

Welcome, graders!

CS 5 fall 2020, hw 2 guidelines...

- Open the [submission site](#) - and the [assignment](#)
- Grading nourishment -- *KitKats!*

[GOLD and BLACK grading]

Reminder: *Please remember -- do NOT change the grading rubrics themselves...
Instead, click the best-fitting rubric, usually less than the total points,
then make adjustments in GradeScope's [submission-specific](#) section*

[STEP 1 - readings]

- Grading reading responses: **5 pts: good!** **4pts: unconsidered** **<=3pts: what?!**
- The article: [Watson wins on Jeopardy!](#)
- The prompts (*OK if they addressed only one, no harm if they touched on them all*):
 1. What was the most interesting or important idea in this article to you—and why?
 2. What is an application that (in your opinion) Watson's technology might be able to contribute to? Alternatively, do you feel Watson's capabilities will not make much of an impact? You can look around for web articles on this, or you're welcome to speculate.
 3. Whether or not you had the chance to interact with the "Watson-like" Jeopardy game, comment on your sense of the similarities and/or differences between Watson-style and human-style thinking.
- Here is a [page of guidelines and possible comments for reading responses](#).

Possible comment theme for week #2: Recursive Jokes!

A recursive bar joke

A guy walks into a bar and tells the Recursive Bar Joke, only he does it *much* better than I do.

Note: this joke gets funnier every time!

A recursive light bulb joke

Q: How many turns does it take to replace a light bulb?

A: Is it already working? Then zero. If not, then twist it once, ask me again, and add 1 to my answer.

[STEP 2 - lab and PythonBat problems] hw2pr1 + hw2pr4

- this week we will create a 2-3 person **"lab team"** to handle the remaining labs
- thank you, lab team! (let's cover both gold and black: they're the same in hw2)

-
- and, we'll create a 2-3 person **"PythonBat team"** to grade hw2pr4
 - + regular credit: 2 pts each for the first 25
 - + **extra credit** beyond 25 [[this sheet has totals](#)] according to these conditionals:

```
if number_solved_for_ex_cr >= 51: (or more)
    points = +21 extra credit
elif number_solved_for_ex_cr > 45: (that is, 20-25 inclusive)
    points = +15 extra credit
else:
    points = (number_solved_for_ex_cr / 2) # let's do floating point division, since GradeScope can handle it
```

[STEP 3: run hw2pr2 (for correctness + style + extra-credit)...]

Gold

- + **look** at the autograded-top (don't run) for **rwpos** (5 pts)
- + **look** at the autograded-top (don't run) for **rwsteps** (10 pts)
- + final 20 points: either download + run OR [paste into repl.it to run](#)
- + 20 regular-credit points
 - 10 points for two independent wanderers
 - 5 points for a working simulation
 - 2 points for the docstring at all
 - 3 points for a helpful docstring explanation (adjust as appropriate, per-case)
- + up to +5 (or more) extra credit points: your discretion ~

Optional Extras ...

In addition, for up to +5 (or more) extra-credit points, feel free to customize things more thoroughly:

- randomly-appearing obstacles?
- the wandering need not be uniform
- inner walls? teleporting? wells?
- power-ups? (perhaps the same thing)
- adding human input (at the beginning or end - not during the simulation)
- see [this page](#) for adding different colors to your wanderers
- or something farther afield...

- + *Please share a phrase or sentence on something you appreciated about their submission (if it did work + did have a context!)*

[STEP 4 - problems to look at (for style + extra-credit)...]

- Please *do* look over these problems' code for style & possible extra features...
(This week's commenting theme: **Recursive jokes**: see above!)
- Here are the problems; below are problem-by-problem specifics:
 - **gold hw2pr4** (extra-recursive graphics)
 - **black hw2pr3** (sorting and analysis)
 - **black hw2pr4** (extra-recursive graphics)

Problem-by-problem grading guidelines

Gold hw2pr3 [Integration ~ autograded for 20pts](#) 10pts manually:

Look over for docstrings

Look at the bottom: did they answer the final two questions reasonably?

- + **Question 1:** it will always underestimate because the function is **increasing/ascending**
 - + (and - optionally - it's a left-anchored sum) **[5pts]**
- + **Question 2:** that integral converges to π (3.14159...) because it's $\frac{1}{4}$ of a circle of radius 2, whose area is 4π (or half a semicircle whose area is 2π) **[5pts]**

No need for a personalized comment here (always welcome, but this is less individualized...)

Gold hw2pr4 [Turtle Art extra-credit \(up to +5 or more pts\)](#)

Look over the screenshots submitted...

Here is a [Google slidedeck](#) to paste the highlights + please create textboxes w/names!!

