Please, verify as soon as possible that you can display images properly, by loading the provided code and running the Test.test_display function.

**Problem 1** [10 points] Exercise 4.8 (pg. 85) of Mitchell

**Problem 2** [10 points] Exercise 4.9 (pg. 85) of Mitchell

**Problem 3** [15 points] Exercise 4.10 (pg. 85) of Mitchell

**Problem 4** [20 points] Given the semantics for our picture language prove the following equalities about shapes

(a) [10 points]
\[ S[scaleXY(a, b, halfplane((x_0, y_0), (x_1, y_1)))] \cong S[halfplane((ax_0, by_0), (ax_1, by_1))] \]

(b) [10 points]
\[ S[translate(p_0, halfplane(p_1, p_2))] \cong S[halfplane(p_1 + p_0, p_2 + p_0)] \]

**Problem 5** [25 points] This problem asks you to complete the implementation of shape.sml

(a) [5 points] Complete the cases of the deno function that are related to the primitive shapes Elipse and HalfPlane. After completing the cases you should be able to run the tests Test.test_circle and Test.test_half_plane.

(b) [10 points] Complete the cases of the deno function that are related to the primitive shapes Intersect, Union, and Diff. After completing the cases you should be able to run the tests Test.test_hole and Test.test_semi_circle.

(c) [10 points] Complete the cases of the deno function that implement the cases related to Trans and ScaleXY. After completing the cases you should be able to run the tests Test.test_letter_e and Test.test_hw.

**Problem 6** [20 points] The function remove_trans recursively rewrites a shape definition into a shape definition which does not contain any translation operations. It uses the equations about scaling and translation in Section 3.2 of our semantics of shapes.

Complete the function remove_scale which preforms a similar operation for scaling operations. After completing the function you should be able to run the test Test.test_opt_hw.
Extra Credit  [Negotiable points] Help flesh out the function PictureLib.charToGlyph to increase the number messages we can write. Please, contact danwang@cs.princeton.edu so we can coordinate the addition of new characters to our font. Alternatively consider adding new textures or just drawing a pretty picture using the existing primitives.