Section 0: Tell Me Who You Are

We will manually check Problem Set 0 and 1 in the required places.

Section 1: Multiple Choices with Only One Correct Answer

(2.5 pts * 16 = 40 pts)

2	**3**	**4**	**5**
B	A	A	B
6	**7**	**8**	**9**
D	A	B	B
10	**11**	**12**	**13**
D	A	D	C
14	**15**	**16**	**17**
B	A	B	D

Section 2: Multiple Choices with No Less Than One Correct Answer

(3.5 pts * 10 = 35 pts)

18	**19**	**20**	**21**	**22**
ABF	AD	ACDF	ABD	ADE
23	**24**	**25**	**26**	**27**
BE	ABCE	BCD	ABC	BCD

Note:

- Partial Scoring will be **disabled** in Problem Set 17-21
- Partial Scoring will be **enabled** in Problem Set 22-26
- Problem Set 23: B: 1.5 pts; E: 2 pts
- Problem Set 24: A, B, C: 1 pt each; E: 0.5 pt
- Problem Set 25: B: 1.5 pts; C, D: 1 pt each
- Problem Set 26: A: 1.5 pts; B, C: 1 pt each
- Problem Set 27: B, C: 1 pt each; D: 1.5 pts

Section 3: Blank Filling

(5 pts + 12 pts + 12 pts = 29 pts)

28. Grader

(5 pts)

Your Answer: `distance\s*>=?\s*100(\.0*)?`

Grading specs:

- (Part A) 0.5 pts for matched `distance`
- (Part B) 1 pt for matched both `>` and `>=` (0.5 pts for matched either)
- (Part C) 0.5 pts for matched `100`
- 1 pt for matched all floats (0.5 pts for matched some floats, or all floats but failed to match the pure integer `100`)
- 1 pt if Parts A, B, C are matched (or partially matched) in order, while all invalid answers are rejected (Note: if using `.` instead of `\.`, invalid answers occur)
- 1 pt if Parts A, B, C are matched (or partially matched) in order, while spacing is properly used at both places (0.5 pts if spaces are at both places but only consider some situations, e.g. using only one single space instead of variable length)
- However, the whole problem is graded as 0 pts if the answer does not look like a regular expression (e.g. writing Python expressions), or the regular expression is invalid somewhere (e.g. using `{}` improperly)
- The whole problem is graded as 0 pts if the answer looks too easy, like simply copying example expressions from the problem. However, if `\s` is added, plus 0.5 pts

29. Zip

(12 pts)

Blank Number	Your Answer
A	`iter(it)` or `it.__iter__()`
B	`next(self.iters[i])` or `self.iters[i].__next__()`
C	`self.remain -= 1` or `self.remain = self.remain - 1`
D	`= None`
E	`tuple(values)`

Note:

- A, B: 3 pts each
- C, D, E: 2 pts each

30. Number System Conversion

(12 pts)

Blank Number	Your Answer
A	`return str(value)`
B	`digit >= "A"` or `ord(digit)>=65` or `ord(digit)>=ord("A")` or similar answers
C	`1`
D	`input_base_shift *= input_base` or similar answers
E	`value2digit(decimal_value % output_base)+output`

- B, E: 3 pts each
- A, C, D: 2 pts each

Section 4: Integrated Problems

(12 pts + 5 pts + 8 pts + 14 pts + 5 pts = 44 pts)

31. Simple Shuttle Bus

(12 pts)

Find errors in your program

(4 pts)

Line No.	Your Explanation	Your Correction
5	IndentationError	add a `tab` before the initial statement
6	`boarded_passengers` should be a `list`	`boarded_passengers = []`
21	Passengers who arrive at the station before the shuttle bus are supposed to be boarded after the bus arrives.	`if len(result) != 0 and boarding_info[1] > depature_time:`

Scoring:

- 0 pt if the answer only indicates the right line number but gives wrong explanation and correction
- Line 5: Line No. and explanation: 0.5 pt; correction: 0.5 pt
- Line 6: Line No. and explanation: 0.5 pt; correction: 0.5 pt
- Line 21: Line No. and explanation: 1 pt; correction: 1 pt

Fill in the blanks

(4 pts)

Blank Number	Your Answer
A	`passenger_tuple[0]`
B	`self.boarding_info[1] + 1200` or `self.boarding_info[1] + self.round_trip_time`
C	`self.result[departure_time] = boarded_passengers`

Scoring:

- A and C: 1 pt each
- B: 2 pts

Debug with `stdin` and `stdout`

(4 pts)

Write down the operation system (and optionally the shell) you are using: `_____` (0 pt)

The following reference commands are given under Linux systems with bash shells. All possible answers consistent with the OS and the shell you write down are accepted.

