C200 Programming Assignment № 1

Functions & Learning About Homework
Fall 2023

Dr. M.M. Dalkilic
Computer Science
School of Informatics, Computing, and Engineering

Indiana University, Bloomington, IN, USA

September 1, 2023

The purpose of this homework is two-fold: (1) you'll develop your skills in implementing functions, formal and actual parameters, environment and scope (2) how homework will be delivered and submitted. Although the problems can be used for individuals, this course uses paired-programming. Please carefully read about pair-programming below.

Instructions

The course uses an online system for grading the Python code. The system can be accessed at `c200.luddy.indiana.edu`, and you should login via your official IU username and password. Upon login, you can submit HW via 'Create Submission' button. Remember that you **must** push the HW files to GitHub and also submit to the Autograder before the HW deadline.

Paired-Programming

Homework is generally done with a randomly selected partner. In the last section, some information about contacting and communicating with this likely new student will be facilitated. **As always, all the work will be with only you and your partner; but *both* of you should contribute.** If your partner does not respond, you must complete the homework on your own. You will not be assigned a new partner–this is infeasible for such a large class. Both students must submit their individual homework, since you are graded individually. Your grade may be different from your partner's grade, since there can be differences in code.

You must complete this before **Friday, September 8, 2023 11:00PM EST**. You will submit your work by pushing your code to your GitHub repository and by uploading your completed python file to the Autograder. Please remember that.

- You will **not** turn anything in on canvas.
- You do **not manually upload files** to your repository using GitHub's "Upload files" tool.

- You must submit your individual program even though you have a partner, since you'll be graded individually.
- Your github will accept pushes at *any* time. Once the deadline for homework is passed, if you push again, you'll rewrite over the correct time and the solution will be past the deadline and not graded. If your timestamp is 11:01PM or later, the homework will not be graded under almost any circumstance. So do not wait until 10:59PM to commit and push your changes. Pushing with only a few minutes left is **not** a valid reason to be graded.
- You cannot discuss homework outside of your partner, use tools like Chat-GPT, copy a solution–a person who cheats cheats himself.
- Struggling is part of any practiced skill–it's how we learn.

A remark: often numerical values will be infinite. Python will give you **by default** about 15 decimal places. We'll learn how to shorten these values later. For now, for example we'll write 104.17 for 104.16666666666667. On this HW, you are allowed to use round() function to round the final answer before returning it in the funtion-for example, if your final answer is 6.85457 then it's rounded equivalent is 6.85, and can be obtained by doing **round(6.8547, 2)**. The first parameter to the round() function is the actual value that you want to round, and the second argument is the number of places you want it rounded to.

Comments about Homework

You will be given the function signature without a body. For each problem, you'll replace the Python keyword **pass** with code. Do not change any signature. If you do, the autograder will fail and you'll receive a zero. If you add any extra functions, they must be local to the function you are building; otherwise, the autograder will fail. This is a very short homework to help you get started with understanding the process.

Keyboard Quickies

To uncomment multiple lines (Windows):

1. select the lines to be commented
2. Ctrl+k+u

To comment multiple lines (Windows):

1. select the lines to be commented
2. Ctrl+/

To comment or uncomment multiple lines (Mac):

1. select the lines to be commented
2. cmd + /

To undo (Windows):

- Ctrl+Z

To undo (Mac)

- Command+Z

Some of these problems were taken or inspired by the excellent introductory texts *Applied Calculus* by Tan, 2005, *Thinking Mathematically* 5^{th} *ed.* by Blitzer, 2011.

Problem 1: Volume of a Cone

The volume of a cone with radius r and height h is:

$$c(r,h) = \frac{1}{3}\pi r^2 h \tag{1}$$

For example, 2 cm radius and 5 cm height,

$$c(2,5) \approx 20.94\ cm^3 \tag{2}$$

For 3 cm radius and 7 cm height

$$c(3,7) \approx 65.97\ cm^3 \tag{3}$$

Deliverables Problem 1

- Complete the function $c()$ in the file a1.py.
- You must use math.pi from the math module.
- Round the answer to 2 decimal places. Use round(x,y) to round x to y decimal places. For example,

```
>>> round(1.252,2)  #rounds to two places
1.25
>>> round(1.252,1)  #rounds to 1 place
1.3
>>> round(1.252,0)  #rounds to 0 place
1.0
```

As an example, here's code with rounding two 1 place in the function:

```
def foo(x,y,z):
    ans_ = round((x + y + z)/3,1)
    return ans_

if __name__ == "__main__":

    """
    If you want to do some of your own testing in this file,
    please put any print statements you want to try in
    this if statement.

    You **do not** have to put anything here  """

    # my test
    print(foo(200,300,500))
```

has output:

```
333.3
```

Problem 2: Oxygen Content of a Pond

The oxygen content t days after organic waste has been dumped into a pond is given by:

$$f(t) = 100\frac{t^2 + 10t + 100}{t^2 + 20t + 100} \tag{4}$$

percent of its normal level. For example,

$$f(0) = 100 \tag{5}$$
$$f(10) = 75 \tag{6}$$

Deliverables Problem 2

- Complete the function $f()$ in the file a1.py.
- Round the value you function returns to 2 decimal places.

