

CSCI H200 INTRODUCTION TO COMPUTERS AND PROGRAMMING

FALL 2019 GRADE REPORT

Anderson, Brady

Computer Science
School of Informatics, Computing, and Engineering

Indiana University, Bloomington, IN, USA

December 16, 2019

Assignment 1

Assigned: September 4, 2019

Due: September 11, 2019

Problem 1

windchill.py

50 points total

10/10 points for correct Assignment1 folder setup

10/10 points for correct module name

10/10 points for proper variable names in the calculation (T and V)

20/20 points for proper calculation

Perfect!

Score: 50/50

Problem 2

creditcard.py

50 points total

10/10 points for correct Assignment1 folder setup

10/10 points for correct module name

10/10 points for proper variable names in the calculation (APR, C, P, i)

20/20 points for proper calculation

Score: 50/50

Total Score: 100/100

Assignment 2

Assigned: September 12, 2019

Due: September 18, 2019

Problem 1

mayhem.py

195 points total

120/120 points for functions [10 points each]:

`speed`, `distance`, `time`, `hours_to_min`, `min_to_sec`, `feet_to_mile`, `miles_to_kilometers`, `kilometers_to_miles`, `miles_to_feet`, `degrees_to_radians`, `parsecs_to_kilometers`, and `lightyears_to_parsecs`.

75/75 points for functions [15 points each]:

`side_length_triangle`, `celsius_to_fahrenheit`, `fahrenheit_to_celsius`, `kelvin_to_fahrenheit`, and `percent_change`.

Perfect!

Score: 195/195

Problem 2

2019tax.py

60 points total

25/25 points for proper implementation of the `unmarriedTax` function.

25/25 points for proper implementation of the `marriedTax` function.

10/10 points for answering observational question.

Great!

Score: 60/60

Problem 3

lestat.py

80 points total

40/40 points for implementation of the `receiveFrom` function with correct output.

40/40 points for implementation of the `donateTo` function with correct output.

Excellent!

Score: 80/80

Problem 4

coolline.py

35 points total

10/10 points for changing the title of the graph.

25/25 points for adding the new function to the plot.

Perfect !

Score: 35/35

Total Score: 370/370

Assignment 3

Assigned: September 19, 2019

Due: September 25, 2019

Problem 1

funwithfunctions.py

135 points total

135/135 points for functions [15 points each]:

Perfect!

Score: 135/135

Problem 3

qc1.py

50 points total

0/15 points for printing a message indicating `complex` or `not complex`.

0/35 points for a correct implementation of the `q` function with appropriate return structure for quadratic solutions.

File does not run because of indendation error at the bottom. q should return a tuple of roots, not empty string. Code for graph is commented out.

Score: 30/50

Problem 4

if.py

75 points total

75/75 points for conditional statements correctly re-written [15 points for each group]:

Great!

Score: 75/75

Problem 5

precmetal.py

75 points total

30/30 points for proper implementation of the `preciousMetalToDollars` function.

45/45 points for proper implementation of the `purchase` function.

Perfect!

Score: 75/75

Problem 6

myclock.py

25 points total

10/10 points for changing title.

15/15 points for changing font.

Interesting name

Score: 25/25

Total Score: 340/360

Assignment 5

Assigned: September 25, 2019

Due: October 2, 2019

Problem 1

funtriangle.py

45 points total

45/45 points for correct triangle output [15 points each]:

nice !

Score: 45/45

Problem 2

makeitrain.py

40 points total

30/30 points for correct implementation of `dollars` function.

5/10 point for appropriate return values and structure.

output off slightly check how pennies are counted!

Score: 35/40

Problem 3

donor.py

60 points total

30/30 points for proper implementation of `red.blood_compatibility` function with appropriate return values.

30/30 points for proper implementation of `transfusion` function with appropriate return values.

great !

Score: 60/60

Problem 4

palindrome.py

40 points total

40/40 points for correct implementation of `palindrome` function.

nice !

Score: 40/40

Problem 5

roman.py

50 points total

50/50 points for correct implementation of `roman` function.

nice !

Score: 50/50

Problem 6

moreloops.py

140 points total

75/75 points for correct implementation [15 points each] of
`maxFor`, `maxWhile`, `minFor`, `myReplace` functions, `StringConcat`

40/40 points for correct implementation [20 points each] of
`RemoveEvens`, `sumOdd`

Score: 115/115

Problem 7

farm.py

50 points total

0/50 points for correct implementation of `roman` function.

great !

