[19sp]:

* (Placeholder for todos added from Slack)

[18fa]:

* (jupierce): add preview for points to the rubric
* (jennche): just a thought, but maybe for rubric include an item for properly adding the change listener for thickness to checkbox and/or properly encapsulating the state of the checkbox/slider. There are (and I know I didn't do this right either when I took the class) ways that you can get it to work without recognizing the value/parameter in the change listener for thickness represents the boolean held by the checkbox. For example, creating another ref in paint.ml that holds the boolean state of whether or not the checkbox is toggled.
* (jennche): MG1: Layout rubric should include points for properly using hlist and vlist for the mode/color/paintwidget
* Change “set\_value” to “change\_value” in test descriptions in widgetTest.ml
* Change comment for undo button to acknowledge that it needs to be modified in task 2 (as well as tasks 3 and 4)
* Include FAQ link somewhere in the HW description
* John: Clarify that checkbox should respond to clicks in the entire area of the widget (It's a common bug where a checkbox composed of smaller widgets results in only some of those smaller widgets responding to clicks, which may be intuitive visually but which does not work with our test case, which explicitly checks for a click at the point (5, 5))
* Daria: Lightbulb defined as canvas was confusing for students
* (jennche): We say `(\* TODO: You will modify this definition in Tasks 2, 4, 5 and maybe 6. \*)` right above type shape. But for task 3 (points), they will need to modify shape.

-- various semesters --

**HW5** -

* MC (17fa): Fix widgetTest! We don’t test vpair/hpair is laid out correctly! People can have bugs that pass the given tests. Example: vpair’s repaint = w1.repaint; Graphics.translate g (snd (0, w1.size())); w2.repaint g;
* ~~BH: forgetting to take away 2-point lines + ellipses is currently a -8. Make it worth fewer points~~

**TODO:**

* **Task 4**
  + Just tell them the ellipse math?
* **Task 5**
  + Tell them NOT to comment out checkbox code bc they need it there for their code to compile
* **Task 6**
  + If the slider is for colors, it is ok if pressing a color button does not change the slider state, but the slider should change the color preview/current color

**TODO from Fall 2017 (not sure if these got considered):**

* (JC) make sure it’s clear only need to make one new cool widget
* ~~(DM) make sure students know the ellipse "bounding box" doesn't actually need to be drawn on the screen~~
* ~~(MC) consider changing the name of `set\_value` to `change\_value` to even more explicitly highlight the fact that setting/changing the value runs all change listeners?~~
* (PP) add type definition for change\_listener?
  + (MM) maybe just move value\_controllers in general to their own file, and force the students to use them in ways that are unrelated to widgets?
  + (MC) I see the merit it demonstrating different uses of the 'a value\_controller, but we want the students to recognize this versatility organically on their own when they do task 6. It's a conceptual leap, but an important one for them to figure out for themselves as opposed to us (once again) telling them the answer (the "answer" here being: use a value\_controller for widgets with state that impact other parts of the project)
* ~~(PP) add autocomplete for PennKeys to guigrade.sh~~
* RUBRIC stuff
  + ~~(PP) The rubric for Task 6 is incredibly vague.~~
  + ~~(ES) may want to make the rubric for task 6 a little more general so people who spend 5+ hrs on radio buttons don’t get 0 points (maybe make 5 points of task 6 be awarded for actually writing a widget and then 15 pts for the rest — this seems fair since we award people 5 points if they just write code for a new shape)~~

TODO Spring 2018:

* ~~(JP) Consider removing the quit button as it doesn't work in codio~~
* (JX) Make sure the headers of all problems contain the task number they belong to
  + Eg, a few people were confused about editing it for task 2 since the header of the function didn’t mention task 2.
  + **(DH) No examples of this found as of Fall 2018, but keep this open in case one is found in the future**
* ~~? (MR) hw5 actually define change\_listener as a type ?~~ 
  + Not implementing currently, because a function is not a change\_listener until it is called by add\_change\_listener
* ~~? (PP) hw5: consider making label controller its own thing so that the repaint and size functions don’t need to directly access the internal state of the label text ?~~
  + ~~Enforces MVC paradigm which we tell them to use and makes the purpose of the value controller in checkbox easier to see~~
  + ~~The idea is that you wouldn’t access the contents of the ref directly (because this should belong to the label controller since it is the model) in your view~~
  + ~~You would be routing the access through the controller~~
* (PP) New idea for kudos-- rewrite hlist and vlist more efficiently so that the depths from the hlist handler to the actual widgets being grouped is roughly log n.
  + If you have an hlist of several thousand elements, clicking on the last one in the list requires the event to be routed through the handlers of all of the elements in the list.
  + Or better yet, let the hlist or vlist handler delegate directly to the correct widget instead of using nested hpairs and vpairs.
* ~~(JC) \*\*\*\*\*\*\* do this \*\*\*\*\* : consider adding to instructions a warning to not do `Graphics.set\_line\_width 0`~~
  + ALSO add this to the FAQ
  + This is weird bc the documentation says it should be fine but it's just in the browser that it doesn't work, it's a bug with js\_of\_ocaml
* ~~(MR) hw5 change gctx.mli to not expose the type definition of event~~
* ~~(PP) hw5 `lightbulb.html`: “This application will not work until you have completed Task 5.” should clarify that we mean the checkbox portion of Task 5~~
* (PP) Use records instead of tuples for Mode data, eg:  
  