

**Algorytmy i struktury danych**

*Laboratorium sprawozdania*

Prowadzący: Autor:

mgr inż. Ewa Żesławska Jakub Dobek

w67227

Kierunek: IID/2022, grupa GL01

Rzeszów r.a. 2022/2023

Spis treści

[**1.** Laboratorium 1 3](#_Toc133251676)

[**2.** Laboratorium 2 7](#_Toc133251677)

[**3.** Laboratorium 3 8](#_Toc133251678)

[4. Laboratorium 4 11](#_Toc133251679)

[5. Laboratorium 5 13](#_Toc133251680)

[6. Laboratorium 6 15](#_Toc133251681)

[7. Laboratorium 7 17](#_Toc133251682)

**Poniższe informacje należy usunąć**

*Informacje:*

*Format arkusza A4, marginesy: górny 2,5 cm, dolny 2,5cm, prawy – 2,5 cm; lewy – 2,5 cm; czcionka – Times New Roman 11; styl – normalny; odstęp między wierszami – 1,5. Tekst wyjustowany.*

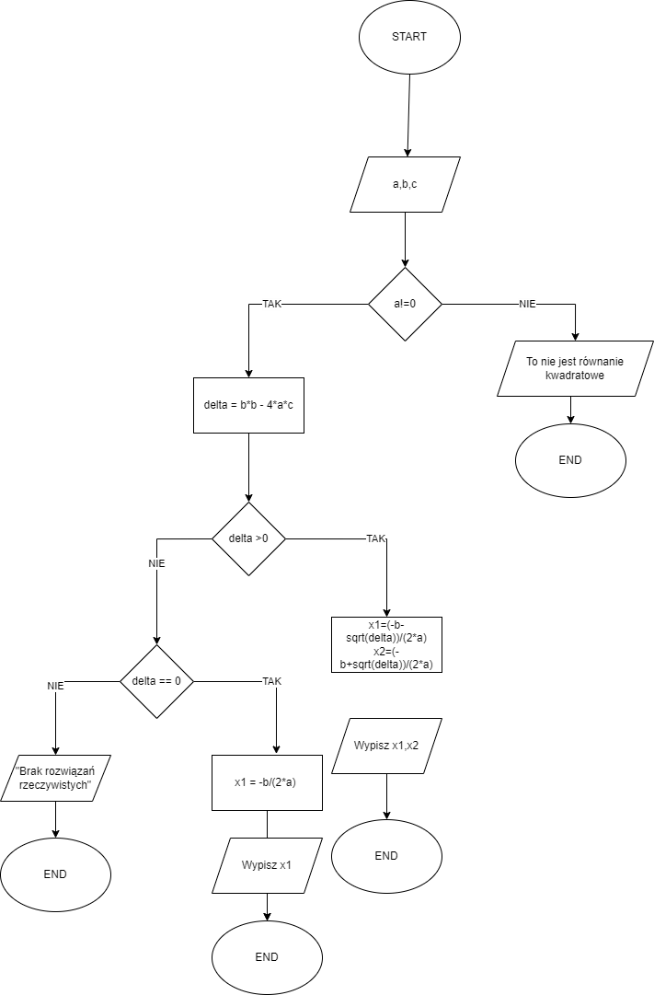
*Spis treści generowany automatycznie.*

*Podpis pod rysunkiem generowany poprzez wstawienie podpisu.*

1. Laboratorium 1

**Zadanie 1.**

Na Rysunku 1 przedstawiono algorytm wyznaczania pierwiastków równania kwadratowego. Implementacja algorytmu w języku Python umieszczona jest w repozytorium pod adresem [www.git.pl](https://github.com/Jo3i/Algorytmy-i-Struktury-Danych) [[1]](#footnote-1).

