New submission for: Exam 2	Due: 11/10/2020 @ 11:00 PM EST
Honor Pledge On my honor, I have neither given nor received any unauthorized aid on this exam. By typing your first and last name in the space provided below you are electronically signing to indicate that:	
By typing your first and last name in the space provided below you are electronically signing to indicate that: 1. You are the person who is taking this exam. 2. You read and understood the Honor Pledge and you agree to be bound by it. 3. You will submit your responses no later than two hours from the time you first accessed the contents of this exam,	
If your first and last name does not appear below, your exam will not be graded and you will receive a grade of zero.	
Write your answer below: Xinhao Luo	
Question 1. (Scoping, Python, 15pts) Consider the following Python 3 program: i = 2	
<pre>j = 3 def A(): print (i)</pre>	
<pre>def B(P): i = 3 def C(k): def D(): i = 5 P()</pre>	
<pre># body of C D() print (i,j,k)</pre>	
<pre># body of B C(j) # body of main B(A)</pre>	
print (i, j) (i) What does this program print? Note: recall that in Python a variable that is written is assumed to be local, unless it is explicitly imported with the keyword global.	
Write your answer below: Press TAB to indent. Press ESC to advance from answer area. 2 3, 3, 3 2, 3	
Clear Use Most Recent Submission (ii) What would the program print if Python used dynamic scoping with shallow binding?	
Write your answer below: Press TAB to indent. Press ESC to advance from answer area. 5 3, 3, 3 2, 3	
Clear Use Most Recent Submission	
(iii) What would the program print if Python used dynamic scoping with deep binding? Write your answer below: Press TAB to indent. Press ESC to advance from answer area.	
3, 3, 3 2, 3 Clear Use Most Recent Submission	
Question 2. (Scoping, Scheme, 10pts) Now consider the following Scheme program: (define B (lambda () (let* ((y 5)))	
(let* ((G (let* ((y 2)) (lambda () y))))	
Write your answer below: Press TAB to indent. Press ESC to advance from answer area. (2 5)	
(ii) What would the program print if Scheme used dynamic scoping with shallow binding? Write your answer below:	
Press TAB to indent. Press ESC to advance from answer area. (2 5) Clear Use Most Recent Submission	
Question 3. (Lambda calculus, 12 pts) Consider the following lambda expression $\lambda x.(\lambda y.y\ y)\ (\lambda y.y\ y)\ y\ x.$ (i) List all free variables in this expression. Write your answer below:	
Press TAB to indent. Press ESC to advance from answer area. {y} Clear Use Most Recent Submission	
 (ii) List all redexes in this expression. Show each redex on a separate line. Note: type λx. as \x. in the Submitty text box. For example, (λy.y) z will turn into (\y.y) z. Write your answer below: 	
Press TAB to indent. Press ESC to advance from answer area. (\y.y y) (\y.y y)	
Use Most Recent Submission (iii) List all normal forms that this expression is in. Write NEITHER if the expression is in neither of the normal forms. Write your answer below:	
Press TAB to indent. Press ESC to advance from answer area. WHNF Clear Use Most Recent Submission	
Question 4. (Lambda calculus, 20 pts) Evaluate the lambda expression $(\lambda x.\lambda y.\ x\ y\ (\lambda x.\lambda y.\ y))\ (\lambda x.\lambda y.\ x)\ (\lambda x.\lambda y.\ y)$. Note: type $\lambda x.$ as $\xspace x$. in Submitty. For example, $(\lambda y.y)\ z$ will turn into $(\yspace y.y)\ z$. (i) Using applicative order reduction. Show each intermediate form on a new line.	
Write your answer below: Press TAB to indent. Press ESC to advance from answer area. $= (\y_1. \ (\x.\y. \ y) \ (\x.\y. \ y) \ (\x.\y. \ y)$ $= (\y_1. \ (\y. \y_1) \ (\x.\y. \ y) \ (\x.\y. \ y)$ $= (\y_1. \ y_1) \ (\x.\y. \ y)$	
$= (\langle x. \rangle y. y)$	
Close Meet Record Submission	
Clear Use Most Recent Submission (ii) Using normal order reduction. Show each intermediate form on a new line. Write your answer below:	
Press TAB to indent. Press ESC to advance from answer area. = (\y. (\x.\y. x) y (\x.\y. y) (\x.\y. y) = ((\x.\y. x) (\x.\y. y) (\x.\y. y)) = ((\y. (\x.\y. y)) (\x.\y. y) = (\x.\y. y)	
Clear Use Most Recent Submission	
Question 5. (Attribute grammar over the Lambda calculus, 13 pts). Recall the context-free grammar that defines the syntax of the Lambda calculus:	
Write an attribute grammar that associates a boolean attribute nf with each E , such that nf is true if and only if E is in normal form. Write your answer below: Press TAB to indent. Press ESC to advance from answer area.	
