Teaching material
Introducing programming environment & Exercises

MISCE project

Mechatronics for Improving and Standardizing Competences in Engineering

Competence: Computer programming

Workgroup: University de Castilla-La Mancha
University of Žilina

This document corresponds to the first exercises of ‘Computer programming’ competence: ‘Introducing programming environment & exercises’

Version: 1.0

Date: September 4th, 2024

Visit https://misceproject.eu/ for more information.

Index of contents

1 Installing programming environment........9
1.1 Python........9
1.2 C/C++ (with GCC or MinGW)........9
1.3 Java........9
1.4 Matlab........10
2 Exercises........11
2.1 Exercise 1........11
2.2 Exercise 2........11
2.3 Exercise 3........11
2.4 Exercise 4........11
2.5 Exercise 5........11
2.6 Exercise 6........11
2.7 Exercise 7........11
2.8 Exercise 8........11
2.9 Exercise 9........11
2.10 Exercise 10........11
2.11 Exercise 11........11
2.12 Exercise 12........11
2.13 Exercise 13........11
2.14 Exercise 14........12
2.15 Exercise 15........12
2.16 Exercise 16........12
2.17 Exercise 17........12
2.18 Exercise 18........12
2.19 Exercise 19........12
2.20 Exercise 20........12
2.21 Exercise 21........12
2.22 Exercise 22........12
2.23 Exercise 23........12
2.24 Exercise 24........12
2.25 Exercise 25........12
2.26 Exercise 26........12
2.27 Exercise 27........12
2.28 Exercise 28........12
2.29 Exercise 29........13

2.30 Exercise 30 13
2.31 Exercise 31 13
2.32 Exercise 32 13
2.33 Exercise 33 13
2.34 Exercise 34 13
2.35 Exercise 35 13
2.36 Exercise 36 13
2.37 Exercise 37 13
2.38 Exercise 38 13
2.39 Exercise 39 13
2.40 Exercise 40 13
2.41 Exercise 41 13
2.42 Exercise 42 13
2.43 Exercise 43 13
2.44 Exercise 44 14
2.45 Exercise 45 14
2.46 Exercise 46 14
2.47 Exercise 47 14
2.48 Exercise 48 14
2.49 Exercise 49 14
2.50 Exercise 50 14
2.51 Exercise 51 14
2.52 Exercise 52 14
2.53 Exercise 53 14
2.54 Exercise 54 14
2.55 Exercise 55 14
2.56 Exercise 56 14
2.57 Exercise 57 14
2.58 Exercise 58 14
2.59 Exercise 59 15
2.60 Exercise 60 15
2.61 Exercise 61 15
2.62 Exercise 62 15
2.63 Exercise 63 15
2.64 Exercise 64 15
2.65 Exercise 65 15
2.66 Exercise 66 15

2.67 Exercise 67 15
2.68 Exercise 68 15
2.69 Exercise 69 15
2.70 Exercise 70 15
2.71 Exercise 71 15
2.72 Exercise 72 15
2.73 Exercise 73 16
2.74 Exercise 74 16
2.75 Exercise 75 16
2.76 Exercise 76 16
2.77 Exercise 77 16
2.78 Exercise 78 16
2.79 Exercise 79 16
2.80 Exercise 80 16
2.81 Exercise 81 16
2.82 Exercise 82 16
2.83 Exercise 83 16
2.84 Exercise 84 16
2.85 Exercise 85 16
2.86 Exercise 86 16
2.87 Exercise 87 16
2.88 Exercise 88 17
2.89 Exercise 89 17
2.90 Exercise 90 17
2.91 Exercise 91 17
2.92 Exercise 92 17
2.93 Exercise 93 17
2.94 Exercise 94 17
2.95 Exercise 95 17
2.96 Exercise 96 17
2.97 Exercise 97 17
2.98 Exercise 98 17
2.99 Exercise 99 17
2.100 Exercise 100 17
2.101 Exercise 101 17
2.102 Exercise 102 17
2.103 Exercise 103 18

