

DPST1091 / CPTG1391

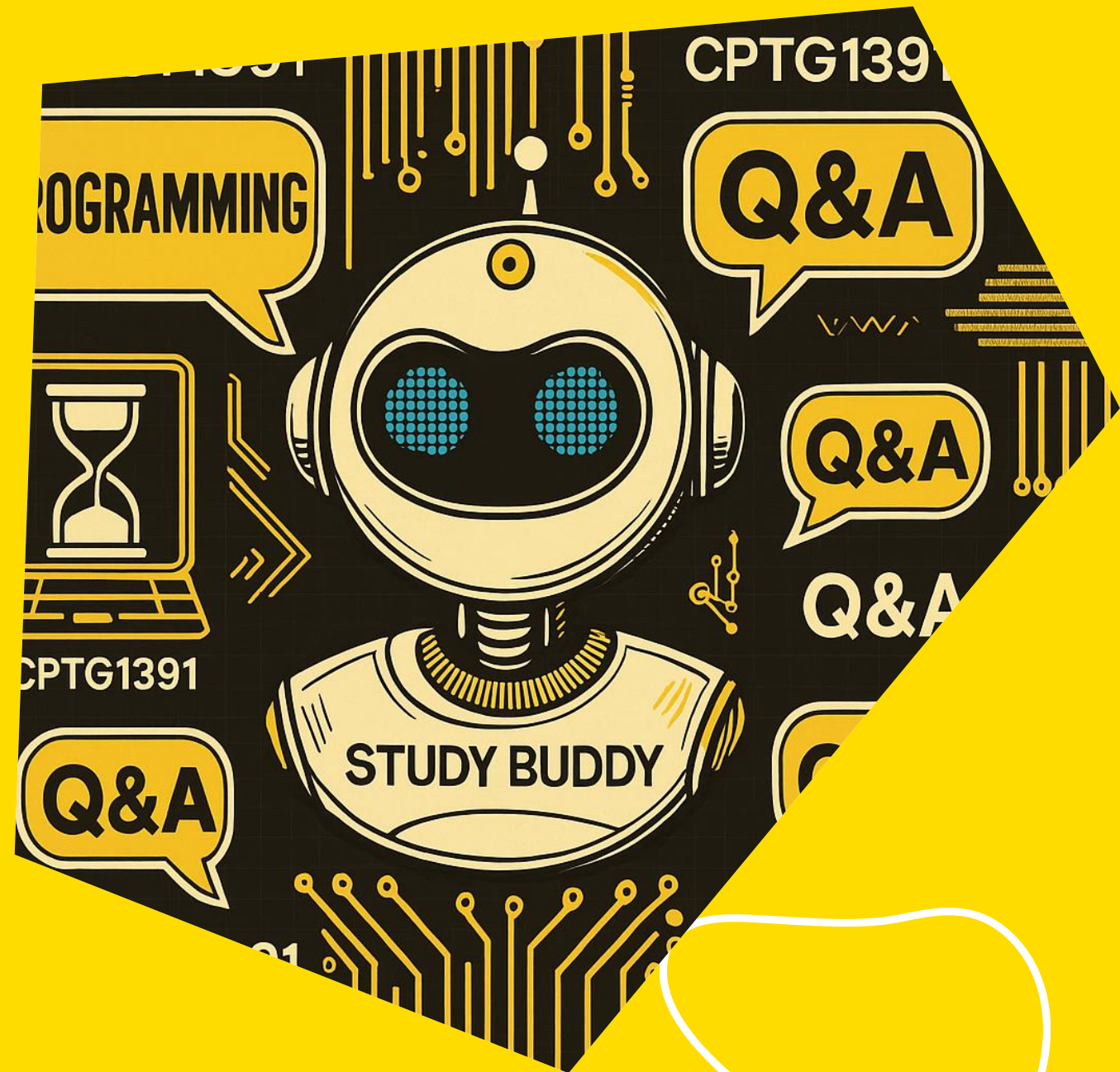
Introduction to Programming

Week 12 Lecture 1

Lecturer and Course Convener:

Dr Pantea Aria

More Revision



Agenda

- Today
- More exam hurdle questions

MyExperience Survey

Tell us what **worked** (and what **didn't**!) — I genuinely value your feedback



Link to the Live Lecture Code

<https://cgi.cse.unsw.edu.au/~dp1091/26T2/code/12>

Live lecture code is written for teaching, not perfection.
It may include extra comments and may not always follow
ideal coding style.



Linked List Hurdle Example (●○○)

File: list_find_divisible.c

Count how many values in the linked list are **divisible by a given number**.

Example:

```
./list_find_divisible 3 6 1 7 12  
2
```

```
int find_divisible(int value, struct node *head);
```

Linked List Hurdle Example (●○○)

File: linked_list_range.c

Find the **range** of a linked list, which is the difference between the largest and smallest values.

```
int range(struct node *head);
```

Linked List Hurdle Example (●○○)

File: list_even_difference.c

Given two linked lists, count the number of **even numbers** in each list and return the difference.

```
int even_difference(struct node *head1, struct node *head2);
```

Array Hurdle Example (●○○)

File: array_count_odd.c

Given two arrays, count how many positions contain **odd numbers in both arrays**.

```
int count_odd(int size1, int arr1[], int size2, int arr2[]);
```

Array Hurdle Example (●○○)

File: array_struct_capitals.c
Given an array of the following struct:

```
struct initials {  
    char first_initial;  
    char last_initial;  
};
```

Count how many entries **do not have both initials in uppercase.**

```
int capitals(int size, struct initials array[]);
```

Linked List Hurdle Example (●●◌)

File: linked_list_duplicate.c

Modify the linked list by **duplicating every node**, inserting a copy immediately after each original node.

```
struct node *duplicate(struct node *head);
```

Linked List Hurdle Example (●●◌)

File: linked_list_delete_duplicate.c
Remove the **first duplicate value** found in the list.

Example:

1 -> 3 -> 5 -> 3 -> 1 -> 7 -> X

becomes

1 -> 3 -> 5 -> 3 -> 7 -> X

```
struct node *delete_duplicate(struct node *head);
```

Debug Example (●○○)

File: debug_list.c

This program is intended to find the **smallest value** in a linked list.

If the list is empty, it should return -1.

Example:

```
$ ./debug_list 4 10 5 2 9
2

$ ./debug_list
-1
```

Debug Example (●○○)

File: debug_string.c

You are given code that should **combine two strings into one**.

⚠ You are NOT allowed to use strcpy or strcat.

Example:

```
$ ./debug_string
Enter the first string: Last
Enter the second string: Lecture
LastLecture
```

Voice of the Student

Anonymous ongoing feedback
Anything you wanted to share with me



Voice of the Student



[Voice of the Student – Fill out form](#)

See you soon ...