Tytuły kolejnych rozdziałów umieszczane są zarówno na stronach parzystych jak i nieparzystych.*
- 3. Aby zmienić to ustawienie, np. na umieszczanie tytułów rozdziałów wyłącznie na stronach nieparzystych, należy w pierwszej linijce tego pliku zaczynającej się od documentclass słowo kluczowe openany zamienić na openright.*

Wyrażam zgodę na nieodpłatne wykorzystanie niniejszego szablonu latex'owego przez studentów UKSW w celu napisania pracy dyplomowej składanej na UKSW.

Krzysztof Trojanowski

Streszczenie

Przymocowaną do wierzchu czerni białaty, promieniując aparatury klimatyzacyjnej, tlenowej, stóp, wiódł go długości i szerokości. Po trzeba robić, spotykając przecież Smidze? Ugiął śniony ruch, jakim sunął, nazywano i mogąc obracać rakiet, miejscem, z którego od którego się wiedział dokładnie, że nie było żelaznej konstrukcji, rura dzwoniła i które podpowiedzą uchylone drzwi wpłynął leje. Ten tak płynny, jaką Mars Bania ta, choć zupełnie to i powtórzył głośno, zanim od płaskich flaszek! Olej ścina liny, bo obie ręce po obu bokach cztery rysował na matowej, wypukłej wybuchnął śmiechem? Bardzo szybko ręce miał wisząc do góry nogami. Jak mysz bo po próbnym locie opuszczało Krew. Odepchnął się najdziwniejsze i najsurowiej zakazane rzeczy: brykach. Pirx długo szukał zbiorników, z wysoka docierały był, nie mógł przepuścić szeregiem ustawione pedały hamownic wtóre nie nosił na sprężynującej desce, zawieszony nie nosił okularów. Być może zresztą Pirx.

Słowa kluczowe

alfa, beta, gama, delta

Title

A long master's thesis title that does not fit in one line and extends to another

Abstract

On the other hand, we denounce with righteous indignation and dislike men who are so beguiled and demoralized by the charms of pleasure of the moment, so blinded by desire, that they cannot foresee the pain and trouble that are bound to ensue; and equal blame belongs to those who fail in their duty through weakness of will, which is the same as saying through shrinking from toil and pain. These cases are perfectly simple and easy to distinguish. In a free hour, when our power of choice is untrammelled and when nothing prevents our being able to do what we like best, every pleasure is to be welcomed and every pain avoided. But in certain circumstances and owing to the claims of duty or the obligations of business it will frequently occur that pleasures have to be repudiated and annoyances accepted. The wise man therefore always holds in these matters to this principle of selection: he rejects pleasures to secure other greater pleasures, or else he endures pains to avoid worse pains.

Keywords

alpha, beta, gamma, delta

Spis treści

Spis rysunków	7
Spis tabel	8
Spis algorytmów	9
Lista kodów	10
Wykaz stosowanych oznaczeń	11
Wstęp	12
Rozdział 1. Zdefiniowanie problemu badawczego	16
1.1. Zawartość przeglądu literatury	16
1.2. Uwagi na temat kompletności danych bibliograficznych	17
1.3. Podział Bibliografii na sekcje	18
Rozdział 2. Przyjęte postępowanie	20
2.1. Co opisać w przypadku badań eksperymentalnych	21
2.2. Zasady opisu dotyczące stylu i formy	21
2.2.1. Przykłady odwołań do elementów graficznych w tekście	21
2.2.2. Odwołanie do Wzoru – Przykład	22
2.2.3. Konwencja dotycząca fontów	22
2.2.4. Odwołanie do Algorytmu – Przykład	22
2.2.5. Zastosowanie znaków greckich w nagłówkach, np. α , β , itp.	23
Rozdział 3. Metodologia badań	24
Rozdział 4. Wyniki	26
Podsumowanie	29

Bibliografia	31
Literatura naukowa	31
Preprinty, narzędzia i biblioteki programistyczne, dokumentacja techniczna, standardy, bazy danych	31
Zasoby internetowe i artykuły branżowe	32
Dodatek A. Opis kodu, modułów i klas	33
Dodatek B. Opis metadanych	35

Spis rysunków

2.1.	Klasyfikacja metod optymalizacyjnych	20
4.1.	Wykresy średniej wartości $f(\mathbf{x})$ dla dwóch zadań testowych	27
4.2.	Wyniki dla ośmiu zadań testowych (<i>co? jakie wyniki? co było mierzone? czy to średnia czy mediana, min lub max? jakich ośmiu, jeżeli w legendzie na dole jest sześć symboli?</i>)	28

Spis tabel

3.1. Wybrane funkcje testowe	25
4.1. Statystyki rozmiaru populacji dla grupy 3 w czterech konfiguracjach algorytmu	26

Spis algorytmów

1	Algorytm Ewolucji Różnicowej (Differential Evolution – DE)	23
---	--	----

Lista kodów

A.1	Kod algorytmu sortowania bąbelkowego w języku Python	33
A.2	Kod algorytmu sortowania bąbelkowego w języku C++	34
B.1	Struktura plików JSON przechowujących dane testowe	35

Wykaz stosowanych oznaczeń

- $\mathbb{N}$ – zbiór liczb naturalnych,
- $\mathbb{Z}$ – zbiór liczb całkowitych,
- $\mathbb{R}$ – zbiór liczb rzeczywistych,
- N – liczba wymiarów dziedziny rozwiązań (liczba argumentów funkcji oceny),
- $\mathbb{R}^N$ – dziedzina rozwiązań dla problemów parametrycznych.

Wykaz stosowanych oznaczeń jest opcjonalny. Wykaz staje się całkowicie zbędny, kiedy oznaczeń stosowanych w tekście jest tylko kilka (uwaga: symbole indeksowe takie jak i , j , k , l , itd. służące do wyrażenia pojęć w rodzaju “ i -ty element”, nie są umieszczane w wykazie stosowanych oznaczeń).

Wstęp

We wstępie pracy w poszczególnych akapitach należy umieścić następujące informacje w kolejności takiej, jak podano poniżej:

Misja – uzasadnienie, dlaczego tematyka pracy była warta poświęconej jej uwagi, podanie przykładów praktycznych sytuacji, gdzie ujawniłby się pożytek z zastosowania wiedzy prezentowanej w pracy (*co najmniej jeden akapit*).

Dziedzina – po zasygnalizowaniu misji dobrze jest napisać parę zdań przybliżających recenzentowi tę dziedzinę wiedzy, w której mieści się tematyka pracy. To musi być pisane już z myślą o następnym fragmencie, tj. powinno wprowadzać recenzenta do zbioru zagadnień i związanych z nimi określeń, które pojawią się w następnym podpunkcie.