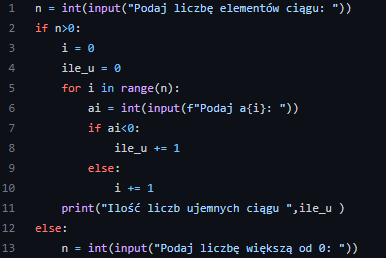


Rysunek 1: Schemat Blokowy algorytmu wyznaczenia pierwiastków równania kwadratowego

**Zadanie 2.**

Zaproponuj algorytm wczytywania ciągu n liczb całkowitych (N>0) i wyznaczania ilości liczb ujemnych w tym ciągu.

1. Wczytaj zmienną „n” z klawiatury
2. Sprawdź czy n>0
3. Zmiennym „i” oraz „ile\_u” przypisać 0
4. Pętla for z stopem na liczbę „n”
5. Wczytaj wartości „ai” do ciągu o długości n
6. Sprawdź czy ai<0
7. Jeżeli ai<0 to dodaj 1 do „ile\_u”
8. Jeżeli nie, to dodaj 1 do „i”
9. Wyjdź z pętli, wypisz „ile\_u”
10. Jeżeli n<=0, wypisz alert o wpisaniu liczby dodatniej „n”



Rysunek 2: Algorytm wczytywania ciągu n liczb całkowyitych i wyznaczania ilości ujemnych w tym ciągu w Python'ie

**Zadanie 3.**

Na Rysunku 2 przedstawiono algorytm sprawdzania czy podana wartość jest w tablicy jednowymiarowej. Implementacja algorytmu w języku Python umieszczona jest w repozytorium pod adresem [www.git.pl](https://github.com/Jo3i/Algorytmy-i-Struktury-Danych)

Obraz zawierający diagram

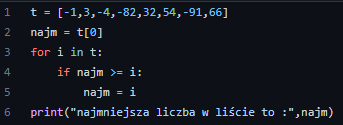
Opis wygenerowany automatycznie

Rysunek 3: Schemat blokowy algorytmu zadania spradzania czy podana wartość występuje w tablicy jednowymiarowej

**Zadanie 4.**

Zaprojektuj algorytm wyszukiwania w tablicy jednowymiarowej minimalnej wartości.

1. Wpisz „na twardo” tablicę
2. Przypisz do zmiennej „najm” pierwszy element tablicy
3. Pętla for do ilości wartości tablicy
4. Jeżeli zmienna „najm” jest >= od zmiennej „i”
5. Przypisz do „najm” zmienną „i”
6. Wypisz „najm”



Rysunek 4: Kod do zadania 4 w Python'ie

**Zadanie 5.**

Zaprojektuj algorytm wyszukiwania w tablicy dwuwymiarowej minimalnej wartości w każdym wierszu. Po znalezieniu minimalnej wartości wstaw ją na początek danego wiersza (poprzez zamianę miejsc).

1. Stwórz tablice dwuwymiarową „t”
2. Wypisz „t”
3. Pętla for w zasięgu „i” do „t”
4. Do zmiennej „najm” przypisać wartość minimalną z „i”
5. Do zmiennej „z” przypisać index „i” o wartości „najm”
6. Zmienić kolejność wartości tablicowych i[0], i[z] = i[z], i[0]
7. Wypisać tablice „t”