2.104 Exercise 104 ... 18
2.105 Exercise 105 ... 18
2.106 Exercise 106 ... 18
2.107 Exercise 107 ... 18
2.108 Exercise 108 ... 18
2.109 Exercise 109 ... 18
2.110 Exercise 110 ... 18
2.111 Exercise 111 ... 18
2.112 Exercise 112 ... 18
2.113 Exercise 113 ... 18
2.114 Exercise 114 ... 18
2.115 Exercise 115 ... 18
2.116 Exercise 116 ... 18
2.117 Exercise 117 ... 18
2.118 Exercise 118 ... 19
2.119 Exercise 119 ... 19
2.120 Exercise 120 ... 19
2.121 Exercise 121 ... 19
2.122 Exercise 122 ... 19
2.123 Exercise 123 ... 19
2.124 Exercise 124 ... 19
2.125 Exercise 125 ... 19
2.126 Exercise 126 ... 19
2.127 Exercise 127 ... 19
2.128 Exercise 128 ... 19
2.129 Exercise 129 ... 19
2.130 Exercise 130 ... 19
2.131 Exercise 131 ... 19
2.132 Exercise 132 ... 19
2.133 Exercise 133 ... 20
2.134 Exercise 134 ... 20
2.135 Exercise 135 ... 20
2.136 Exercise 136 ... 20
2.137 Exercise 137 ... 20
2.138 Exercise 138 ... 20
2.139 Exercise 139 ... 20
2.140 Exercise 140 ... 20

2.141 Exercise 141 20
2.142 Exercise 142 20
2.143 Exercise 143 20
2.144 Exercise 144 20
2.145 Exercise 145 20
2.146 Exercise 146 20
2.147 Exercise 147 20
2.148 Exercise 148 21
2.149 Exercise 149 21
2.150 Exercise 150 21
2.151 Exercise 151 21
2.152 Exercise 152 21
2.153 Exercise 153 21
2.154 Exercise 154 21
2.155 Exercise 155 21
2.156 Exercise 156 21
2.157 Exercise 157 21
2.158 Exercise 158 21
2.159 Exercise 159 21
2.160 Exercise 160 21
2.161 Exercise 161 21
2.162 Exercise 162 21
2.163 Exercise 163 22
2.164 Exercise 164 22
2.165 Exercise 165 22
2.166 Exercise 166 22
2.167 Exercise 167 22
2.168 Exercise 168 22
2.169 Exercise 169 22
2.170 Exercise 170 22
2.171 Exercise 171 22
2.172 Exercise 172 22
2.173 Exercise 173 22
2.174 Exercise 174 22
2.175 Exercise 175 22
2.176 Exercise 176 22
2.177 Exercise 177 22

2.178 Exercise 178 ... 23
2.179 Exercise 179 ... 23
2.180 Exercise 180 ... 23
2.181 Exercise 181 ... 23
2.182 Exercise 182 ... 23
2.183 Exercise 183 ... 23
2.184 Exercise 184 ... 23
2.185 Exercise 185 ... 23
2.186 Exercise 186 ... 23
2.187 Exercise 187 ... 23
2.188 Exercise 188 ... 23
2.189 Exercise 189 ... 23
2.190 Exercise 190 ... 23
2.191 Exercise 191 ... 23
2.192 Exercise 192 ... 23
2.193 Exercise 193 ... 24
2.194 Exercise 194 ... 24
2.195 Exercise 195 ... 24
2.196 Exercise 196 ... 24
2.197 Exercise 197 ... 24
2.198 Exercise 198 ... 24
2.199 Exercise 199 ... 24
2.200 Exercise 200 ... 24
2.201 Exercise 201 ... 24
2.202 Exercise 202 ... 24
2.203 Exercise 203 ... 24
2.204 Exercise 204 ... 24
2.205 Exercise 205 ... 24
2.206 Exercise 206 ... 24
2.207 Exercise 207 ... 24
2.208 Exercise 208 ... 25
2.209 Exercise 209 ... 25
2.210 Exercise 210 ... 25
2.211 Exercise 211 ... 25
2.212 Exercise 212 ... 25
2.213 Exercise 213 ... 25
2.214 Exercise 214 ... 25