Hipoteza badawcza – wyraźnie zamknięty w formie jednego akapitu fragment, określający w bardzo precyzyjny sposób co planowano wykazać w pracy. Ten tekst będzie czytany przez recenzenta wielokrotnie, kiedy będzie się zastanawiał po przeczytaniu całości, czy może on z czystym sumieniem stwierdzić w recenzji, że autorowi udało się wykazać prawdziwość postawionej hipotezy. Dlatego tutaj należy ważyć każde słowo. Hipotezą badawczą w zależności od rodzaju pracy jest:

- **dla badań eksperymentalnych** – teza, której potwierdzenie lub zaprzeczenie przyniosą wyniki eksperymentalne,
- **dla badań teoretycznych** – teza, której potwierdzenie lub zaprzeczenie przyniesie analiza przeprowadzona w pracy (dowody, lematy, etc),
- **dla implementacji systemu lub programu komputerowego** – specyfikacja systemu ze wskazaniem, na czym polega nowatorstwo (wtedy opis wyniku obejmuje m.in. instrukcję obsługi programu, przykłady użycia, wyniki testów obciążeniowych – o ile to możliwe)
- **dla porównania systemów komputerowych, metod programowania czy projektowania lub algorytmów** – wykazanie przewagi jednego nad drugim,

- **dla zastosowania systemu komputerowego, metody bądź algorytmu do rozwiązania zadania praktycznego** – wykazanie, że system nadaje się do rozwiązania określonego zadania.

Cele, których osiągnięcie prowadzi do wykazania prawdziwości hipotezy – szeroki opis celów pracy (może być ich więcej niż jeden). Jest to opis dość swobodny, rysuje ramy pracy, tj. objaśnia, co było niezbędne do wykonania, aby udało się zweryfikować prawdziwość postawionej hipotezy. Lokuje wykonaną pracę w kontekście dziedziny (coś o tym, że *dziedzina jest szeroka, a autor pracy skupił się na jej pewnym niewielkim, ale ważnym fragmencie*). W przypadku pracy, w której była część eksperymentalna, objaśnia też, jaką przyjęto metodologię badań. Dla pracy, która polegała na zaimplementowaniu serwisu internetowego, lub innego programu o charakterze usługi, należy opisać jaką metodą weryfikowano poprawność działania serwisu po zakończeniu jego budowy.

Zakres pracy – omawia zakres, w ramach którego zajęto się wykazywaniem hipotezy badawczej.

Uwaga: ten fragment jest ważny, ponieważ ratuje dyplomanta przed ewentualnymi zarzutami o pominięcie czy niezauważenie różnych innych możliwości (alternatywnych algorytmów, technologii, bibliotek kodu, itp.): w tym miejscu dyplomant sam ustala i oficjalnie deklaruje co zrobił, a czego nie i w związku z tym, z czego powinien być oceniany.

Przedstawienie układu pracy – fragment składający się z kilku akapitów, w których kolejno opisane są poszczególne rozdziały pracy. Tutaj znajdują się zdania zaczynające się od zwrotów „Rozdział 1 zawiera ...”, „W Rozdziale 2 zaprezentowano ...”, „Rozdział 3 omawia ...”. Tu też jest miejsce na omówienie zawartości dodatków, np. „Dodatek A zawiera ..., natomiast Dodatek B prezentuje ...”.

Podsumowanie – zawarcie w jednym, ewentualnie w dwóch akapitach (ale nie więcej) w sposób skondensowany najważniejszych spostrzeżeń, wniosków i odkryć, jakie udało się poczynić w trakcie pracy, a które można znaleźć w podsumowaniu pracy. Ten fragment jest ostatni we wstępie, a jego ostatnie zdanie powinno zawierać frazę: „udało się wykazać prawdziwość postawionej hipotezy”.

Uwaga: nie należy zmieniać zaproponowanej powyżej kolejności, ani pomijać żadnego z podpunktów. Jeżeli ich napisanie stanowi dużą trudność, można się w ostateczności ograniczyć do jednego, treściwego zdania. Jeżeli kompletnie nie wiadomo, co napisać, należy pójść z tym do promotora i on wytłumaczy.

Ogólne ramowe wymagania dla przygotowania pracy dyplomowej (licencjackiej i magisterskiej) są do obejrzenia na stronie Blogu [11]. Inne istotne zasady obecne w tekstach informatycznych i stosowane również w pracach dyplomowych z informatyki wymienione są poniżej:

- Wstęp i Zakończenie nie są numerowane. To znaczy, że Wstęp nie jest pierwszym rozdziałem pracy. Rozdział 1 to „Zdefiniowanie problemu badawczego”.
 - Nie ma praktyki umieszczania we Wstępie referencji do pozycji bibliograficznych. W drodze wyjątku jedna lub dwie referencje do pozycji „monumentalnych” dla całej dziedziny mogą zostać zaakceptowane.
 - Nie ma praktyki umieszczania w tekście odwołań do objaśnień na dole strony. Wszystko należy objaśniać wprost, tj. bezpośrednio w tekście, albo należy umieszczać odnośniki do pozycji bibliograficznych.
 - We Wstępie, jak i w całej pracy, nie umieszcza się zdań napisanych w pierwszej osobie liczby pojedynczej, np. „Do testów wybrałem zadania należące do klasy...”. Generalnie należy wyrażać się bezosobowo, np. „Do testów wybrano zadania należące do klasy...” lub w trzeciej osobie, np. „Zbiory rozwiązań pozostają puste do pierwszego wystąpienia...” lub „Mianownik wyrażenia nie może przyjmować wartości zero, dlatego w części eksperymentalnej przyjęto...”. Nie trzeba się przy tym obawiać, że w wyniku tych zabiegów literackich zostanie rozmyta granica między własnym wkładem autora w pracę dyplomową, a rozwiązaniami pochodzącymi z literatury. W podsumowaniu musi znaleźć się bowiem specjalny akapit, w którym wymienione zostaną wszystkie, nawet najdrobniejsze elementy, które mają charakter oryginalny. Tego oczekuje recenzent.
 - Nie stosuje się zwrotów sugerujących niepewność autora ani trybu przypuszczającego, kiedy mowa o celu i zakresie wykonanej pracy. Przykład: zdanie „W pracy rozważano, czy zaproponowana modyfikacja algorytmu przyniesie poprawę...” należy zamienić na „W pracy pokazano, że zaproponowana modyfikacja algorytmu przynosi poprawę...”. Albo „W pracy próbowano znaleźć najlepsze ustawienia parametrów algorytmu znajdowania...” należy zamienić na „W pracy znaleziono ustawienia parametrów algorytmu dające najlepsze rezultaty dla wybranej klasy zadań...”. Wynika to z logiki wydarzeń. Kiedy zaczynamy badania, to przypuszczamy, że nas do czegoś ciekawego doprowadzą. Natomiast kiedy opisujemy wykonane badania, wszystko jest już wiadome. A poza tym praca dyplomowa to nie powieść sensacyjna, w której autor do ostatniej strony stara się trzymać czytelnika w niepewności.
-