Score: 50/50

Total Score: 395/400

Assignment 5

Assigned: October 3, 2019

Due: October 9, 2019

Problem 1

entropy.py

60 points total

30/30 points for correct implementation of the `makeProbability` function:

30/30 points for correct implementation of the `entropy` function:

great job!

Score: 60/60

Problem 2

magic.py

60 points total

40/40 points for correct encantation [8 points each]

10/10 points for correct order of encantation

10/10 points for correct return value

nice!

Score: 60/60

Problem 3

ones.py

40 points total

40/40 points for correct implementation of the `lr` function

nice!

Score: 40/40

Problem 4

nines.py

40 points total

40/40 points for correct implementation of the `div_9` function

good !

Score: 40/40

Problem 5

squares.py

40 points total

40/40 points for correct implementation of the `sq` function

nice !

Score: 40/40

Problem 6

luddy.py

70 points total

15/15 points for correct implementation of the `area` function

15/15 points for correct implementation of the `f` function

20/20 points for brute force solution

20/20 points for numpy solution

good work !

Score: 70/70

Problem 7

wish.py

50 points total

50/50 points for correct implementation of the `is_magic` function

nice job !

Score: 50/50

Total Score: 360/360

Assignment 6

Assigned: October 10, 2019

Due: October 14, 2019

Problem 1

alpha.py

80 points total

10/10 points for correctly opening and reading the file from the correct location

20/20 points for correctly reading the file contents

10/10 points for setting up the dictionary

20/30 points for counting lowercase letters

10/10 points for properly returning the dictionary of counted letters

This counts anything if the ASCII code is above 60. It is supposed to count only lowercase letters

Score: 70/80

Total Score: 70/80

Assignment 7

Assigned: October 24, 2019

Due: October 30, 2019

Problem 1

recpractice.py

175 points total

150/150 points for correct implementation of the ten recursive functions [15 points each]

25/25 points for including a for-loop to show the first ten values of each recursive function

15/15 points for answering critical thinking questions

s1 should say $n+1$ instead of $n-1$

Score: 190/190

Problem 2

minime.py

90 points total

90/90 points for correct implementation of the six min functions [15 points each]

Great

Score: 90/90

Problem 3

twoMax.py

35 points total

0/35 points for proper implementation of **twoMax** function

Supposed to return a list of the greatest of each pair, not the greatest in the whole 2d list

Score: 30/35

Problem 4

isogram.py

30 points total

30/30 points for correct implementation of `is_isogram` function

Nice

Score: 30/30

Problem 5

hexagram.py

35 points total

35/35 points for correct implementation of `hex_dec` function

Sweet

Score: 35/35

Problem 6

doctor.py

50 points total

30/30 points for correct implementation of appendicitis prediction logic

10/10 points for correct input functionality

10/10 points for correct and informative output

10/10 points for meaningful and informative comments

Score: 60/60

Problem 7

astronomy.py and stellar.py

50 points total

20/20 points for completing the `astronomy.py` module

30/30 points for completing the functions in the `stellar.py` module

Incredible

Score: 50/50

Problem 8

pecan.py

35 points total

35/35 points for correct implementation of `myip` function

Beautiful

Score: 35/35

Total Score: 485/490

Assignment 8

Assigned: November 1, 2019

Due: November 6, 2019

Problem 1

fignewton.py

50 points total

15/20 points for correct implementation and integration of user input: function and initial estimate [10 points each].

30/30 points for correct implementation and integration of user input: threshold and iteration bound [15 points each].

*Supposed to let me input the function and use eval for that.
Otherwise correct*

Score: 45/50

Problem 2

mybisection.py

50 points total

15/15 points for correct implementation of the **sign** function.

35/35 points for correct implementation of the **bisect** function.

Nice, +20 ec

Score: 70/50

Problem 3

game1.py

50 points total

45/50 points for proper implementation of color changing square.

You're changing the background color for a frame, not the square

Score: 45/50

Problem 4

secant.py

50 points total

50/50 points for correct implementation of `secant` function.

Great

Score: 50/50

Problem 5

easycalc.py

50 points total

50/50 points for correct implementation of `simpson` function.

Sweet

Score: 50/50

Problem 6

rec.py

200 points total

0/200 points for correct implementation of `even`, `odd`, `b`, `btr`, `bm`, `gg`, `gtr`, `gm` functions [25 points each].

File doesn't run but functions look good

Score: 190/200

Total Score: 450/450