2.215 Exercise 215 25
2.216 Exercise 216 25
2.217 Exercise 217 25
2.218 Exercise 218 25
2.219 Exercise 219 25
2.220 Exercise 220 25
2.221 Exercise 221 25
2.222 Exercise 222 25
2.223 Exercise 223 26
2.224 Exercise 224 26
2.225 Exercise 225 26
2.226 Exercise 226 26
2.227 Exercise 227 26
2.228 Exercise 228 26
2.229 Exercise 229 26
2.230 Exercise 230 26
2.231 Exercise 231 26
2.232 Exercise 232 26
2.233 Exercise 233 26
2.234 Exercise 234 26
2.235 Exercise 235 26
2.236 Exercise 236 26
2.237 Exercise 237 26
2.238 Exercise 238 27
2.239 Exercise 239 27
2.240 Exercise 240 27
2.241 Exercise 241 27
2.242 Exercise 242 27
2.243 Exercise 243 27
2.244 Exercise 244 27
2.245 Exercise 245 27
2.246 Exercise 246 27
2.247 Exercise 247 27
2.248 Exercise 248 27
2.249 Exercise 249 27
2.250 Exercise 250 27
2.251 Exercise 251 27

2.252 Exercise 252 27
2.253 Exercise 253 28
2.254 Exercise 254 28
2.255 Exercise 255 28
2.256 Exercise 256 28
2.257 Exercise 257 28
2.258 Exercise 258 28
2.259 Exercise 259 28
2.260 Exercise 260 28
2.261 Exercise 261 28
2.262 Exercise 262 28
2.263 Exercise 263 28
2.264 Exercise 264 28

Index of figures

Index of tables

1 Installing programming environment

1.1 Python

- Recommended Version: Python 3.11 or later
- Installation Instructions:
    - Go to https://www.python.org/downloads/
    - Download the installer for your OS (Windows, macOS, Linux)
    - During installation on Windows, check the box that says “Add Python to PATH”
    - Follow the setup wizard to complete installation
- Test (Basic Check):
    - Open a terminal or command prompt
    - Type: python --version
    - You should see: Python 3.11.x
    - Then type: python → Enter the interactive shell
    - Run:

```
print("Python is ready!")
```

1.2 C/C++ (with GCC or MinGW)

- Recommended Compiler: GCC 13.x (Linux/macOS), or MinGW-w64 for Windows
- Installation Instructions:
    - Download MinGW-w64 from https://www.mingw-w64.org/
    - Install and add bin/ folder to the system PATH
- Test (Basic Check):
    - Open terminal or command prompt
    - Type: gcc --version or g++ --version
    - Create a test file test.c with:

```
#include <stdio.h>
int main() {
    printf("C environment is working!\n");
    return 0;
}
```

- Compile with: gcc test.c -o test → Run with: ./test
- You should see: C environment is working!

1.3 Java

- Recommended Version: Java SE 17 (LTS version)
- Installation Instructions:
    - Download from Oracle: https://www.oracle.com/java/technologies/javase-downloads.html
    - Install the JDK and ensure the JAVA_HOME environment variable is set

        - Add the bin folder to your system PATH

- Test (Basic Check):
    - Open terminal or command prompt
    - Type: java -version → Should output java version "17.x"
    - Type: javac -version → Should output javac 17.x
    - Create a file Test.java:

```
public class Test {
    public static void main(String[] args) {
        System.out.println("Java is installed correctly!");
    }
}
```

- Compile: javac Test.java → Run: java Test
- You should see: Java is installed correctly!

1.4 Matlab

- Recommended Version: MATLAB R2023b or later
- Installation Instructions:
    - Access the installer from https://www.mathworks.com/downloads/
    - You need a license (student, institutional, or trial)
    - Run the installer and follow on-screen instructions
    - On first launch, log in with your MathWorks account and activate the license
- Test (Basic Check):
    - Open MATLAB
    - In the Command Window, type:

```
disp('MATLAB is ready!')
```

    - You should see: MATLAB is ready!

2 Exercises

2.1 Exercise 1

Write a script that prints the phrase "Hello, world!" on the screen. Use the appropriate display or output function.

2.2 Exercise 2

Create a program that displays the result of a basic arithmetic operation (addition, subtraction, multiplication, or division).

2.3 Exercise 3

Write a function that adds two numbers defined within the script and displays the result.

2.4 Exercise 4

Design a function that calculates the area of a circle given its radius.

2.5 Exercise 5

Create a program that determines whether a number is positive, negative, or zero.

2.6 Exercise 6

Write a function that returns the larger of two input numbers.

2.7 Exercise 7

Develop a program that calculates the average of three user-defined numbers.

2.8 Exercise 8

Create a function that receives a temperature in Celsius and returns its equivalent in Fahrenheit.