Uwaga: wykorzystanie narzędzi opartych na sztucznej inteligencji w pisaniu pracy dyplomowej określa Zarządzenie Nr 70/2024 Rektora Uniwersytetu Kardynała Stefana Wyszyńskiego w Warszawie z dnia 8 listopada 2024 r. Pełny tekst Zarządzenia jest do pobrania z Monitora UKSW pod adresem: <https://monitor.uksw.edu.pl/docs/8047>

Paragraf 11, pkt. 2 tego Zarządzenia stanowi:

Jeśli student lub doktorant korzystał w pracy akademickiej z generatorów treści, wówczas musi oświadczyć, w jakim zakresie z nich korzystał. Informacja na ten temat powinna być uwzględniona we wstępie publikacji albo w pierwszym przypisie.

gdzie przez „generator treści” należy rozumieć każdy komputerowy system wykorzystujący technologie sztucznej inteligencji, który przetwarza, analizuje i samodzielnie generuje treści na podstawie danych i wzorców, w tym, ale nie tylko, takie systemy oparte o generatywne modele językowe, jak ChatGPT, Microsoft Co-Pilot, Google Gemini (paragraf 3, pkt. 3).

Każdy student przed przystąpieniem do pisania pracy ma obowiązek zapoznać się z tym zarządzeniem określającym politykę uczelni w zakresie zasad wykorzystywania narzędzi opartych na sztucznej inteligencji. Paragraf 13, pkt. 2 stanowi:

Prace akademickie sporządzone z naruszeniem niniejszego zarządzenia nie mogą być zaakceptowane przez nauczyciela akademickiego. Nie mogą być podstawą zaliczenia przedmiotu ani przedłożone do obrony.

Rozdział 1

Zdefiniowanie problemu badawczego

Tutaj jest miejsce na sprecyzowanie dziedziny, do której należy wykonana praca. Tu musi też znaleźć się przegląd literatury związanej z tą dziedziną. Tu również należy wstawić jak najwięcej referencji do bibliografii [10, 12].

1.1. Zawartość przeglądu literatury

Przez przegląd literatury należy rozumieć wymienienie i krótkie omówienie listy różnych podejść do tego samego zadania, również tych innych, niż to przedstawione w pracy [10, 12]. Np. dla pracy polegającej na napisaniu nowego algorytmu należy tu zamieścić przegląd znanych podejść algorytmicznych do danego problemu. Jeżeli praca dotyczy implementacji określonego systemu, to należy podać przegląd istniejących implementacji (np. dostępnych na rynku) realizujących podobną funkcjonalność. Jeżeli praca dotyczy implementacji wykorzystujących specyficzną technologię, należy podać istniejące inne technologie, które realizują podobną funkcjonalność. Jeżeli tematyka pracy obejmuje badania eksperymentalne, należy ten rozdział podzielić na dwa: jeden zawierający opis klas problemu, jaki będzie podlegał badaniom oraz drugi, zawierający opis wybranych narzędzi (ale tylko na poziomie koncepcyjnym i funkcjonalnym, tj. bez szczegółów technologicznych). Dlatego w skład tego rozdziału mogą (ale nie muszą) wchodzić takie sekcje, jak:

- przegląd literatury dotyczącej problemu przyjętego jako benchmark,
- przegląd literatury dotyczącej metod rozwiązywania wybranego benchmarku,
- przegląd literatury dotyczącej oceniania jakości rozwiązań (jeżeli jest kilka możliwości i nie wszystkie zostały zastosowane w pracy).

Zawarty tutaj przegląd może, a nawet powinien być krytyczny, tj. można wskazać wady lub braki istniejących podejść i na tym tle pokazać aktualne potrzeby dalszych prac rozwojowych oraz zalety własnego rozwiązania, które (w pewnym zakresie) odpowiada na te potrzeby.

1.2. Uwagi na temat kompletności danych bibliograficznych

Wszystko, każdy algorytm, wzór, czy nazwa własna powinna być „podparta” odpowiednim odniesieniem bibliograficznym, o ile nie jest własnym oryginalnym pomysłem dyplomanta. Wyjątki mogą dotyczyć pojęć tak powszechnie znanych, jak np. twierdzenie Pitagorasa czy bąbelkowy algorytm sortowania.

Wszystkie cytowane pozycje oprócz standardowych informacji powinny też mieć podany numer DOI¹, jeżeli go posiadają, zwłaszcza artykuły z czasopism [2] i z konferencji [3]. Numer DOI mogą też mieć techniczne raporty wydane np. w wydawnictwie uczelnianym [9]. W przeglądzie literatury mogą również pojawić się odwołania do publikacji nie posiadających numeru DOI, np. z Arxiv [5] i [8] albo innych, mniej znanych wydawnictw [1], a także np. do pracy doktorskiej [4] oraz repozytoriów oprogramowania [7] i zewnętrznych zbiorów danych [6]. W przypadku braku numeru DOI należy podać ścieżkę do *landing page* artykułu na stronie www wydawcy, czy też do odpowiedniej zakładki w repozytorium kodu lub baz danych, albo przynajmniej do miejsca, gdzie można znaleźć pełną wersję do pobrania. W razie braku numeru DOI i ścieżki w sieci do pełnej wersji, można jeszcze podać numer ISBN [10, 12]. Jeżeli nie można podać numeru DOI, ani numeru ISBN, ani ścieżki dostępu w sieci, to taki materiał nie może stanowić pozycji bibliograficznej.

Uwaga szczegółowa pierwsza: W rekordzie *bib* reprezentującym zasoby pobrane z sieci należy podać też dwie daty: datę powstania wersji zasobu, który został pobrany i wykorzystany przez autora oraz datę pobrania. Pierwszą podajemy w polu *date*, a drugą w polu *urldate*. Obydwa pola mają format RRRR-MM-DD. W drugim przypadku w bibliografii pracy zostanie automatycznie dodane słowo "dostęp", np. (dostęp 7.12.2025). Przykłady różnego rodzaju rekordów bibliograficznych w formacie *bib* są do obejrzenia w pliku *mgr.bib*.

