Grade 12 Computing

Unit 1 · Algorithms for Hard Problems

ANSWER KEY · FOR TEACHERS

Full marks 32

Give the marks for the points in the marking scheme; an answer in other words that says the same is right.

1 Key words

1. weighted
2. priority queue
3. relaxed
4. heuristic
5. greedy
6. memoization
7. dynamic programming
8. edit distance

Each carries 0.5 mark. **K · 4**

2 Multiple choice

i. **c.** the waiting town with the smallest distance found so far **K · 1**

Nearest first, by distance along the roads from the start, is the whole idea. Taking them in the order reached would be breadth-first search, which counts roads. Choosing by the distance left to the goal is a different idea, which comes later in this unit.

ii. **c.** It may stop with a route that is not the shortest. **U · 1**

Too high an estimate makes good towns look worse than they are, so the search can reach the goal along a longer way before it ever tries them, and stop. An estimate that is never too high keeps A* exact; the closer it is to the truth without going over, the fewer towns it visits.

iii. **b.** 10 + 1 + 1: three coins, but 6 + 6 is only two **A · 1**

Greedy takes the biggest coin that fits, 10, then 1 and 1: three coins. Two coins of 6 make 12 in two. (There is no 2 coin, so 10 + 2 can't be paid.)

iv. **b.** a problem whose answer is built from answers to smaller problems, where the same smaller problems are needed again and again **U · 1**

Dynamic programming pays off when small problems **overlap**: fewest(2) is needed by fewest(3), fewest(4), fewest(5) and fewest(6). Solve each once and remember it. Finding the largest number has no overlapping smaller problems, and where greedy is provably right, greedy is simpler and faster.

v. **c.** about 3 minutes **A · 1**

With the start fixed, 12 places have 11! ÷ 2 round trips and 14 places 13! ÷ 2. 13! ÷ 11! = 13 × 12 = 156, so about 156 seconds, close to 3 minutes. Two more places multiplied the work by 156.

3 Dijkstra's algorithm by hand

ANSWER

Visited	A	B	C	D	E	F
A	0	4	2	–	–	–
C	0	**3**	2	10	12	–
B	0	3	2	**8**	12	–
D	0	3	2	8	**10**	14
E	0	3	2	8	10	**13**
F	0	3	2	8	10	13

Science Playground, Katunje, Bhaktapur
Book a class visit: https://science-playground.com/booking

The figures in bold went down (were relaxed) at that visit. Fikkal is taken at 13 and the search stops. Following where each came from: **A → C → B → D → E → F, 13 km.** A · 4

MARKING SCHEME

- ✓ Villages visited in the order A, C, B, D, E, F (always the nearest waiting one).
- ✓ B goes down from 4 to 3 through Chisapani, and D from 10 to 8 through Bhanjyang.
- ✓ E goes down to 10 and F from 14 to 13, with F final only when it is visited.
- ✓ The route A → C → B → D → E → F, 13 km.

4 Edit distance

ANSWER A · 3

	–	b	a	t	h
–	0	1	2	3	4
b	1	0	1	2	3
h	2	1	1	2	2
a	3	2	1	2	3
t	4	3	2	1	2

i. The distance is **2**. "bhat" is "bath" with h moved two places, but this measure has no single edit for moving or swapping letters: it takes two edits (delete h, insert h at the end; or change two letters), the same as any word two changes away.

MARKING SCHEME

- ✓ The rows for b and h correct (0 1 2 3 and 1 1 2 2).
- ✓ The rows for a and t correct (2 1 2 3 and 3 2 1 2).
- ✓ Distance 2: there is no edit for moving or swapping letters, so the change counts as two.

5 Fewest coins, by a table

ANSWER A · 4

Amount a	`best[a]`	**Last coin**
0	0	–
1	1	1
2	2	1
3	3	1
4	1	4
5	1	5
6	2	1 (or 5)
7	3	1 (or 5)
8	2	4

i. best[8] = 2: the last coin is a 4, leaving 4, whose last coin is a 4: **4 + 4**, two coins. Greedy gives 5 + 1 + 1 + 1, **four** coins.

MARKING SCHEME

- ✓ 1, 2, 3 for amounts 1 to 3, and 1 for both 4 and 5.
- ✓ 2 for 6 and 3 for 7.
- ✓ 2 for 8, with last coin 4.
- ✓ 4 + 4, two coins, against greedy's four.

Science Playground, Katunje, Bhaktapur
Book a class visit: https://science-playground.com/booking

6 What does it print?

MARKING SCHEME **A · 3**

- ✓ `[5, 1, 1, 1]`: the biggest coin that fits, 5, then three 1s.
- ✓ `[5, 5, 1, 1, 1]`: the coins are sorted biggest first, whatever order they are given in.
- ✓ 7: 500, 100, 100, 50, 20, 10 and 5.

7 Find and fix the bug

MARKING SCHEME **A · 3**

- ✓ It stops when F is first reached (lines 19–20), at 14 through Dhunche, before the shorter way through Ekrani is found.
- ✓ The check moved to after `done.add(place)`: `if place == goal: return km`, with lines 19–20 removed.
- ✓ The corrected program prints 13.

8 Think

ANSWER **HA · 2**

She is right that it visits fewer places, but not that it always finds the shortest route. Where a road runs almost straight, its length is less than 1.5 times the straight line, so the estimate is sometimes **too high**. Then places on the real shortest route can look worse than they are, and the search may reach the goal along a longer way and stop before trying them, as three times the straight line chose 320 km through Pokhara instead of 305 km. A* is sure to find the shortest route only when the estimate is never too high.

MARKING SCHEME

- ✓ 1.5 times the straight line can be more than the real road distance (an estimate that is too high).
- ✓ Then A* can stop with a longer route: it is faster, but no longer sure to be shortest.

9 Write a program

ANSWER **A · 4**

```
def book_hall(requests):
    booked = []
    free_from = 0          # the hall is free from this hour
    # the request that finishes earliest comes first
    for start, end in sorted(requests, key=lambda r: r[1]):
        if start >= free_from:
            booked.append((start, end))
            free_from = end
    return booked

clubs = [(9, 11), (10, 12), (11, 13), (8, 10), (12, 14), (13, 16), (14, 15)]
print(book_hall(clubs))
```

```
OUTPUT
[(8, 10), (10, 12), (12, 14), (14, 15)]
```

MARKING SCHEME

- ✓ The requests sorted by their end time (or the earliest-finishing one found each time).
- ✓ A request is booked only if it starts at or after the end of the last one booked.
- ✓ The last end time kept up to date, and the booked events given back.
- ✓ Used on the list, giving four events: 8–10, 10–12, 12–14 and 14–15.

Science Playground, Katunje, Bhaktapur
Book a class visit: https://science-playground.com/booking

10 Activity: Plan the trek

(not marked)

WHAT STUDENTS SHOULD FIND

- Nearest next: 82.4 km, 19% longer than the best. It leaves village D, near the base, until last, then walks across the map to it.
- Then uncross: 70.6 km, about 2% longer than the best.
- Try every route: 69.1 km, the best of all 20,160 round trips.
- A careful order of your own is usually close to the uncrossed one; a route that crosses itself can always be made shorter.

Marks by level	Knowledge	Understanding	Application	Higher ability	**Total**
Marks	5	2	23	2	**32**

Science Playground, Katunje, Bhaktapur
Book a class visit: https://science-playground.com/booking