2.9 Exercise 9

Write a script that converts an amount in euros to dollars using a fixed exchange rate.

2.10 Exercise 10

Develop a program that reads two numbers and displays their sum, difference, and product.

2.11 Exercise 11

Create a program that calculates the area of a rectangle given base and height.

2.12 Exercise 12

Write a function that returns the volume of a cube given the side length.

2.13 Exercise 13

Design a function that checks whether a given year is a leap year.

2.14 Exercise 14

Create a script that calculates the Body Mass Index (BMI) from weight and height.

2.15 Exercise 15

Write a function that converts hours and minutes into total minutes.

2.16 Exercise 16

Create a program that converts minutes into hours and minutes (e.g., 130 min → 2 h 10 min).

2.17 Exercise 17

Develop a script that determines the absolute value of a number.

2.18 Exercise 18

Write a function that returns the square of a number.

2.19 Exercise 19

Create a program that determines whether a number is divisible by 2 and by 3.

2.20 Exercise 20

Write a script that receives a grade (0–10) and returns a qualitative evaluation (e.g., fail, pass, excellent).

2.21 Exercise 21

Create a program that determines whether a character is a vowel or a consonant.

2.22 Exercise 22

Write a function that calculates the final price of a product after applying a discount.

2.23 Exercise 23

Develop a program that calculates the perimeter and area of a triangle given the lengths of its sides.

2.24 Exercise 24

Write a script that determines the maximum and minimum of three numbers.

2.25 Exercise 25

Create a function that calculates the factorial of a number using a loop.

2.26 Exercise 26

Write a function that calculates the sum of the first n natural numbers.

2.27 Exercise 27

Develop a program that returns the multiplication table of a given number from 1 to 10.

2.28 Exercise 28

Create a script that determines whether a number is prime.

2.29 Exercise 29

Write a function that calculates the nth Fibonacci number using iteration.

2.30 Exercise 30

Develop a program that returns all prime numbers up to a given limit.

2.31 Exercise 31

Write a function that calculates the average of a list of grades and determines if it passes.

2.32 Exercise 32

Create a program that counts how many even and odd numbers exist in a given list.

2.33 Exercise 33.

Write a function that sums all elements in a numeric vector.

2.34 Exercise 34

Develop a script that calculates the square root of a positive number.

2.35 Exercise 35

Create a function that displays the absolute values of all elements in a vector.

2.36 Exercise 36

Write a program that finds the position of the largest value in a vector.

2.37 Exercise 37

Create a function that removes all negative values from a numeric vector.

2.38 Exercise 38

Write a script that calculates the standard deviation of a list of numbers.

2.39 Exercise 39

Develop a function that returns the number of elements greater than a given threshold in a vector.

2.40 Exercise 40

Write a program that counts how many times a specific number appears in a list.

2.41 Exercise 41

Create a function that reverses the elements of a list or array.

2.42 Exercise 42

Write a script that multiplies each element in a vector by a scalar.

2.43 Exercise 43

Develop a program that returns the sum of the diagonal elements in a square matrix.

2.44 Exercise 44

Create a function that identifies and returns all negative values in a matrix.

2.45 Exercise 45

Write a program that finds the maximum value in each row of a matrix.

2.46 Exercise 46

Create a function that generates a matrix with random integers between 1 and 100.

2.47 Exercise 47

Write a script that swaps two rows in a matrix.

2.48 Exercise 48

Develop a program that calculates the trace of a square matrix.

2.49 Exercise 49

Create a function that replaces all even numbers in a matrix with zero.

2.50 Exercise 50

Write a program that computes the sum of all elements in a two-dimensional array.

2.51 Exercise 51

Create a function that checks whether a square matrix is symmetric.

2.52 Exercise 52

Write a program that counts how many elements of a matrix are greater than its average.

2.53 Exercise 53

Develop a function that creates a matrix of size n × m filled with a given constant.

2.54 Exercise 54

Write a script that rotates a matrix 90 degrees clockwise.

2.55 Exercise 55

Create a program that calculates the average of the values on the main diagonal of a square matrix.

2.56 Exercise 56

Write a function that finds the row with the greatest sum in a matrix.

2.57 Exercise 57

Develop a script that replaces each value in a matrix with its absolute value.

2.58 Exercise 58

Create a function that returns the smallest value in each column of a matrix.

2.59 Exercise 59

Write a script that transposes a matrix.