Problem 3: TV Viewing Patterns

According to A.C. Nielsen Co., the percent of U.S. households waching television during the weekdays (about a decade ago) starting at 4:00PM for eight hours is modeled as $P(t)$:

$$P(t) = 0.01354t^4 - 0.49375t^3 + 2.58333t^2 + 3.8t + 31.60704 \tag{7}$$

if $0 \leq t \leq 8$ where $t = 0$ corresponds to 4:00P. For example,

$$P(0) = 31.61\% \tag{8}$$
$$P(3) = 54.02\% \tag{9}$$
$$P(8) = 30.0\% \tag{10}$$

Deliverables Problem 3

- Complete the function $P()$ in the file a1.py.
- Round the answer to 2 decimal places.

Problem 4: Toxic Waste

A city's main well was recently found to be contaminated with trichloroethylene, a cancer-causing chemical, as a result of an abandoned chemical dump leaching chemicals into

the water. A proposal submitted to city council members indicates that the cost, measured in millions of dollars to remove x% of the toxic pollutant is given by:

$$cost(x) = \frac{0.5x}{100 - x} \tag{11}$$

for $0 < x < 100$. For eample, 50%, 70%, and 90% cost

$$cost(50) = \$.5 \text{ million} \tag{12}$$
$$cost(70) \approx \$1.17 \text{ million} \tag{13}$$
$$cost(90) = \$4.5 \text{ million} \tag{14}$$

Deliverables Problem 4

- Complete the function $cost()$ in the file a1.py.
- Round the answer to 2 decimal places.

Problem 5: Cowling's Rule

Cowling's rule is a method for calculating pediatric drug dosages. If a denotes the adult dosage(in milligrams) and t is the age of the child (in years), then the child's dosage is given by:

$$D(t, a) = \frac{t + 1}{24} a \tag{15}$$

For example, if $a = 500$ mg and $t = 4$ years, then

$$D(t, a) = 104.17 \text{ mg} \tag{16}$$
$$\tag{17}$$

Deliverables Problem 5

- Complete the function $D()$ in the file a1.py.
- Round the answer to 2 decimal places.

Problem 6: Flu Outbreak

During a flu outbreak in a school of 1000 children, the number of infected children, I, was expressed in terms of the number of susceptible (but still healthy) children, S, by:

$$I(S) = 192 \log_2(\frac{S}{762}) - S + 763 \tag{18}$$

For example,

$$I(100) = 101 \text{ students} \tag{19}$$

$$I(300) = 205 \text{ students} \tag{20}$$

Deliverables Problem 6

- The math module is imported for this problem.
- Complete the $I()$ function in the file a1.py.
- For this problem, you'll need to make your own input/outputs.
- You may get negative values for some inputs values that you provide to the function, and that's fine.
- Use **math.ceil()** from the math module discussed in lecture.

Problem 7: Average Cost

Let $C(q)$ be the cost of producing q items. A company's cost function is (in $) given by

$$C(q) = 0.01q^3 - 0.6q^2 + 13q + 1000 \tag{21}$$

The **average cost** $A(q)$ is given by

$$A(q) = \frac{C(q)}{q} \tag{22}$$

Deliverables Problem 7

- Complete both functions C, A in the file a1.py
- For this problem, you'll need to make your own input/outputs

Problem 8: Sales Model

A company contracted by you has built a sales model that returns the number of plastic ducks sold ($\times 10^3$)

$$hh(t) = \left\lfloor \frac{532}{1 + 869e^{-1.33t}} \right\rfloor \tag{23}$$

for $0 \leq t \leq 11$ where t is months. The floor function $\lfloor\ \rfloor$ is in the math module named math.floor() For example,

$$hh(0) = 0 \text{ ducks} \tag{24}$$

$$hh(5) = 25 \text{ ducks} \tag{25}$$

$$hh(10) = 531 \text{ ducks} \tag{26}$$

Deliverables Problem 8

- The math module is imported for this problem.
- Complete the hh function in the file a1.py.
- Round your answer to the nearest integer using math.floor()

Problem 9: Throwing a Stone

A stone is thrown straight up from the roof of an 80 ft building with an initial velocity of $64\,\text{ft/sec}$. The height (in feet) of the stone at any time t seconds is given by:

$$height(t) \quad = \quad -16t^2 + 64t + 80 \tag{27}$$

The rock will hit the ground after 5 seconds:

$$height(5) \quad = \quad 0 \text{ feet} \tag{28}$$

Deliverables Problem 9

- Complete the $height$ function in the file a1.py
- Round the answer to 2 decimal places.