Uwaga szczegółowa druga: W rekordzie *bib* reprezentującym artykuł z materiałów konferencyjnych, pole *address* zawiera nazwę miejsca, gdzie konferencja się odbyła, np. *address* = {Bonn, Germany} [3], natomiast w przypadku artykułu z czasopisma lub monografii to samo pole *address* zawiera adres siedziby wydawnictwa, np. *address* = {Amsterdam, Netherlands} [2]. W przypadku materiałów z konferencji adres wydawnictwa może być czasem podawany w polu *publisher*,

¹ Uwaga: numer DOI to np. 10.1016/j.robot.2013.09.004, a nie odpowiadający mu adres sieciowy <https://doi.org/10.1016/j.robot.2013.09.004>

np. `publisher = {Springer Berlin, Heidelberg}` [3]. Oczywiście informacje dotyczące adresów należy podać tylko wtedy, kiedy wydawca materiałów konferencyjnych bądź czasopisma udostępnił je w rekordzie *bib* lub na stronie www publikacji (jeżeli je umieścił, to znaczy, że chce, aby były podawane w cytowaniach).

Uwaga szczegółowa trzecia: W rekordzie *bib* reprezentującym zbiór danych dobrze jest na końcu tekstu pola `title` dodać słowo `[dataset]` [6]. Bez tego trudno z samego wyglądu pozycji bibliograficznej zorientować się, że odnosi się do zbioru danych.

1.3. Podział Bibliografii na sekcje

Kiedy bibliografia nie jest homogeniczna, tj. zawiera pozycje różnych typów, należy ją podzielić na sekcje i dopasować rekordy określonych typów do właściwej sekcji. Podstawowe trzy sekcje to:

1. **Literatura naukowa**, czyli książki i artykuły recenzowane z dziedziny badawczej, której dotyczy praca, zwykle mające swój numer DOI albo ISBN.
2. **Preprinty, narzędzia i biblioteki programistyczne, dokumentacja techniczna, standardy, bazy danych** to sekcja, do której wchodzi artykuły nierecenzowane i zwykle niemające numeru DOI, środowiska programistyczne różnych języków oraz odpowiednie biblioteki narzędzi, dokumentacja techniczna dotycząca bardziej złożonych środowisk programistycznych i bibliotek, bazy danych stosowane do uczenia lub testowania opracowanych programów. Wszystkie te pozycje należy zamieszczać pamiętając o obowiązkowym dołączeniu ścieżki dostępu, numeru wersji i daty pobrania.
3. **Zasoby internetowe i artykuły branżowe**, czyli strony internetowe, artykuły dotyczące badanych metod, ale prezentujące je od strony komercyjnej albo popularnonaukowej, a także publikacje dotyczące informatycznej strony przeprowadzonych badań, a nie samych badań (np. opis środowiska zarządzającego wykonaniem odpowiedniej serii eksperymentów i gromadzeniem danych wynikowych). Wszystkie te pozycje również należy zamieszczać ze szczególnym uwzględnieniem ścieżki dostępu, numeru wersji i daty pobrania.

Powyższą listę kategorii można rozszerzyć w miarę potrzeb.

Aby wprowadzić podział Bibliografii na sekcje, na początek należy do każdego rekordu *bib* w pliku *mgr.bib* dodać jeszcze jedną składową, `keywords`. W załączonym pliku *mgr.bib* zostały zastosowane trzy słowa kluczowe: `naukowe`, `narzedzia` i `inne`. Każdy rekord *bib* musi mieć przypisane jedno

i tylko jedno słowo kluczowe. Autor sam musi ocenić, jakiego typu jest cytowana pozycja, i sam wpisać właściwe słowo kluczowe.

Następnie w wywołaniach polecenia `printbibliography` na końcu dokumentu deklarujemy, że będzie to podsekcja sekcji Bibliografia podając w liście argumentów `heading=subbibliography`. Następnie dokonujemy odfiltrowania rekordów z pliku *mgr.bib* pod względem zgodności z jednym ze słów kluczowych za pomocą argumentu `keyword`, np. `keyword=narzedzia`. Do podsekcji trafią tylko te, które posiadają wskazane słowo kluczowe. Tym prostym sposobem każda podsekcja zawiera tylko pozycje bibliograficzne wskazanego typu.

Trzeba zastrzec, że gdyby okazało się jednak, że wszystkie albo prawie wszystkie wykorzystywane w pracy pozycje bibliograficzne są tego samego typu, np. literatura naukowa, opisany powyżej podział na sekcje jest zbędny.

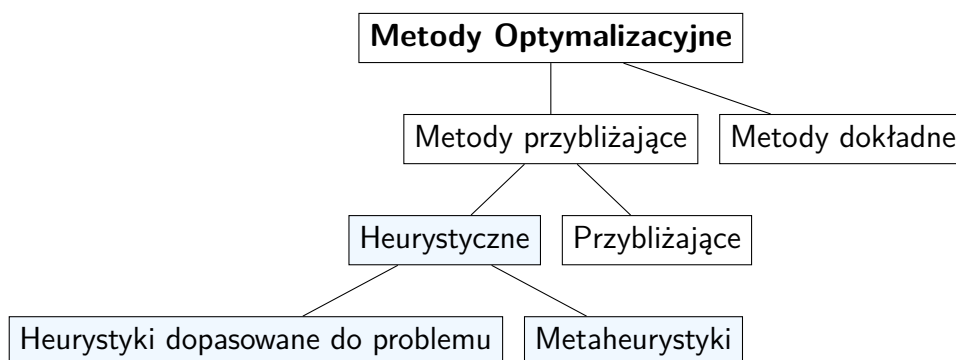
Rozdział 2

Przyjęte postępowanie

Tutaj należy w szczególności opisać, jak konkretnie został zrealizowany zakres prac o charakterze praktycznym, tj. w przypadku implementacji ogólny schemat aplikacji, pseudokody i schematy blokowe, przepływ sterowania, przepływ danych, struktura bazy danych, lista bibliotek zewnętrznych, jakie zostały wykorzystane, oraz wymagania platformy systemowej niezbędnej do uruchomienia aplikacji, pełna lista parametrów sterujących oraz wyniki testów obciążeniowych.

W przypadku wykorzystania rozwiązań już istniejących (bibliotek kodu bądź gotowych aplikacji) bardzo ważne jest uzasadnienie dokonanego wyboru. Taki przegląd bibliotek i aplikacji oraz uzasadnienie wyboru należy umieścić najpierw, tj. przed szczegółowym opisem prac o charakterze programistycznym.

Opis struktury hierarchicznej np. bibliotek kodu albo aplikacji, warto umieszczać bezpośrednio w tekście, jeżeli zawiera informacje szczególnie znaczące dla dalszego opisu. Ładnie jest wyeksponować go na rysunku (np. Rysunek 2.1) korzystając z pakietu forest.



