



Name _____ Class _____ Date _____

AFTER THIS WORKSHEET YOU CAN

✓ Create arrays ✓ Dot and matmul ✓ Transpose and inverse in code ✓ Vectorised speed

 Section A · Concept Check • BEGINNER $4 \times 1 = 4$ _____

- 1** Dot product in NumPy:

☐ A. np.dot(a,b) ☐ B. a.plus(b) ☐ C. np.join

☐ D. a*b always
- 2** Matrix multiply operator:

☐ A. @ ☐ B. % ☐ C. & ☐ D. ::
- 3** Transpose of array A:

☐ A. A.T ☐ B. A.flip() ☐ C. A.rev ☐ D. tr(A)
- 4** Inverse comes from:

☐ A. np.linalg.inv ☐ B. np.flip ☐ C. np.inverse2

☐ D. math.inv

 Section B · Problem Solving • INTERMEDIATE $2 + 3 \times 3 = 11$ _____

- 5** np.array([1,2,3]).shape is (_____,).
6 np.zeros((2,3)) makes a 2x3 of _____.
7 Code: dot of [1,2] and [3,4].

- 8** Code: 2x2 identity times [[1,2],[3,4]].

- 9** Why is NumPy 100x faster than loops?

 Section C · Challenge • CHALLENGE 5 _____

- 10** Check a matrix times its inverse.

 Reflection — the most useful thing I learned: _____

A ___/4 B ___/11 C ___/5 **Total** ___/20

Teacher's Signature _____

Parent's Signature _____

⌘ ANSWER KEY — FOLD OR CUT BEFORE HANDING OUT

1-A 2-A 3-A 4-A | 5 3 6 zeros 7 = np.dot([1,2],[3,4]) → 11 8 = np.eye(2) @ A → A 9
= C-optimised vectorised operations on whole arrays | 10 = A @ np.linalg.inv(A) is approx np.eye(n)