E -> x: E.abs = false E.nf = true E -> (\x. E_1): E.nf = E_1.nf	
<pre>E.abs = true E -> (E_1 E_2):</pre>	
E.nf = E_1.nf and E_2.nf	
Clear Use Most Recent Submission Question 6. (Scheme Programming, 20 pts). Consider the following Scheme function.	
<pre>(define (fun lis) (cond ((null? lis) lis)</pre>	
(i) Give two examples of usage of fun showing the input and output as you did in your Scheme homework. Trivial inputs do not count. Write your answer below:	
Press TAB to indent. Press ESC to advance from answer area. (fun '(1 3 3 4 1)) should produce (1 3 4 1) (fun '(1 1 3 3 4)) should produce (1 3 4)	
Clear Use Most Recent Submission	
(ii) What does fun do? Write your answer below: Press TAB to indent. Press ESC to advance from answer area.	
When encountering consecutive duplicate elements in the list, keeping only one of them.	
Clear Use Most Recent Submission	
(iii) Is fun tail-recursive? If your answer is YES, explain why. If your answer is NO, write a tail-recursive version of fun. Write your answer below: Press TAB to indent. Press ESC to advance from answer area. No, and the tail-recursive version shown below. The main entry is `fun`	
<pre>(define (fun_helper lis tlis) (cond ((null? lis) tlis)</pre>	
<pre>(define (fun lis) (reverse (fun_helper lis '())))</pre>	
Clear Use Most Recent Submission Question 7. (Scheme Programming, 25 pts). We will write a Scheme function that finds the depth of the minimally	
nested atom in an arbitrarily nested list. Example calls are as follows: • (min-nest '(1 ((2) (3)))) yields 1 • (min-nest '((((((8)) 9)) a))) yields 2 You may assume that the input is a non-empty list, the maximal depth of the input list is 100, and the null list () does	
not appear as an element of any of the lists. There is no restriction on the built-in functions that you can use, e.g., min can be a useful function. In addition to built-in functions you can use predicate atom? as defined in class. Your answer should be a single function.	
Write a function min-nest using (deep or shallow) implicit recursion. For full credit, you should use map and fold (either fold or foldr as defined in class). Write your Scheme below: Press TAB to indent. Press ESC to advance from answer area. (define (min-nest lis)	
Clear Use Most Recent Submission Additional Notes Press TAB to indent. Press ESC to advance from answer area.	
Clear Use Most Recent Submission	
By clicking "Submit" you are confirming that you have read, understand, and agree to follow the Academic Integrity Policy. Submit	
Select Submission Version: Version #6 GRADE THIS VERSION Do Not Grade This Assignment Note: This version of your assignment will be graded by the instructor/TAs and the score recorded in the gradebook.	
Note: This version of your assignment will be graded by the instructor/TAs and the score recorded in the gradebook.	
111/115 Total	
Autograding Total (Without Hidden Points)	
Autograding Total (Without Hidden Points) Autograding Total (With Hidden Points)	
Test 1 Honor Pledge signature submission Test 2 Additional Notes submission Test 3 Check Honor Pledge	Show Details Show Details Show Details
Test 3 Check Honor Pledge Test 4 Check Time Limit HIDDEN: Test 5 Curve	Show Details Show Details
111/115 TA / Instructor Grading Total 15/15 Problem 1 (Graded by: Haussmann)	
 ✓ 0 Full Credit 6/10 Problem 2 (Graded by: Chastain) ✓ -4 (ii) Incorrect numbers. 	
12 / 12 Problem 3 (Graded by: Chawla) ☑ 0 Full Credit	
20 / 20 Problem 4 (Graded by: Chawla) ☑ 0 Full Credit 13 / 13 Problem 5 (Graded by: Chastain)	
20 / 20 Problem 6 (Graded by: Paul) ☑ 0 Full Credit	
Problem 7 (Graded by: Mankowski) ☑ 0 Full Credit	