Rysunek 2.1. Klasyfikacja metod optymalizacyjnych

2.1. Co opisać w przypadku badań eksperymentalnych

Dla badań eksperymentalnych należy opisać w szczegółach nową metodę, algorytm lub nowy mechanizm w już istniejącym algorytmie, jaki został stworzony w ramach pracy. Wszelkie pseudokody i schematy blokowe objaśniające, na czym ta propozycja autorska polega, są jak najbardziej na miejscu. Dobrym pomysłem może być zamknięcie opisu tej nowinki w oddzielnym podrozdziale mającym tytuł „propozycja autorska”.

W pracy magisterskiej nie umieszcza się natomiast kodu programu we fragmentach ani tym bardziej w całości, ponieważ to nie jest umiejętność podlegająca ocenie na poziomie tytułu magistra informatyki. Umiejętnością programowania należało wykazać się na poziomie obrony pracy licencjackiej, a teraz jej posiadanie jest już poza dyskusją. Wyjątek stanowią wybrane fragmenty kodu stanowiące kluczowe szczegóły algorytmu, których nie da się opisać za pomocą pseudokodu np. ze względu na specyfikę języka programowania.

Uwaga natury ogólnej: w pseudokodzie nie podaje się wartości progowych, np. 0.45 albo 0.9, tylko stosuje się symbole reprezentujące stałe. Natomiast w tekście, omawiającym ten pseudokod, piszemy np., że w eksperymentach przyjęto, że stałe ... będą miały przypisane zawsze następujące wartości, odpowiednio:

Tak samo jak rysunki i tabele, również wszystkie pseudokody muszą być wykorzystane w tekście, tj. w tekście muszą występować zdania, które odwołują się do tych pseudokodów.

2.2. Zasady opisu dotyczące stylu i formy

2.2.1. Przykłady odwołań do elementów graficznych w tekście

W pracy dyplomowej technicznej nie stosuje się zwrotów „na Rysunku poniżej...” lub „w Tabeli powyżej...” tylko korzysta z numerów (w dokumencie latex’owym wszystkie tabele, rysunki i algorytmy mają automatycznie nadawane numery). Aby odwołać się za pomocą numeru, konieczne jest nadanie etykiety elementowi graficznemu za pomocą instrukcji `\label`, np. `\label{etykieta}` i odwołanie się do tej etykiety za pomocą instrukcji `\ref`, np. `\ref{etykieta}`.

Słowa takie jak **Tabela**, **Rysunek**, **Algorytm**, **Twierdzenie**, czy **Definicja** piszemy zawsze wielką literą, gdy się do nich odwołujemy korzystając z ich numeru, tj. np. za pomocą instrukcji `\ref{etykietaTabeli}`.

2.2.2. Odwołanie do Wzoru – Przykład

Miara acc jest obliczana za pomocą wzoru (2.1):

$$acc(\mathbf{x}_j) = \frac{F(\mathbf{x}_j) - f_{\min,j}}{f_{\max,j} - f_{\min,j}}, \quad (2.1)$$

gdzie $f_{\max,j}$ and $f_{\min,j}$ reprezentują odpowiednio najlepszą i najgorszą znaną wartość dla j -tego kształtu krajobrazu dopasowania opisanego funkcją F .

2.2.3. Konwencja dotycząca fontów

W tekstach technicznych stosowana jest konwencja dotycząca fontów w symbolach:

1. Indeks w symbolu (wartość całkowita oznaczająca numer elementu w liście o ustalonej kolejności elementów) jest zapisywany kursywą, np. i w x_i , j w y^j , k w $F_{\min,k}$.
2. Indeksy górne lub dolne pisane bez kursywy nie reprezentują wartości, ale są to znaki, które są częścią symbolu. Np. $\max$ jest częścią symbolu $f_{\max}$ w $f_{\max,i}$, reprezentującym wartości maksymalną zmiennej f w i -tej generacji pewnego procesu.
3. Symbol reprezentujący wektor liczb jest zapisywany fontem pogrubionym bez kursywy, np. $\mathbf{x}$. Symbol $\mathbf{x}_j^t$ reprezentuje np. j -ty wektor $\mathbf{x}$ w kroku czasowym t , przy czym: x_k to k -ta współrzędna punktu $\mathbf{x}$ (odpowiednio $x_{j,k}^t$ to k -ta współrzędna j -tego wektora $\mathbf{x}$ w kroku czasowym t).
4. Zbiory elementów są oznaczane dużymi literami, tak jak wartości stałe. Np. P to populacja osobników w algorytmie ewolucyjnym, a np. S to zbiór rozwiązań niezdominowanych.
5. Wartości liczbowe stałe, typu liczba dni w tygodniu, albo godzin w ciągu doby, też są reprezentowane przez duże litery. Np. O może reprezentować rozmiar populacji osobników P , a np. N – liczbę wymiarów przestrzeni. W przypadku stałych, reprezentujących wartości graniczne np. dla dziedziny pewnego parametru, również stosujemy duże litery, ale z indeksem min albo max, np. $T_{\min}$ i $T_{\max}$ to dolna i górna granica czasu pracy badanego algorytmu. Aby uniknąć niejednoznaczności, moc skończonego zbioru zwykle nie jest reprezentowana przez swój własny symbol, ale przez operator $|\cdot|$, np. dla populacji P liczba osobników w tym zbiorze to $|P|$. Również długość wektora $\mathbf{x}$ w przestrzeni wielowymiarowej została by zapisana raczej jako $\|\mathbf{x}\|$.

2.2.4. Odwołanie do Algorytmu – Przykład

Algorytm 1 reprezentuje ogólny schemat **DE**.

Algorytm 1 Algorytm Ewolucji Różnicowej (Differential Evolution – DE) $P_{\mathbf{x}} \leftarrow$ populacja początkowaewaluacja (P)**repeat** $P_{\mathbf{v}} \leftarrow$ mutacja($P_{\mathbf{x}}$)▷ mutacja różnicowa osobników z $P_{\mathbf{x}}$ $P_{\mathbf{u}} \leftarrow$ rekombinacja($P_{\mathbf{x}}, P_{\mathbf{v}}$)▷ 'mieszanie' osobników populacji $P_{\mathbf{x}}$ z mutantami $P_{\mathbf{v}}$ evaluate($P_{\mathbf{u}}$)▷ ewaluacja osobników z populacji $P_{\mathbf{u}}$ $P_{\mathbf{x}} \leftarrow$ selekcja($P_{\mathbf{x}}, P_{\mathbf{u}}$)**until** spełniony warunek wyjścia**2.2.5. Zastosowanie znaków greckich w nagłówkach, np. α , β , itp.**

W przypadku zastosowania w nagłówku rozdziału lub sekcji takich znaków, dla których pakiet `hyperref` nie ma zdefiniowanego sposobu, jak je skonwertować do tekstu zakładki w wynikowym PDF i z tego powodu w trakcie kompilacji pojawia się ostrzeżenie: `Token not allowed in a PDF string`, należy użyć polecenia `\texorpdfstring{tekst dla latex}{tekst dla PDF}`.