Problem 10: Treating Heart Attacks

According to the American Heart Association, the treatment benefit for heart attacks depends on the time (hours) until treatment and is described by:

$$B(t) \quad = \quad \frac{0.44t^4 + 700}{0.1t^4 + 7} \tag{29}$$

for $(0 \leq t \leq 24)$

Deliverables Problem 10

- Complete the B function in the file a1.py.
- For this problem, you'll need to make your own input/outputs.
- Round the answer to 2 decimal places.

Problem 11: Roots to the Quadratic

Recall that a quadratic is a function:

$$q(x) \quad = \quad ax^2 + bx + c \tag{30}$$

A root is a number that makes the function zero. For example, if

$$q(x) \quad = \quad 2x^2 + 5x - 12 \tag{31}$$

then the two roots are $x = -4$ and $x = 3/2$:

$$q(-4) \quad = \quad 2(-4)^2 + 5(-4) - 12 = 32 - 20 - 12 = 0 \tag{32}$$

$$q(3/2) \quad = \quad 2(3/2)^2 + 5(3/2) - 12 = 2(9/4) + 15/2 - 12 = (24/2) - 12 = 0 \tag{33}$$

On the other hand, 1 is not. Implement a function quad(a,b,c,x) that returns True if v is a root for $ax^2 + bx + c$ and False otherwise. Running the function:

```
1 print(quad(2,5,-12,-4))
2 print(quad(2,5,-12,3/2))
3 print(quad(2,5,-12,1))
```

gives output:

```
1 True
2 True
3 False
```

Deliverables Problem 11

- Complete the $quad$ function in the file a1.py

Problem 12: Sinking Fund

This model describes a “sinking fund." Money is periodically set aside until a date is reached. One important kind is retirement–money that will be available when you stop working. You must know the payment amount P, the number of times a year you make the payment n, the number of years you make the payment t, and the interest rate r. For example,

- If you pay $22,000 once a year for seven years that has 6% compounded annually, you’ll have, at the end $184,664.43.
- If you make $500 monthly that has 4% (about the current best rate) compounded monthly for 20 years, you’ll have $183,387.31
- If $1,200 is deposited quarterly with 8% compounded quarterly for 10 years, you’ll have $72,482.38

The sinking fund is:

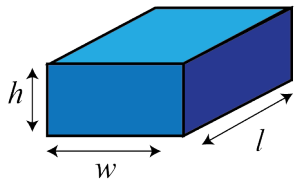
$$R(P, r, n, t) \quad = \quad P\left[\frac{(1+\frac{r}{n})^{nt}-1}{\frac{r}{n}}\right] \tag{34}$$

Deliverables Problem 12

- Complete the R function in the file a1.py
- Round to two decimal places.

Problem 13: Rectangular Solid

Complete the Python function that gives the total surface area S of a rectangular solid with dimensions w, l, h shown here:



The surface area is:

$$S(w, l, h) \quad = \quad (?) \tag{35}$$

Deliverables Problem 13

- Complete the S function in the file a1.py
- You must create your own solutions for this problem.

Communication

When connecting with a new person, you should introduce yourself, give a short message describing the purpose, then salutation. Please use and adapt this format when connecting to a new partner. The student pairs are given on the next page.

```
Dear Student,

My name is X Y and I'm a student in your C200,
but a different section.  It looks like we're
partners this week.  I'm generally free on
Thursday, Saturday, Sunday.  I'm hoping we
can knock most of this out on our first meeting.
Luddy is a great place if we can snag a
```

conference room.

Take care,

Another Student

Format for InScribe post

Hi Instructor/ TA <Whom ever they want to
ask question/ Share something>,

If you are posting about lecture/Notes
My name is Student-A. I am in Dr. (mention faculty instructor name)
lecture section. This is regarding ....

If you are posting about labs
My name is Student-A. I am in the lab section that meets on
DAY, TIME of your lab and NAME OF THE TA.
This is regarding ....