Obraz zawierający tekst

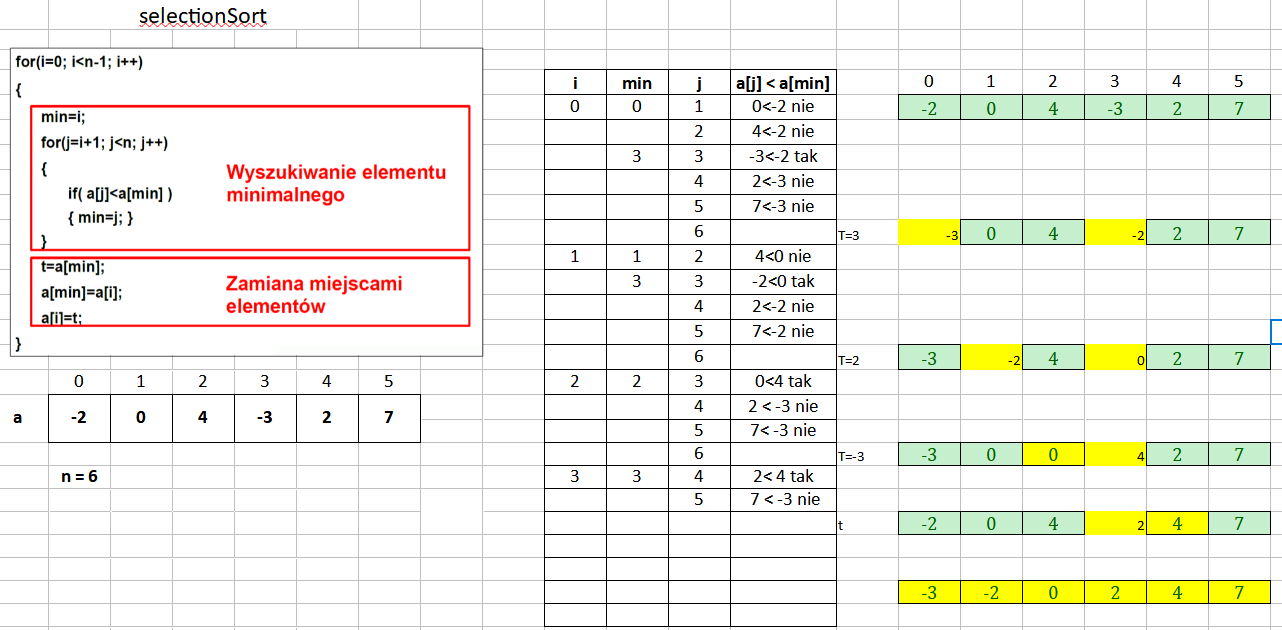
Opis wygenerowany automatycznie

Rysunek 5: Kod zadania 5 w Python'ie

# Laboratorium 2

**Zadanie 5.**

Na Rysunku 6 przedstawiono wykonane zadanie 5 z arkusza kalkulacyjnego Excel wykorzystywanego podczas drugich labolatoriów AiSD. Wykonane zadanie w pliku xlsx umieszczone jest w repozytorium pod adresem [www.git.pl](https://github.com/Jo3i/Algorytmy-i-Struktury-Danych)



Rysunek 6: Screen Excela z wykonanym zadaniem 5

**Zadanie 6.**

Na Rysunku 7 przedstawiono wykonane zadanie 6 z arkusza kalkulacyjnego Excel wykorzystywanego podczas drugich labolatoriów AiSD. Wykonane zadanie w pliku xlsx umieszczone jest w repozytorium pod adresem [www.git.pl](http://www.git.pl)

Obraz zawierający wykres

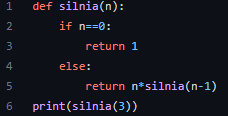
Opis wygenerowany automatycznie

Rysunek 7: Screen z Excela z wykonanym zadaniem 6

# Laboratorium 3

**Zadanie 1.**

Na Rysunku 8 przedstawiono wykonane zadanie 1 „Zaproponuj rekurencyjny algorytm obliczania silni dla liczby całkowitej dodatniej n„. Wykonane zadanie w pliku .py umieszczone jest w repozytorium pod adresem www.git.pl

****

Rysunek 8: Kod obliczający silnie w Python'ie

**Zadanie 2.**

Dana jest następująca funkcja rekurencyjna

Obraz zawierający tekst

Opis wygenerowany automatycznie

Rysunek 9: Podana funkcja zadania 2

**Obraz zawierający stół

Opis wygenerowany automatycznie**

Rysunek 10: Wykonana tabela zadania 2

**Zadanie 3.**

Dana jest funkcja f określona wzorem rekurencyjnym:

**Obraz zawierający stół

Opis wygenerowany automatycznie**

Rysunek 11: Wykonana tabela zadania 3

**Zadanie 4.**

Zaproponuj rekurencyjny algorytm zamiany liczby dziesiętnej na binarną. Należy zaprojektować schemat blokowy oraz implementacje. Kod pod nazwą „L3 zd.6” znajduje się na [www.git.pl](https://github.com/Jo3i/Algorytmy-i-Struktury-Danych)

Obraz zawierający diagram

Opis wygenerowany automatycznie

Rysunek 12: Schemat blokowy przedstawiający zmiany liczby dziesiętnej na binarne

**Zadanie 5.**

Zapoznaj się z problemem wieży Hanoi, a następnie zaproponuję listę kroków, schemat blokowy oraz implementację algorytmu.

Kod reprezentujący algorytm znajduje się w repozytorium w linku na 3 stronie.

Kroki algorytmu rozwiązującego problem wieży Hanoi dla n dysków i trzech kijów:

1. Przenieś n - 1 dysków z kija A na kij B, wykorzystując kij C jako kij pomocniczy.
2. Przenieś największy dysk (dysk n) z kija A na kij C.
3. Przenieś n - 1 dysków z kija B na kij C, wykorzystując kij A jako kij pomocniczy.