2.60 Exercise 60

Create a program that generates a matrix of random integers and displays only the values above 50.

2.61 Exercise 61

Write a function that generates a random list of integers and returns the number of elements greater than a threshold.

2.62 Exercise 62

Create a script that generates 100 random numbers between 1 and 10 and counts how many times each value appears.

2.63 Exercise 63

Develop a function that simulates a lottery draw with unique random numbers.

2.64 Exercise 64

Write a script that simulates flipping a coin 1000 times and reports the number of heads and tails.

2.65 Exercise 65

Create a program that randomly selects a name from a list of participants.

2.66 Exercise 66

Write a function that generates a matrix of random floating-point numbers between 0 and 1.

2.67 Exercise 67

Develop a script that shuffles the elements of a given list randomly.

2.68 Exercise 68

Create a function that simulates a die roll and displays the result.

2.69 Exercise 69

Write a script that generates a random password of specified length using letters and digits.

2.70 Exercise 70

Develop a program that creates a list of n random integers and displays the list sorted in ascending order.

2.71 Exercise 71

Write a function that opens a text file and counts the number of characters, words, and lines.

2.72 Exercise 72

Create a script that saves a given list of numbers to a text file.

2.73 Exercise 73

Write a program that reads a file of numbers and calculates their average.

2.74 Exercise 74

Develop a function that writes the multiplication table of a given number to a file.

2.75 Exercise 75

Create a function that appends a line of text to the end of an existing file.

2.76 Exercise 76

Write a script that reads a text file and prints its content line by line.

2.77 Exercise 77

Create a function that deletes all empty lines from a text file.

2.78 Exercise 78

Write a script that reads data from a file and plots the values if they are numeric.

2.79 Exercise 79

Develop a program that checks whether a file exists before trying to read it.

2.80 Exercise 80

Create a script that copies the content of one text file into another.

2.81 Exercise 81

Write a function that counts how many vowels appear in a given sentence.

2.82 Exercise 82

Create a script that removes all whitespace from a given string.

2.83 Exercise 83

Develop a function that checks whether a string is a palindrome.

2.84 Exercise 84

Write a program that reverses a given text string.

2.85 Exercise 85

Create a script that replaces all spaces in a sentence with underscores.

2.86 Exercise 86

Write a function that returns the number of words in a given sentence.

2.87 Exercise 87

Develop a function that checks if a string contains only digits.

2.88 Exercise 88

Write a script that converts all letters in a sentence to uppercase.

2.89 Exercise 89

Create a function that returns the position of the first occurrence of a specific character in a string.

2.90 Exercise 90

Write a program that replaces every vowel in a sentence with an asterisk.

2.91 Exercise 91

Create a script that reads a string and prints its length.

2.92 Exercise 92

Write a function that removes all vowels from a text string.

2.93 Exercise 93

Develop a program that determines whether a password is valid (must contain upper/lowercase letters and a number).

2.94 Exercise 94

Create a script that joins two given strings with a hyphen.

2.95 Exercise 95

Write a function that counts how many times a given word appears in a paragraph.

2.96 Exercise 96

Create a program that compares two strings and returns whether they are identical.

2.97 Exercise 97

Develop a function that converts a string of digits into a list of integers.

2.98 Exercise 98

Write a script that counts how many words start with a specific letter in a sentence.

2.99 Exercise 99

Create a function that checks whether a text contains a specific word.

2.100 Exercise 100

Write a program that removes punctuation from a given text.

2.101 Exercise 101

Write a script that calculates the factorial of a number using recursion.

2.102 Exercise 102

Create a function that returns the first n even numbers.

2.103 Exercise 103

Develop a program that determines if a number is a perfect square.

2.104 Exercise 104

Write a function that returns the number of divisors of an integer.

2.105 Exercise 105

Create a script that returns the greatest common divisor (GCD) of two numbers.

2.106 Exercise 106

Write a program that determines the least common multiple (LCM) of two integers.

2.107 Exercise 107

Create a function that sums all the digits of a given number.

2.108 Exercise 108

Write a program that converts a decimal number into binary representation.

2.109 Exercise 109

Develop a function that converts a binary number into decimal.

2.110 Exercise 110

Write a script that returns the number of digits of a positive integer.

2.111 Exercise 111

Create a function that reverses the digits of a positive integer.

