

C++ Functions, Pointers & Dynamic Memory

20 Exercises — Organised by Topic

Sections	
1–10	Functions & Pointers — functions, nested calls, pointers, matrices
11–20	Dynamic Memory — new, delete, 1D/2D/3D arrays, heap objects, growth

These exercises assume familiarity with the basics covered in the beginner booklet. Section 1 focuses on writing and composing functions and working with pointers. Section 2 covers dynamic memory — allocating and freeing heap memory, including 2D and 3D arrays.

Table of Contents

Section 1 — Functions & Pointers (exercises 1–10)

- 01. Simple Function (Functions) ★★★
- 02. Temperature Converter (Functions) ★★★
- 03. Power Function (Functions) ★★★
- 04. Grade Classifier (Functions) ★★★
- 05. Pointer to a Variable (Pointers) ★★★
- 06. Pointer to an Array (Pointers) ★★★
- 07. Pointer to a Structure (Pointers) ★★★
- 08. Matrix Construction and Arithmetic (Matrices) ★★■
- 09. Transpose and Symmetry Check (Matrices) ★★■
- 10. Determinant, Inverse and Linear System (Matrices) ★★★

Section 2 — Dynamic Memory (exercises 11–20)

- 11. Single Value on the Heap (Dynamic Memory) ★■■
- 12. Dynamic 1D Array (Dynamic Memory) ★★■
- 13. Dynamic 2D Array (Dynamic Memory) ★★■
- 14. Dynamic 3D Array (Dynamic Memory) ★★★
- 15. Dynamic Array of Objects (Dynamic Memory) ★★★
- 16. Swap via Pointers (Pointers) ★★■
- 17. Dynamic String Buffer (Dynamic Memory) ★★★
- 18. Pointer Arithmetic on a Heap Array (Dynamic Memory) ★★★
- 19. Resizable Array (Dynamic Memory) ★★★

Section 1 — Functions & Pointers

Exercise 1 of 20

Simple Function

★ ★ ★

Topic: Functions

Description

Write a function `int square(int n)` that returns the square of `n`. Call it from `main` for values 1 to 5 and print each result.

Hint

Define the function before main or use a prototype.

Expected Output

1 4 9 16 25

Your Solution



Hint

Expected Output

Fahrenheit: 212



Description

Write `char letterGrade(int score)` returning A/B/C/D/F. Write `bool isPassing(int score)` that calls `letterGrade` and returns true if not F. Read a score and print the letter grade and whether the student passed.

Hint

isPassing only needs to check letterGrade(score) != 'F'.

Expected Output

Score: 74
Grade: C
Passed: Yes

Your Solution

★ ★ ★

Description

Hint

Expected Output

```
After change -- x: 99 *p: 99
```

Pointer to an Array

★ ★ ★

Topic: Pointers / Array

Description

Declare an integer array `int arr[] = {10, 20, 30, 40, 50}`. Declare a pointer `int* p` that points to the first element. Use pointer arithmetic (no index notation `arr[i]`) to iterate through all elements and print them. Then use the pointer to double the value of the third element and print the full array again to confirm the change.

Hint

Point to the start with `int* p = arr;` — an array name decays to a pointer to its first element. Access elements as `*(p + i)` and modify the third with `*(p + 2) *= 2;`.

Expected Output

Elements: 10 20 30 40 50

After change: 10 20 60 40 50

Your Solution

★ ★ ★

Define *struct Point* with *double x* and *double y*. Create a *Point* and a pointer to it. Using only the pointer, set *x = 3.0* and *y = 4.0*, then compute and print the distance from the origin using *std::sqrt(x*x + y*y)*.

Use the arrow operator `p->x` to access members through a pointer.

```
Point: (3, 4)
Distance from origin: 5
```

★★■

Include `matrix.h`. Create two 2x2 `math::matrix<double>`: `A = {{1,2},{3,4}}` and `B = {{5,6},{7,8}}`. Print `A`, `A+B`, `A*B`, and `A*2.0` using the built-in operators and `<<`.

Declare as `math::matrix<double> A = {{1,2},{3,4}}`; All operators are overloaded.

```
A:
1 2
3 4

A+B:
6 8
10 12

A*B:
19 22
43 50
```

Transpose and Symmetry Check



Topic: Matrices / Transpose / IsSymmetric

Description

Create $M = \{\{1,2,3\},\{4,5,6\},\{7,8,9\}\}$ and $S = \{\{1,2,3\},\{2,5,6\},\{3,6,9\}\}$ as 3×3 `math::matrix<double>`.
For each: print it, print its transpose using `~`, and print whether it is symmetric using `IsSymmetric()`.

Hint

The `~` operator returns the transpose. `IsSymmetric()` returns bool.

Expected Output

```
M symmetric: No
S symmetric: Yes
Transpose of S equals S: Yes
```

Your Solution

★ ★ ★

Description

Hint

Expected Output

-1

Exercise 11 of 20

★

Description

Hint

Expected Output

Your Solution

★★■

Description

Hint

Expected Output

Sum: 55

★★■

Description

Hint

Expected Output

Your Solution

★ ★ ★

Description

Hint

Expected Output

```
Layer 1:
100 101 102 103
110 111 112 113
120 121 122 123
```

★ ★ ★

Description

Hint

Expected Output

Your Solution

★★■

Description

Hint

Expected Output

After: $x=9$ $y=5$

★ ★ ★

Description

Hint

Expected Output

Your Solution

★ ★ ★

Description

Hint

Expected Output

Reverse: 128 64 32 16 8 4 2 1

★ ★ ★

Description

Hint

Expected Output

Your Solution