

CS 100 - Homework 2

Deadline:

Due by 11:59 pm on Friday, September 7, 2012

How to submit:

- Go to the course Moodle site.
 - (either log into myHumboldt and click the Moodle button in the upper right corner,
 - or get to it via learn.humboldt.edu,
 - ...and then click the "my moodle" button in the upper right under the big yellow moodle logo to get your list Moodle course sites,
 - or follow the link from the public course web site, users.humboldt.edu/smtuttle/f12cs100)
- Once you are at the course Moodle site, find the section titled "Submit your HOMEWORK FILES here",
 - and click on the link "Click HERE to submit Homework 2".
- You should see, in the middle of the page, a place where you can upload your homework file.
 - Note that I believe I set this up to permit multiple submissions. I couldn't set it to "unlimited", because the largest number available was 20, but that seems like it should be ample.
 - Also note that I will grade the latest file submitted before the deadline unless you e-mail and tell me otherwise.

Purpose:

To practice some of the computer-related ideas discussed in Week 2 of class (working up to Chapter 2's concepts of argument, statement, premise, and conclusion).

Important notes:

- You do not have to run or execute any of these answers in DrRacket, Python, or Prolog (although you may if you wish).
- Type:
 - your name, then
 - the problem number and your answer for each of the following questions,
 - into a file named `cs100hw2` (optionally followed by your name, including only letters and underscores -- NO blanks or other unusual characters)
 - Its format may be `.txt`, `.odt`, `.pdf`, `.doc`, `.docx`, or `.jpg`. (Please ask me before

submitting files of other formats.)

- It is possible that your answers may be compiled and posted to the course Moodle site.

The Problems:

Problem 1:

Decide whether you would like to answer the questions below in terms of either Racket or Python. Write the name of your choice, then your answers for each of the following (each preceded by their part number).

1 part a:

Write an expression that would be an expression of type **number** in that language.

1 part b:

Write an expression that would be an expression of type **string** in that language.

1 part c:

Write an expression that would be an expression of type **boolean** in that language.

1 part d:

Write a boolean expression that includes the boolean `and` operator, and follow that expression with its result (the value that expression should have when evaluated).

1 part e:

Write a boolean expression that includes the boolean `or` operator, and follow that expression with its result (the value that expression should have when evaluated).

1 part f:

Write a boolean expression that includes the boolean `not` operator as well as either the boolean `and` or `or` operator, and follow that expression with its result (the value that expression should have when evaluated).

Problem 2:

Consider the Prolog predicates and facts discussed in class. Answer each of the following questions (each preceded by their part number).

2 part a:

Write an example of a fact of your choice, using Prolog syntax, as could appear in a Prolog knowledge base.

2 part b:

Write a second, different fact of your choice, using Prolog syntax, as could appear in a Prolog knowledge base.

2 part c:

Write a different fact of your choice, using Prolog syntax, that uses the same predicate as you used in either your part a or part b answer (or both, if you already happened to use the same predicate in parts a and b).

2 part d:

Write a Prolog query, as you could type into a Prolog interpreter, that includes a Prolog variable, and that would be able to be proven to be true if your answers to parts a through c were part of the knowledge base currently loaded into Prolog.