2.112 Exercise 112

Write a program that checks if a number is an Armstrong number.

2.113 Exercise 113

Create a script that calculates the sum of the series 1 + 1/2 + 1/3 + ... + 1/n.

2.114 Exercise 114

Write a function that calculates the sum of the squares of the first n natural numbers.

2.115 Exercise 115

Develop a program that evaluates a polynomial for a given value of x.

2.116 Exercise 116

Write a script that simulates a basic calculator (addition, subtraction, multiplication, division).

2.117 Exercise 117

Create a function that determines whether a given number is part of the Fibonacci sequence.

2.118 Exercise 118

Write a program that converts degrees to radians.

2.119 Exercise 119

Create a script that converts radians to degrees.

2.120 Exercise 120

Develop a function that generates a multiplication table up to 12×12.

2.121 Exercise 121.

Write a script that draws a right triangle using asterisks (*) based on a given height.

2.122 Exercise 122.

Create a function that returns the n-th triangular number.

2.123 Exercise 123.

Develop a program that determines whether three lengths can form a triangle.

2.124 Exercise 124.

Write a function that calculates the Euclidean distance between two points in 2D space.

2.125 Exercise 125.

Create a script that simulates a number guessing game.

2.126 Exercise 126.

Write a function that returns the sum of all prime numbers up to a given limit.

2.127 Exercise 127.

Develop a program that simulates a simple stopwatch (start/stop and elapsed time).

2.128 Exercise 128.

Write a script that generates a list of Fibonacci numbers up to a limit.

2.129 Exercise 129.

Create a function that determines the quadrant of a point (x, y) on the Cartesian plane.

2.130 Exercise 130.

Write a program that calculates the angle between two vectors.

2.131 Exercise 131.

Create a script that draws a square using asterisks of a given size.

2.132 Exercise 132.

Write a function that finds the second largest number in a list.

2.133 Exercise 133

Develop a program that computes the mean, median, and mode of a dataset.

2.134 Exercise 134.

Write a script that simulates rolling two dice 1000 times and analyzes the results.

2.135 Exercise 135

Create a function that determines if a list of numbers is sorted in ascending order.

2.136 Exercise 136

Write a program that calculates the harmonic mean of a set of values.

2.137 Exercise 137

Create a script that simulates a basic banking system (deposit, withdraw, balance).

2.138 Exercise 138

Write a function that converts a time in seconds to hours, minutes, and seconds.

2.139 Exercise 139

Develop a program that simulates a traffic light changing colors every few seconds.

2.140 Exercise 140

Write a script that displays a multiplication table from 1 to 10 in grid format.

2.141 Exercise 141

Write a function that generates the first n terms of a geometric sequence.

2.142 Exercise 142

Create a script that checks if a number is both prime and a palindrome.

2.143 Exercise 143

Develop a program that counts how many perfect squares are less than a given number.

2.144 Exercise 144

Write a function that returns the binary representation of a number as a string.

2.145 Exercise 145

Create a program that calculates the dot product of two vectors.

2.146 Exercise 146

Write a script that replaces negative values in a matrix with zero.

2.147 Exercise 147

Develop a function that returns the cumulative sum of elements in a list.

2.148 Exercise 148

Create a script that generates a matrix with alternating 1s and 0s.

2.149 Exercise 149

Write a program that creates a spiral matrix of size n × n.

2.150 Exercise 150

Create a function that validates a date (day, month, year) is correct.

2.151 Exercise 151

Write a script that simulates a basic login system with user verification.

2.152 Exercise 152

Develop a function that checks if a matrix is upper triangular.

2.153 Exercise 153

Create a program that multiplies two matrices of compatible sizes.

2.154 Exercise 154

Write a script that counts how many values in a matrix are divisible by 5.

2.155 Exercise 155

Create a function that determines the rank of a matrix.

2.156 Exercise 156

Write a program that swaps two columns in a matrix.

2.157 Exercise 157

Develop a script that sorts each row of a matrix in ascending order.

2.158 Exercise 158

Write a function that finds all local maxima in a matrix.

2.159 Exercise 159

Create a program that fills a matrix with random values and highlights the diagonal.

2.160 Exercise 160

Write a script that rotates a square matrix 180 degrees.

2.161 Exercise 161

Write a script that determines if a given matrix is a magic square.

2.162 Exercise 162

Create a function that checks whether two strings are anagrams.