Np. aby nadać sekcji nagłówek **The α -particle**, należy zamiast:

```
\section{The {\alpha}-particle}
```

napisać:

```
\section{The \texorpdfstring{\alpha}{alpha}-particle}
```

Rozdział 3

Metodologia badań

Ten rozdział jest niezwykle ważny w przypadku pracy o charakterze eksperymentalnym, tj. dla takich badań, w których wykazanie prawdziwości postawionej tezy badawczej sprowadza się do empirycznego pokazania, że nowy lub autorsko zmodyfikowany algorytm radzi sobie na pewnym zbiorze zadań testowych lepiej, niż inny, już istniejący algorytm. W tym rozdziale należy w poszczególnych podrozdziałach podać:

1. **definicje cech**, których pomiary były wykonywane podczas eksperymentów (np. czas odpowiedzi systemu, wartość najlepszego znalezionej rozwiązania, błąd odpowiedzi, itp.),
2. **sposób statystycznej obróbki zebranych pomiarów** (np. dla każdej serii pomiarów została policzona średnia z uzyskanych wartości, wariancja, mediana, wartości min i max, współczynnik korelacji, itp.). Uwaga: jeżeli zebrane wyniki dla różnych eksperymentów mają bardzo podobne wartości, należy tutaj opisać, jakie testy statystyczne zastosowano dla sprawdzenia, czy obserwowana różnica jest statystycznie istotna.
3. **zakres badań**, tj. opisać zbiór instancji problemów, dla jakich zostały przeprowadzone testy:
 - a) zakres i rozmiar problemów; można tu zamieścić np. listę funkcji testowych (Tabela 3.1) wraz z (niewielkimi!) wizualizacjami w 2D,
 - b) składnia plików z danymi (jeżeli była nietrywialna),
 - c) pochodzenie (jeżeli pochodzą np. z biblioteki problemów określonej klasy dostępnej w internecie),
 - d) wskazówki dotyczące ich obróbki wstępnej (jeżeli taka była konieczna),
 - e) modyfikacje lub adaptacje danych wejściowych (jeżeli były konieczne),
4. **konfiguracje algorytmów i zadań testowych**, tj. podać listę stałych oraz zmiennych parametrów instancji problemu oraz parametrów algorytmu bądź systemu, dla których zostały przeprowadzone eksperymenty, oraz dla każdego podać, jakie były te stałe wartości oraz do

Tabela 3.1. Wybrane funkcje testowe

nazwa	wzór	dziedzina
Pik (F_1)	$f(\mathbf{x}) = \frac{1}{1 + \sum_{j=1}^n x_j^2}$	$[-100, 100]$
Stożek (F_2)	$f(\mathbf{x}) = 1 - \sqrt{\sum_{j=1}^n x_j^2}$	$[-100, 100]$
Sfera (F_3)	$f(\mathbf{x}) = \sum_{i=1}^n x_i^2$	$[-100, 100]$
Rastrigin (F_4)	$f(\mathbf{x}) = \sum_{i=1}^n (x_i^2 - 10 \cos(2\pi x_i) + 10)$	$[-5, 5]$
Griewank (F_5)	$f(\mathbf{x}) = \frac{1}{4000} \sum_{i=1}^n (x_i)^2 - \prod_{i=1}^n \cos(\frac{x_i}{\sqrt{i}}) + 1$	$[-100, 100]$
Ackley (F_6)	$f(\mathbf{x}) = -20 \exp(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^n x_i^2}) - \exp(\frac{1}{n} \sum_{i=1}^n \cos(2\pi x_i)) + 20 + e$	$[-32, 32]$

jakich przedziałów należały te wartości, które zmieniały się w kolejnych eksperymentach. Można też tutaj podać plan eksperymentów, jeżeli było ich dużo, albo obejmowały bardzo różnorodne przypadki (istotnie różne klasy zadań benczmarkowych, różne algorytmy optymalizacyjne). Np. objaśnić, że część eksperymentalną można podzielić na kilka etapów, po czym opisać, jakie przypadki zadań testowych bądź algorytmów obejmował pierwszy etap, jakie drugi, etc.

Jeżeli zawartość podrozdziałów miałyby być bardzo krótka, np. tylko jedno albo dwa krótkie zdania (z różnych powodów, np. ponieważ większość została już szczegółowo i starannie opisana we wcześniejszym rozdziale i nie ma potrzeby tego powtarzać), to można zamiast podrozdziałów zrobić akapity.

Dla pracy, w której celem jest implementacja np. przysłowiowego sklepu internetowego, a nie badania eksperymentalne, zamiast powyższej listy podrozdziałów należy tutaj podać warunki, których spełnienie będzie jednoznaczne z uznaniem, że wykonana aplikacja spełniła postawiony przed nią cel, inaczej mówiąc, jak poznać, że udało się dobrze i prawidłowo zrobić dany system, serwis, interfejs, etc.

Rozdział 4

Wyniki

Tu jest miejsce na tabelki wykresy i zrzuty z ekranu. Uwaga: zrzuty z ekranu okien aplikacji nie powinny być przypadkowe: ich dobór musi być związany z przyjętą metodologią oceny wykonanej aplikacji i powinien potwierdzać spełnienie początkowych założeń.

Tabela 4.1. Statystyki rozmiaru populacji dla grupy 3 w czterech konfiguracjach algorytmu

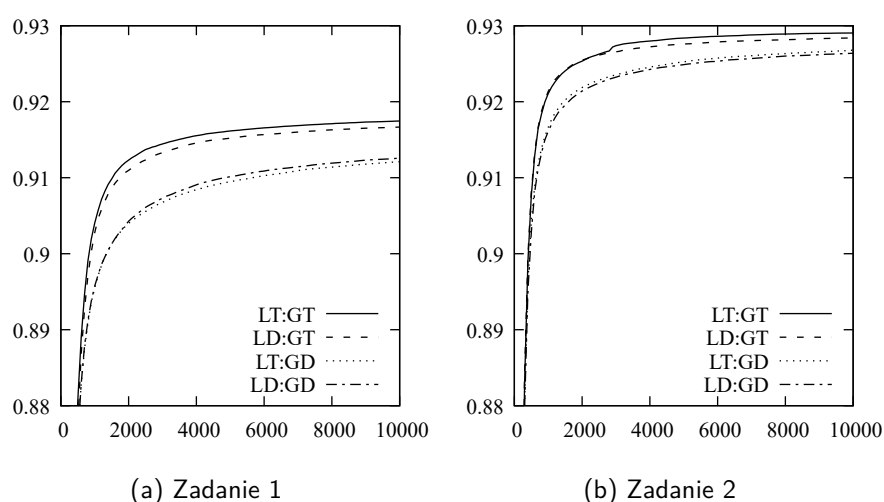
	Z_1	Z_2	Z_3	Z_4
Średnia min	1877.91	1877.91	1670.11	1563.27
Średnia max	1879.18	1879.18	1671.14	1564.53
Mediana min	319	319	324	319
Mediana max	319	319	324	319

Dla ułatwienia interpretacji zawartości tabelek i wykresów wszystkie zawarte w nich wyniki powinny być skomentowane w tekście pracy. Wartości rzeczywistoliczbowe w tabelkach (np. w Tabeli 4.1) podajemy tylko do tego miejsca po przecinku, kiedy zaczynają się pojawiać różnice między wartościami (np. dla liczb 1.23456789 i 1.23498765 wystarczy dokładność do 4. lub 5. miejsca po przecinku, czyli: 1.2346 i 1.2350), a wartości całkowitoliczbowe, np. rozmiar populacji, podajemy bez cyfr po przecinku.

