Grading Rubric

- Not strict on case

#	answer	points	scoring	notes

1	`c`
	`d`
	`e`
	`h`
	`2`
	`50`
	`3`
		`3.5`	-0.5 for incorrect, additional, or missing line
			-0.25 for each of last 3 lines if var name included w/ correct number (e.g. x = 2)	

2	`good night`
	`name`
	`ok`
	`done`
		`2.5`	-0.5 pt for each line except 2nd
			-1 if "name" line missing, 0.5 if "james"
			-0.25 lines 1 and 2 merged ("good night name"), -0.5 if 1 and 2 merged with var name ("good night james")
				Expect 2nd line to be common error (variable vs literal).

3a	`1`
	`5`
	`OUTPUT:`
	`3.9`
		`3`	-0.5 pt each for first 3 lines (-0.25 for each additional line, e.g. var update)
			1.5 pts for last line correct (-0.5 if OUTPUT and 3.9 on same line, so 1.5 total for "OUTPUT: 3.9")	

3b	`Finds difference between max and min of 3 numbers and prints them.`	`3`	2 pt for mentioning max/min 1 pt for mentioning finding *difference* between max and min ~~1 pt for mentioning printing~~	

3c	`Number is always printed. Output not printed when a, b, c all negative.`
	
	`OR when a,b,c all same number`
		`3`	1 pt for mentioning number always printed
			2 pt for saying output not printed when a, b, c all negative or same (-1pt if they just say nothing printed when output < 0, -1.5 pt if only mention when conditional false)
			
			1 pt if they say "something is always printed" w/o explanation
				Question is ambiguous, so may drop it

4a	`2`	`2`	2 pts for right answer (-1 pt if write "u" or "u =")	
4b	`Prints the number of even numbers stored`	`3`	3 points if summarize code	

Supplementary material for
Xie, Benjamin, Dastyni Loksa, Greg L. Nelson, Matthew J. Davidson, Dongsheng Dong, Harrison Kwik, Alex Hui Tan, Leanne Hwa, Min Li, and Andrew J. Ko. 2019. “A Theory of Instruction for Introductory Programming Skills.” Computer Science Education, January, 1–49. https://doi.org/10.1080/08993408.2019.1565235

	in variables		(-1 if say "number of variables where %2 == 0", -2 if only mentions conditionals or boolean)	
4c	When a,b, and c are all odd numbers.	2	-1 if don't mention even/off and instead say "when a,b, and c %2 ==0" -1.5 if only mentions when u equals 0 -1.5 if only mentions if statements (e.g. "when all if statements invalid")	

5a	if winner==follower:
	  follower = leader
	  leader = winner/follower
	  current_streak = 0
	  print("new leader")
	
	elif winner == leader:
	  current_streak = current_streak + 1
	  print("same leader")
	else:
	  print("unknown player")
		3	1.5 for if condition (-1 if swap wrong),
			1 for elif
			0.5 for else
			
			-0.25-0.5 for minor syntax errors (logic is correct, code may need small adjustment to run correctly).
			
			-0.5-1 for major syntax errors: unclear what code is trying to do, major refactor for code to work0	

5b	comments	3	1.5 pt for if condition explained 1 pt for elif 0.5 pt for else	
6a	cost_per_person = cost / num_people print(cost_per_person)	2	-1 for line of code incorrect	
6b	comments	1	-0.5 if line does not explain code	
6c	1. sum amt1, amt2, amt3 and store 2. Determine difference between amount paid and cost 3. If difference <0 and absolute diff > threshold, print underpaid and absolute difference. 4. Else if absolute difference is less than	5	0.5 point for summing amounts 0.5 point for getting difference between amt paid and cost 1 points for mentioning conditional w/ 3 options (0.25 for each mentioned + 0.25 if all 3 mentioned) 1 point for correct print statements for each account	

	threshold then print "paid in full" 5. Else: print overpaid and the absolute difference.			

6d	paid = amt1 + amt2 + amt3
	thres = 1e-6
	
	diff = paid-cost
	
	if diff<0 and abs(diff)>thres:
	  print("underpaid")
	  print(abs(diff))
	elif abs(diff)<thres:
	  print("paid in full")
	else:
	  print("overpaid")
	  print(abs(diff))
		4	0.5 point for total paid
			0.5 point for difference between cost and sum of amounts paid
			1 point for 3 conditionals (-0.5 if only used "if)
			1.5 point for float equality check in conditonals w/ threshold, abs value. (-0.5 for each incorrect w/ major error (logic, major syntax). -0.25 for each w/ minor syntax error)
			
			0.5 point for correct print statements (-0.25 if values not printed)	

6E	comments	2	-0.5 if a part of code not explained	
7A	1. Extract digits in inp (using %10 and / 10) and store each digit 2. Check if sum of first 3 digits % 7 == the last digit 3. Based on result of step 2, print "valid" is True or "NOT valid" if False	4	2 pts for digit extraction (-0.5-1.5 if vague) 1 point for each step afterwards mentioned. -0.5 for each ambiguous step	

7B	digit_4 = inp % 10
	inp = inp / 10
	digit_3 = inp % 10
	inp = inp / 10
	digit_2 = inp % 10
	inp = inp / 10
	digit_1 = inp %10
	
	sum_3 = digit_1 + digit_2 + digit_3
	
	if(sum_3 % 7 == digit_4):
	  print("valid")
	else:
	  print("NOT valid")
		5	2 pts for digit processing (-0.5-1 if inp not truncated properly, -0.5-1 if digits not saved properly)
			
			1 point for summing digits properly
			
			1 point for conditional % 7 (-0.5 if 2 ifs used)
			
			1 point for writing correct prints	

7C	comments	3	explain digit processing, explain sum digits,	

			explain conditional w/ mod	

See also: Google sheet: Points by skills

Supplementary material for
Xie, Benjamin, Dastyni Loksa, Greg L. Nelson, Matthew J. Davidson, Dongsheng Dong, Harrison Kwik, Alex Hui Tan, Leanne Hwa, Min Li, and Andrew J. Ko. 2019. “A Theory of Instruction for Introductory Programming Skills.” Computer Science Education, January, 1–49. https://doi.org/10.1080/08993408.2019.1565235