2.163 Exercise 163

Develop a program that replaces each digit in a number with its word equivalent (e.g., 123 → one two three).

2.164 Exercise 164

Write a function that generates a chessboard pattern using 0s and 1s.

2.165 Exercise 165

Create a script that simulates the Rock, Paper, Scissors game between two players.

2.166 Exercise 166

Write a program that simulates the Monty Hall problem and displays statistics.

2.167 Exercise 167

Develop a script that creates an hourglass shape using asterisks with a given height.

2.168 Exercise 168

Write a function that checks if a square matrix is diagonally dominant.

2.169 Exercise 169

Create a script that counts how many prime numbers are present in a matrix.

2.170 Exercise 170

Write a function that calculates the determinant of a 2×2 or 3×3 matrix.

2.171 Exercise 171

Create a program that implements a basic Caesar cipher for encrypting text.

2.172 Exercise 172

Write a script that simulates a basic stopwatch using system clock functions.

2.173 Exercise 173

Develop a function that checks if a list contains duplicate elements.

2.174 Exercise 174

Write a program that finds the intersection and union of two sets.

2.175 Exercise 175

Create a function that finds the longest word in a sentence.

2.176 Exercise 176

Write a script that calculates the day of the week for a given date.

2.177 Exercise 177

Create a program that simulates a basic voting system with multiple candidates.

2.178 Exercise 178

Write a function that detects whether a number is a palindrome in binary form.

2.179 Exercise 179

Develop a script that sorts a list of words alphabetically.

2.180 Exercise 180

Create a program that generates random math problems and checks the user's answers.

2.181 Exercise 181

Write a function that returns the most frequent character in a text.

2.182 Exercise 182

Create a script that capitalizes the first letter of every word in a sentence.

2.183 Exercise 183

Develop a program that calculates the difference in days between two dates.

2.184 Exercise 184

Write a function that checks whether a string is a valid email address.

2.185 Exercise 185

Create a script that removes duplicate words from a sentence.

2.186 Exercise 186

Write a program that simulates a basic shopping cart system with item prices and quantities.

2.187 Exercise 187

Develop a function that determines whether a matrix contains only binary values (0 or 1).

2.188 Exercise 188

Write a script that creates a visual histogram of a list of numbers using asterisks.

2.189 Exercise 189

Create a function that extracts all digits from a mixed alphanumeric string.

2.190 Exercise 190

Write a program that encrypts a message by shifting characters according to a custom key.

2.191 Exercise 191

Create a script that determines whether a given year is a century year.

2.192 Exercise 192

Write a function that returns the smallest and largest elements in a numeric list.

2.193 Exercise 193

Develop a program that checks if a given point lies inside a circle with given center and radius.

2.194 Exercise 194

Write a script that generates a square pattern of alternating characters (e.g., X and O).

2.195 Exercise 195

Create a function that validates a password based on length and character criteria.

2.196 Exercise 196

Write a program that computes the moving average of a list of numbers.

2.197 Exercise 197

Create a script that extracts all uppercase words from a block of text.

2.198 Exercise 198

Write a function that removes consecutive duplicate characters from a string.

2.199 Exercise 199

Develop a script that detects and corrects misspelled words from a given list.

2.200 Exercise 200

Create a program that simulates a queue using a list or array.

2.201 Exercise 201

Write a function that computes the number of combinations (n choose k).

2.202 Exercise 202

Create a script that counts the frequency of each letter in a given text.

2.203 Exercise 203

Develop a program that splits a text into sentences based on punctuation.

2.204 Exercise 204

Write a function that generates a list of prime numbers using the Sieve of Eratosthenes.

2.205 Exercise 205

Create a script that simulates a parking lot management system.

2.206 Exercise 206

Write a program that analyzes the number of occurrences of each digit in a given number.

2.207 Exercise 207

Create a function that computes the Hamming distance between two strings.

2.208 Exercise 208

Write a script that converts a hexadecimal number to its decimal equivalent.

2.209 Exercise 209

Develop a function that computes the power of a number without using built-in exponentiation.

2.210 Exercise 210

Write a program that extracts the domain name from an email address.

2.211 Exercise 211

Create a script that finds all palindromic words in a sentence.

2.212 Exercise 212

Write a function that compresses a string using a simple run-length encoding.

2.213 Exercise 213

Develop a program that validates a credit card number using the Luhn algorithm.