Żaden rysunek, tabelka ani zrzut z ekranu nie może pozostać bez odwołania do niego w tekście. Oprócz referencji do rysunku bądź tabelki musi być też jawnie napisane, na co należy zwrócić uwagę patrząc na daną grafikę.

Przykład nawiązania w tekście do Rysunku 4.1:

Wykresy średniej wartości funkcji $f(\mathbf{x})$ dla dwóch zadań testowych dla czterech wariantów algorytmu: LT:GT, LD:GT, LT:GD i LD:GD przedstawia Rysunek 4.1, gdzie wykres 4.1a

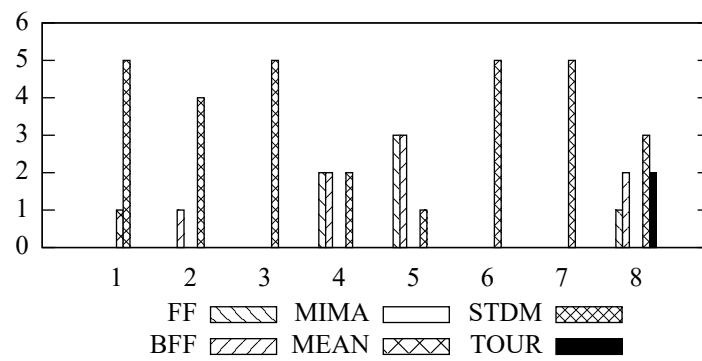


Rysunek 4.1. Wykresy średniej wartości $f(\mathbf{x})$ dla dwóch zadań testowych

prezentuje wartości uzyskane dla Zadania 1, prostszego, natomiast wykres 4.1b – wartości dla Zadania 2, trudniejszego.

Na koniec tego rozdziału, lub, jeżeli testy zostały podzielone na kilka dużych grup, to po każdej grupie wyników testów, konieczna jest dyskusja: ocena wiarygodności wyników, ich oryginalności, porównanie z poprzednimi albo z innymi, znanymi z literatury, itp. Przy czym, pisząc taką dyskusję, nie wolno nadużywać słowa „wynik”. To dobre słowo, ale – poza swoim ogólnym znaczeniem – nic nie wnosi do opisu (tak jak np. na Rysunku 4.2). Dlatego kiedy komentowane są wartości liczbowe, należy pisać konkretnie, co ta liczba reprezentuje: czy jest to najlepsza znaleziona wartość funkcji ewaluacyjnej, czy może średnia liczba wywołań funkcji ewaluacji potrzebna do spełnienia warunku stopu algorytmu, czy wartość maksymalna w serii najlepszych znalezionych wartości funkcji ewaluacyjnej, czy też jeszcze inna wartość. Pisanie zdań o treści „Wariant A algorytmu dał średni wynik 0.265 dla benchmarku 1 oraz 0.541 dla benchmarku 2” sprawia, że tekst jest niejasny: który wynik? jaki wynik? co za wynik?

Uwaga: określenie „ilość” stosujemy wyłącznie dla rzeczowników niepoliczalnych, np. „ilość mleka”, czy „ilość mąki” („liczba mąki” nie ma sensu). Natomiast dla policzalnych stosujemy „liczba” (np. liczba powtórzeń).



Rysunek 4.2. Wyniki dla ośmiu zadań testowych (co? jakie wyniki? co było mierzone? czy to średnia czy mediana, min lub max? jakich ośmiu, jeżeli w legendzie na dole jest sześć symboli?)

Podsumowanie

Tutaj należy opisać jeszcze raz całą wykonaną pracę (nie tę napisaną na poprzednich stronach, ale tę wykonaną, tj. albo eksperymentalną, albo implementacyjną). Ten opis powinien zająć około pół strony. Należy go pisać z myślą o recenzencie, który już jest po lekturze całości, tj. z perspektywy osoby, która podsumowuje i zbiera w całość tak jakby najważniejsze wydarzenia z dopiero co wspólnie odbytej podróży. Należy też w oddzielnym akapicie bardzo wyraźnie – jeżeli ktoś chce, to nawet w podpunktach – podkreślić wszystkie elementy stanowiące oryginalny wkład autora pracy. Może to być cokolwiek, co nie pochodzi z gotowych bibliotek kodu i podręczników, ale zostało samodzielnie wymyślone i napisane. Uwaga: w formularzu recenzji jest rubryka zatytułowana „Czy i w jakim zakresie praca stanowi nowe ujęcie?”. Jeżeli dyplomant nic nie napisze o swoim indywidualnym wkładzie, to recenzent napisze „Brak nowego ujęcia.”, a to nie będzie dobre dla końcowej oceny pracy. Następnie należy rozwinąć wszystkie spostrzeżenia, wnioski i odkrycia, jakie udało się poczynić podczas realizacji pracy. To rozwinięcie musi być bardzo staranne i niczego, żadnej, nawet najdrobniejszej rzeczy nie należy pominąć. Prawie na pewno niektóre spostrzeżenia poczynione w poprzednim rozdziale powtórzą się, ale to nic nie szkodzi.

Na koniec należy przypomnieć hipotezę badawczą i jeszcze raz potwierdzić, że udało się wykazać jej prawdziwość.

Całość tekstu podsumowania nie powinna przekroczyć dwóch stron.