First command:

```
# python3 shuttle_bus.py < testcase.in
```

Second command:

```
# python3 shuttle_bus.py < testcase.in > testcase.out
```

Third command:

```
# diff testcase.out testcase.ref
```

Scoring:

- 4 pts for answering all 3 commands correctly
- 2 pts for 2 correct commands
- 1 pt for 1 correct command

32. Left Outer Join

(5 pts)

Blank Number	Your Answer
A	`matched = True`
B	`chain(['.'.join([lname, key]) for key in left.keys()], ['.'.join([rname, key]) for key in right.keys()])`
C	`chain([lrow[key] for key in lrow.keys()], [None]*len(right.keys()))`

Scoring:

- A: 1 pt
- B, C: 2 pts each

33. Bad Table

(2 pts * 4 = 8 pts)

Line No.	Your Explanation	Your Correction
6	redundant white space before the statement	remove a white space before the statement of Line 6
7	code would crash if `len(rows) == 0`.	change line 7 to self.__keys = keys
24	forget to sort the lines with their primary key	add `self.__rows = sorted(self.__rows)` at the end of `__init__()`
27	does not reset `self.__iter_idx`, could not iterate for multiple times: broken iterator	add `self.__iter_idx = -1` before `return self`
55	open file with mode that does not support write	change to `open(fn, mode='w+')`

34. Determinant and Inverse

(14 pts)

Determinant

1. (2 pts)

Your Answer:

Yes.

$det(A) = \sum_{i=1}^{n}(-1)^{i+j}det(cross - col - row\ matrix\ of\ A)$

When calculating determinant of `cross-col-row matrix`, recursion will be used.

Scoring:

- Only `Yes` or `No` with no and wrong explanation will be graded as 0 point.
- The answer should explain how recursion is used in the calculation. If the answer only roughly explains how recursion is used, it will get only 1 point.

---

2. (4 pts)

Blank Number	Your Answer
A	`1` or `2`
B	`return mat_list[0][0]` or `return mat_list[0][0]*mat_list[1][1] - mat_list[0][1]*mat_list[1][0]`
C	`res += mat_list[0][numi] * ((-1) ** (numi % 2)) * determinant(Matx)`

Scoring:

- A and B: 1 pt each
- C: 2 pts

Note:

- Blank A and B are bundled when grading. In the other word, Blank A and B must be both correct to get 2 points. If either A or B is incorrect, A and B will be both graded as wrong answers
- Blank C: equivalent expressions will also be accepted.

3. (2 pts)

Your Answer:

`6.0000`

Inverse

4 (1 pt)	5 (1 pt)
B	D

6. (2 pts * 2 = 4 pts)

Line Number	Description of the Mistake	Your Correction for the Mistake
15	Inverse calculation for 2×2 matrices incorrect	`detx[0] = [mat_list[0][1], -mat_list[1][0]]`
2	Row and column number of the transpose matrix should be swapped	`return [[mat_list[b][a] for b in range(len(mat_list))] for a in range(len(mat_list[0]))]`
32	Negativity indicator should be `rowNum+colNum`	`detx[numi].append(determinant(detxx) * (-1) ** (numi+numj))`

Scoring:

- 0 pt if the answer only indicates the right line number but gives wrong explanation and correction
- Line 2: Line No. and explanation: 1 pt; correction: 1 pt
- Line 32: Line No. and explanation: 1 pt; correction: 1 pt

35. Tic-Tac-Toe

(5 pts)

Line No.	Your Explanation	Your Correction
18	Syntax Error! No `else if` in Python!	`elif player == 'x':`
5	by multiplying `[['_'] * 3]` by `3`, the three sub-lists in the list refer to the same object. Thus they are not independent to each other, and changing one will result in changes in the other two	`gameBoard = [['_'] * 3 for _ in range(3)]` or `gameBoard = [['_' for _ in range(3)] for _ in range(3)]`
17	UnboundLocalError: local variable 'boardO' referenced before assignment	Add `global boardO, boardX`
23	Wrong `winCondition`	`winCondition = [7, 7 << 3, 7 << 6, 73, 73 << 1, 73 << 2, 273, 84]`

Scoring:

- 0 pt if the answer only indicates the right line number but gives wrong explanation and correction
- Line 17: Line No. and explanation: 0.5 pt; correction: 0.5 pt
- Line 5: Line No. and explanation: 1 pt; correction: 1 pt
- Line 21: Line No. and explanation: 1 pt; correction: 1 pt

Section 5: Bonus

(4 pts)

36. Monte Carlo Estimation

(2 pts * 2 = 4 pts)

Blank Number	Your Answer
A	`< 1` or `< 1.0` or `<= 1` or `<= 1.0`
B	`sum(pts_in)`

END OF THE FINAL EXAM