2.214 Exercise 214

Write a script that detects the longest increasing subsequence in a list.

2.215 Exercise 215

Create a function that rotates a string by a given number of positions.

2.216 Exercise 216

Write a program that simulates a basic ticketing system for an event.

2.217 Exercise 217

Create a script that visualizes a list of values using a horizontal bar chart made of characters.

2.218 Exercise 218

Write a function that splits a string into substrings of a fixed length.

2.219 Exercise 219

Develop a program that returns the sum of ASCII values of the characters in a string.

2.220 Exercise 220

Create a script that computes the greatest difference between any two elements in a list.

2.221 Exercise 221

Write a function that computes the average word length in a given sentence.

2.222 Exercise 222

Create a script that finds the longest substring without repeating characters.

2.223 Exercise 223

Develop a program that validates an IP address format (IPv4).

2.224 Exercise 224

Write a function that generates a random hexadecimal color code.

2.225 Exercise 225

Create a script that simulates a random walk on a grid.

2.226 Exercise 226

Write a program that analyzes a paragraph and outputs word frequency in descending order.

2.227 Exercise 227

Create a function that detects whether a string contains balanced parentheses.

2.228 Exercise 228

Write a script that simulates a typing effect by printing one character at a time.

2.229 Exercise 229

Develop a function that finds the longest common prefix among a list of strings.

2.230 Exercise 230

Write a program that calculates the checksum digit of an ISBN-10 code.

2.231 Exercise 231

Create a script that visualizes numeric data as a vertical bar chart using characters.

2.232 Exercise 232

Write a function that generates an acronym from a given phrase.

2.233 Exercise 233

Develop a program that sorts a list of dates in chronological order.

2.234 Exercise 234

Write a script that converts a text to Pig Latin.

2.235 Exercise 235

Create a function that evaluates if a matrix is sparse (most elements are zero).

2.236 Exercise 236

Write a program that simulates Conway’s Game of Life on a small grid.

2.237 Exercise 237

Create a script that encodes and decodes text using a simple substitution cipher.

2.238 Exercise 238

Write a function that formats a number with commas as thousand separators.

2.239 Exercise 239

Develop a program that checks if a string follows a specific pattern (e.g., 'a*b*a').

2.240 Exercise 240

Write a script that highlights specific words in a sentence by surrounding them with asterisks.

2.241 Exercise 241

Write a function that finds all numbers in a list that appear more than once.

2.242 Exercise 242

Create a script that filters out all stopwords from a sentence.

2.243 Exercise 243

Develop a program that visualizes prime numbers on a spiral matrix.

2.244 Exercise 244

Write a function that removes HTML tags from a string.

2.245 Exercise 245

Create a script that calculates the entropy of a text string.

2.246 Exercise 246

Write a program that converts text into Morse code and vice versa.

2.247 Exercise 247

Develop a function that finds the longest palindromic substring in a given string.

2.248 Exercise 248

Write a script that generates a simple QR code from a string.

2.249 Exercise 249

Create a program that identifies the language of a short text snippet (basic keyword matching).

2.250 Exercise 250

Write a function that generates a histogram of letter frequencies from a text file.

2.251 Exercise 251

Create a script that formats a list of numbers into a nicely aligned table.

2.252 Exercise 252

Write a program that extracts email addresses from a block of text.

2.253 Exercise 253

Develop a function that computes Levenshtein distance between two strings.

2.254 Exercise 254

Write a script that parses a CSV file and outputs the average of numeric columns.

2.255 Exercise 255

Create a program that simulates an elevator system with multiple floors and buttons.

2.256 Exercise 256

Write a function that plots a sine wave using sampled points.

2.257 Exercise 257

Develop a script that generates and displays a calendar for a given month and year.

2.258 Exercise 258

Write a program that simulates a digital clock updating in real time.

2.259 Exercise 259

Create a function that validates the syntax of a simple arithmetic expression.

2.260 Exercise 260

Write a script that compares the performance of different sorting algorithms.

2.261 Exercise 261

Create a program that records user keystrokes and analyzes typing speed.

2.262 Exercise 262

Write a function that determines if a matrix is orthogonal.

2.263 Exercise 263

Develop a script that generates a maze and solves it using backtracking.

2.264 Exercise 264

Write a program that creates a heatmap from a matrix of values.