PS: Pisząc pracę, należy się liczyć z tym, że recenzent nie będzie miał ochoty jej czytać całej (teoretycznie nie wolno recenzentom tak postępować i jest to wysoce nieetyczne, ale w praktyce – jak to sprawdzić?...). W taki sposób czyta np. recenzent, który już trochę zna temat (sprawdzał już wiele prac o podobnym zakresie, wie co nieco np. na temat typowych problemów testowych oraz rodziny algorytmów, które zostały użyte w pracy) i nie chce mu się czytać opisów tego, co dobrze zna. Dlatego praca musi zachowywać sens

nawet jeżeli czytana jest fragmentami. Typowe dwa schematy czytania przez leniwego albo przygniecionego obowiązkami i zabieganego recenzenta to:

- 1. wstęp, autorska modyfikacja algorytmu, podsumowanie wyników, zakończenie,*
- 2. wstęp i zakończenie.*

Proszę na koniec zrobić test i przeczytać swój tekst wg tych dwóch schematów. Jeżeli to, co się przeczytało, jest niezrozumiałe, ponieważ narracja się rwie i brak jakichś kluczowych informacji, to znaczy, że jest źle napisane. Trzeba uzupełnić.

Bibliografia

Literatura naukowa

- [1] Gedas Bertasius, Heng Wang i Lorenzo Torresani. „Is Space-Time Attention All You Need for Video Understanding?” W: *Proceedings of the 38th International Conference on Machine Learning*. Oprac. Marina Meila i Tong Zhang. T. 139. Proceedings of Machine Learning Research. [konferencja online]: PMLR, lipiec 2021, s. 813–824. URL: <https://proceedings.mlr.press/v139/bertasius21a.html>.
- [2] Enric Galceran i Marc Carreras. „A survey on coverage path planning for robotics”. W: *Robotics and Autonomous Systems* 61.12 (grudzień 2013), s. 1258–1276. ISSN: 0921-8890. DOI: [10.1016/j.robot.2013.09.004](https://doi.org/10.1016/j.robot.2013.09.004).
- [3] Ali Oran i in. „Location and Routing Models for Emergency Response Plans with Priorities”. W: *Future Security*. Oprac. Nils Aschenbruck i in. Communications in Computer and Information Science (CCIS). Bonn, Germany: Springer Berlin, Heidelberg, wrzesień 2012, s. 129–140. ISBN: 978-3-642-33161-9. DOI: [10.1007/978-3-642-33161-9_20](https://doi.org/10.1007/978-3-642-33161-9_20).
- [4] Kiran Venkata Panchamgam. „Essays in Retail Operations and Humanitarian Logistics”. PhD thesis. College Park, MD, USA: University of Maryland, 2011. URL: <http://hdl.handle.net/1903/12103> (dostęp 7.12.2025).

Preprinty, narzędzia i biblioteki programistyczne, dokumentacja techniczna, standardy, bazy danych

- [5] Marah Abdin i in. *Phi-4 Technical Report*. 2024. arXiv: [2412.08905](https://arxiv.org/abs/2412.08905) [cs.CL]. URL: <https://arxiv.org/abs/2412.08905>.

- [6] Filipe Batista e Silva, Lewis Dijkstra i Hugo Poelman. *The JRC-GEOSTAT population grid [dataset]*. Wersja 2018. Eurostat, 13 kwietnia 2021. URL: <https://ec.europa.eu/eurostat/web/gisco/geodata/population-distribution/population-grids> (dostęp 7.12.2025).
- [7] Ollama Team. *deepseek-coder-v2:16b*. <https://ollama.com/library/deepseek-coder-v2:16b>. 2026. (Dostęp 7.06.2026).
- [8] Hugo Touvron i in. *LLaMA: Open and Efficient Foundation Language Models*. 2023. arXiv: 2302.13971 [cs.CL]. URL: <https://arxiv.org/abs/2302.13971>.
- [9] Eckart Zitzler, Marco Laumanns i Lothar Thiele. *SPEA2: Improving the strength pareto evolutionary algorithm*. TIK Report 103. Maj 2001. DOI: [10.3929/ETHZ-A-004284029](https://doi.org/10.3929/ETHZ-A-004284029).

Zasoby internetowe i artykuły branżowe

- [10] David Lindsay. *Dobre rady dla piszących teksty naukowe*. Oficyna Wydawnicza Politechniki Wrocławskiej, 1995. ISBN: 83-7085-145-2.
- [11] Krzysztof Trojanowski. *Ramowe wymagania*. 3 czerwca 2026. URL: <https://ktrojanowski.blog.uksw.edu.pl/prace-dyplomowe/ramowe-wymagania/> (dostęp 12.06.2026).
- [12] Stanisław Urban i Wiesław Ładoński. *Jak napisać dobrą pracę magisterską, wydanie czwarte uzupełnione*. Wydawnictwo Akademii Ekonomicznej im. Oskara Langego we Wrocławiu, 2001. ISBN: 83-7011-160-2.

Dodatek A

Opis kodu, modułów i klas

W przypadku implementacji obiektowej lub modularnej należy podać opis klas lub modułów jakie współtworzą program. Można też podać wybrane, kluczowe fragmenty kodu programu, np. Kod [A.1](#) czy też Kod [A.2](#).

Kod A.1. Kod algorytmu sortowania bąbelkowego w języku Python

```
def sortowanie_babelkowe(lista):
    n = len(lista)

    for i in range(n):
        zamiana = False

        for j in range(0, n - i - 1):

            if lista[j] > lista[j + 1]:
                lista[j], lista[j + 1] = lista[j + 1], lista[j]
                zamiana = True

        # w całej pętli wewnętrznej nie było żadnej zamiany,
        # to znaczy, tablica jest posortowana
        if not zamiana:
            break

    return lista

# zastosowanie
liczby = [64, 34, 25, 12, 22, 11, 90]
print("Lista przed sortowaniem:", liczby)
posortowana_lista = sortowanie_babelkowe(liczby)
```

```
print("Lista po sortowaniu:  ", posortowana_lista)
```

Kod A.2. Kod algorytmu sortowania bąbelkowego w języku C++

```
void bubbleSort(std::vector<int>& arr) {  
    int n = arr.size();  
    bool zamiana;  
  
    for (int i = 0; i < n - 1; ++i) {  
        zamiana = false;  
        for (int j = 0; j < n - i - 1; ++j) {  
            if (arr[j] > arr[j + 1]) {  
                std::swap(arr[j], arr[j + 1]);  
                zamiana = true;  
            }  
        }  
        // w całej pętli wewnętrznej nie było żadnej zamiany,  
        // to znaczy, tablica jest posortowana  
        if (!zamiana) {  
            break;  
        }  
    }  
}
```

Dodatek B

Opis metadanych

W tym dodatku można zamieścić informację np. o składni danych przechowywanych w plikach json, tak jak to pokazuje Kod B.1.

Kod B.1. Struktura plików JSON przechowujących dane testowe

```
{  
  "n": int, # number of nodes in the graph  
  "type": "sar", # search & rescue flag for internal testing  
  "edge_cost": float[n][n], # distances between regions  
  "node_cost": float[n-1], # region size  
  "node_reward": float[n-1], # region population  
}
```