**Schemat blokowy wieży Hanoi:**

Obraz zawierający diagram

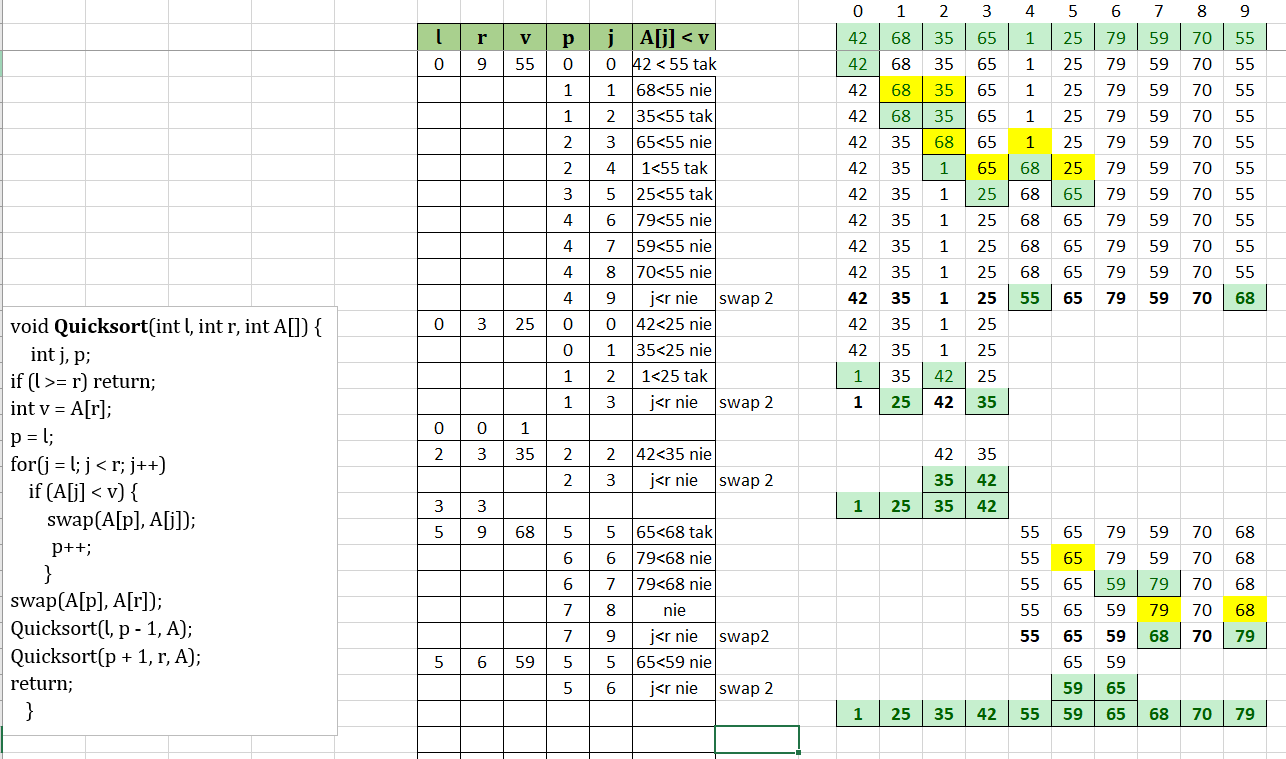
Opis wygenerowany automatycznie

Rysunek 13: Schemat blokowy wieży Hanoi

# Laboratorium 4

**Zadanie 1.**

**Zadanie 2**



Rysunek 14: Screen zadania 1 z pliku Excel z laboratorii 4

Obraz zawierający diagram

Opis wygenerowany automatycznie

Rysunek 15: Screen zadania 2 z pliku Excel z laboratorii 4

**Zadanie 3**

Obraz zawierający tekst

Opis wygenerowany automatycznie

Rysunek 16: Kod pythona przedstawiający wyznaczanie największego elementu wektora

**Zadanie 4**

**Obraz zawierający tekst

Opis wygenerowany automatycznie**

Rysunek 17: Kod pythona przedstawiający sume elementów tablicy metodą "dziel i zwyciężaj"

# Laboratorium 5

**Zadanie 1**

Korzystając z techniki programowania dynamicznego napisz program obliczania elementów ciągu Fibonacciego:

Obraz zawierający tekst

Opis wygenerowany automatycznie

Rysunek 18: Kod wykonujący ciąg Fibonacciego w Python'ie

**Zadanie 2**

Korzystając z techniki programowania dynamicznego wyznacz wartość wyrażenia:

Obraz zawierający tekst

Opis wygenerowany automatycznie

Rysunek 19: Treść wyrażenia do zadania 2

Obraz zawierający tekst, ekran

Opis wygenerowany automatycznie

Rysunek 20: Kod do zadania 2 w Python'ie

**Zadanie 3**

Dany jest ciąg o wyrazie ogólnym S(n) zdefiniowany rekurencyjnie:

Obraz zawierający wykres

Opis wygenerowany automatycznie

Rysunek 21: Treść wyrażenia do zadania 3

Obraz zawierający tekst

Opis wygenerowany automatycznie

Rysunek 22: Kod do zadania 3 w Python'ie

# Laboratorium 6

**Zadanie 1**

Napisz algorytm wczytywania ciągu dziesięciu liczb całkowitych i wyznaczania wartości średniej wczytanego ciągu liczb:

Obraz zawierający diagram

Opis wygenerowany automatycznie

Rysunek 23: Schemat Blokowy przedstawiający wyznaczania średniej z wczytanego ciągu liczb

Obraz zawierający tekst, ekran, zrzut ekranu

Opis wygenerowany automatycznie

Rysunek 24: Kod do zadania 1 w Python'ie

**Zadanie 2**

Narysuj schemat blokowy algorytmu wyświetlającego liczby parzyste z przedziału [a,b], gdzie a,b - liczby całkowite podane przez użytkownika:

Obraz zawierający tekst

Opis wygenerowany automatycznie

Rysunek 25: Schemat blokowy przedstawiający liczby parzyste z przedziału [a,b]

Obraz zawierający tekst

Opis wygenerowany automatycznie

Rysunek 26: Kod do zadania 2 w Python'ie

# Laboratorium 7

1. [↑](#footnote-ref-1)