- **Style - don't worry about**

- **Extra-credit points**

- Is it a creative variation of one of the turtle-graphics problems? **+2-5 pts**
 - a single variation on one of the turtle functions alone might be **+2 or +3 points**
 - **+5pts** for a colorful tree with varying widths (or changing any **two** characteristics of the drawing)
 - even more extra-credit is possible for extra-crazy implementations!
 - let me know of the logins of any **really great ones** worth sharing!
- Is it something different and above-and-beyond in creativity or effort? **+8+pts**

Black hw2pr2, hw2pr3, hw2pr4 [these are largely autograded.... plus](#)

- **Style points**
- - take off **-2** points for any function with **no docstring**, only **up to -5 pts overall**
- In 42+ me, be sure they have top-of-file results aligning
ran = "GTACGTCGATAACTG" vs reference = "TGATCGTCATAACGT"
- In RNA folding look for
 - the 1st extra credit ("getFold") - test [these cases](#) by hand, e.g., in repl.it or by download (+10)
 - the 2nd extra credit "drawFold" (up to +10 points...) try by hand!
 - perhaps have s/o be an "extra-credit expert" on this problem (with a ready set-up)

If time: **Screenshots!** (now 1: w/images 2: no images 3: no annotation!)

How to get to this page:

The CS5 site should be top link when Googling

hmc cs5

These notes are linked at the bottom of CS5's page as

for grutors - guidelines

Also, the submissions site is at

cs.hmc.edu/submit

The hw we're grading:

<https://www.cs.hmc.edu/twiki/bin/view/CS5/Homework2Gold>

Thank you, CS5 graders!!!

perhaps not needed in 2020?

GREEN graders: [editable guidelines](#) [read-only full-course guidelines](#)

Gold hw2pr2 Sleepwalking and analysis (55 pts)

Download this file and run it...

There _may_ have been_ an "autograded" run -- feel free to see how well it worked, e.g.,

```
rwpos( 5, 4 ) # should start at 5 and run 4 steps...  
rwsteps( 5, 0, 10 ) # should start at 5 and run until it hits 0 or 10
```

If it doesn't work at all -- **take off 15-35 points** (look over the code to get a sense of effort)
-- and invite the student to resubmit / come to grutoring hours

If it works, then check out ***their own*** variation

- + does it work? [if not, -5 to -10 pts]
- + is it at least a little different than rwsteps? [if not, -2 to -5 pts]
- + is it awesome? See extra-credit below:

- Extra-credit points

- upto **+5 or more pts** for unusual ASCII animations or features!
 - an unusual "sleepwalker" alone might be **1 or 2 points**
 - multiple sleepwalkers or state-changing ones would be **+5pts**
 - even more is possible for extra-crazy implementations!
 - **send me an email/add to the google doc/let me know** if any **really great ones** worth sharing!

Sleepwalker analysis: this is now in the "Loops" module

- + Not part of summer 2020's recursion module...

Correctness points - you'll need to run two tests:

- **only if the above tests did not run...**
 - Copy/load their code into repl.it and run `rwpos(40, 10)` and `rwsteps(11, 1, 21)`
 - The first one prints 10 (or 11) positions, starting from 40
 - The second one should animate a walker starting at location 11, wandering between location 1 and 21 until they get to a wall...
- **-7pts** if there's no `rwpos` (make sure they *did* use return...; else -2)
- **-7pts** if there's no `rwsteps` (do make sure there *is* a return value...; else -2)
They didn't have to get to the wall -- *next to* the wall is OK.
- **-7pts** if they didn't do anything on the **signed-displacement q'n**
 - 5 if they didn't have an `ave_signed_displacement(numtrials)` function
 - 3 if they didn't have any comment on it (should have been averaging to 0)
- **-5pts** if they didn't do anything on the **squared-displacement q'n** somehow
 - 5 if they didn't have an `ave_squared_displacement(numtrials)` function
 - 3 if they didn't have any comment on it (should have been averaging to N)

- Style points

- take off **-2** points for any function with **no docstring**, only **up to -5 pts overall**
- take off **-3** points for any misnamed function (capitalization etc.)
- take off **-2** points for too little spacing (no space between functions, for example)

- Overall comment to the student...

- Please be encouraging! Copy-and-paste for efficiency (this week: **recursive jokes**: see below!)
- Be sure to briefly mention any style points missed and correctness problems
 - Docstrings: **-2** each, up to **-4** or **-5** at most...

Black hw3pr2 Longest Common Substring (30 pts)

- **Correctness points**

- should be autograded
- please return some of the points (50% or more), if a small error caused a set of cascading problems

- **Check for an alignment of these two strings in their file:**

-7pts if it's not there **-3pts** if it has noticeably less overlap than these:

- `EGS = "GTACGTCGATAACTG" # string1`
- `WGS = "TGATCGTCATAACGT" # string2`
- *The alignment should look something like this - if it's reasonably close, that's fine:*
- `#'GT-A-CGTCGATAACTG- '`
- `#'-TGATCGTC-ATAAC-GT '`

- **Style points** same as in Gold hw2pr2