Thanks and regards,
<Name>
<User name>

Programming partners

ayoajayi@iu.edu, avraya@iu.edu
swconley@iu.edu, rorymurp@iu.edu
lpfritsc@iu.edu, tangtom@iu.edu
oeichenb@iu.edu, wtrucker@iu.edu
jc168@iu.edu, trenstev@iu.edu
braybrya@iu.edu, asidda@iu.edu
escolber@iu.edu, wattsra@iu.edu
gmhowell@iu.edu, aayushah@iu.edu
daminteh@iu.edu, dsummit@iu.edu
keswar@iu.edu, nmulheri@iu.edu
bkante@iu.edu, ilynam@iu.edu
oakinsey@iu.edu, amanocha@iu.edu
ek37@iu.edu, annaum@iu.edu, sezinnkr@iu.edu
kjj6@iu.edu, mmunaf@iu.edu
browdeon@iu.edu, dyashwar@iu.edu
mbrockey@iu.edu, aselki@iu.edu
jhhudgin@iu.edu, lreddell@iu.edu
mrfehr@iu.edu, evataylo@iu.edu
ahavlin@iu.edu, wilsori@iu.edu
aakindel@iu.edu, emisimps@iu.edu
sgaladim@iu.edu, majtorm@iu.edu
maladwa@iu.edu, jlzhao@iu.edu
ameydesh@iu.edu, jsadiq@iu.edu
tcconnol@iu.edu, ism1@iu.edu
mohiambu@iu.edu, anajmal@iu.edu
brownset@iu.edu, wtubbs@iu.edu
mkleinke@iu.edu, amyawash@iu.edu
masharre@iu.edu, kamaharj@iu.edu
bcison@iu.edu, reddyrr@iu.edu
kraus@iu.edu, jactrayl@iu.edu
celechav@iu.edu, rvinzant@iu.edu
maudomin@iu.edu, masmatth@iu.edu
micahand@iu.edu, ao9@iu.edu
andhugo@iu.edu, marystre@iu.edu
matgarey@iu.edu, hasiddiq@iu.edu
criscruz@iu.edu, abiparri@iu.edu
nabussel@iu.edu, ruska@iu.edu
egoldsto@iu.edu, pravulap@iu.edu

micedunn@iu.edu, clmcevil@iu.edu
mdoxsee@iu.edu, sahishah@iu.edu
fkanmogn@iu.edu, epautsch@iu.edu
jakchap@iu.edu, wilsonnf@iu.edu
lawmat@iu.edu, audtravi@iu.edu
johnguen@iu.edu, liwitte@iu.edu
ethbrock@iu.edu, crmoll@iu.edu
leegain@iu.edu, lqadan@iu.edu
kyhigg@iu.edu, myswill@iu.edu
wanjiang@iu.edu, apathma@iu.edu
amkurz@iu.edu, phamvinh@iu.edu
evacoll@iu.edu, patel89@iu.edu
aketcha@iu.edu, rosenbbj@iu.edu
elybrewe@iu.edu, perezand@iu.edu
jongrim@iu.edu, fmahamat@iu.edu
jeonjunh@iu.edu, wtatoole@iu.edu
jacklapp@iu.edu, surapapp@iu.edu
dkkosim@iu.edu, dwo@iu.edu
nolakim@iu.edu, giomayo@iu.edu
kapgupta@iu.edu, chrimanu@iu.edu
skatiyar@iu.edu, lmadiraj@iu.edu
mrcoons@iu.edu, isaramir@iu.edu
huhasan@iu.edu, jwember@iu.edu
bjdahl@iu.edu, ammulc@iu.edu
jdemirci@iu.edu, leolin@iu.edu
aberkun@iu.edu, qshamsid@iu.edu
coopelki@iu.edu, ansakrah@iu.edu
ajgrego@iu.edu, cltran@iu.edu
jhar@iu.edu, aptheria@iu.edu
dblackme@iu.edu, scotbray@iu.edu
xknabel@iu.edu, ap79@iu.edu
aroraarn@iu.edu, jarlmint@iu.edu
bencho@iu.edu, nichojop@iu.edu
gcopus@iu.edu, mmarotti@iu.edu
coopjose@iu.edu, cialugo@iu.edu
nbernot@iu.edu, thnewm@iu.edu
daxbills@iu.edu, aidnschi@iu.edu
gandhira@iu.edu, thomps16@iu.edu
caegrah@iu.edu, dukthang@iu.edu
davgourl@iu.edu, patedev@iu.edu

quecox@iu.edu, apavlako@iu.edu
enahern@iu.edu, muyusuf@iu.edu
hamac@iu.edu, coenthom@iu.edu
ryanbren@iu.edu, mszczas@iu.edu
gdhuber@iu.edu, voram@iu.edu
howamatt@iu.edu, jwetherb@iu.edu
delkumar@iu.edu, cnyarko@iu.edu
mbeigie@iu.edu, asultano@iu.edu
spdamani@iu.edu, asteini@iu.edu
liansia@iu.edu, jpochyly@iu.edu
ag69@iu.edu, ijvelmur@iu.edu
adhuria@iu.edu, sasayini@iu.edu
althart@iu.edu, jcn1@iu.edu
nfarhat@iu.edu, arirowe@iu.edu
allencla@iu.edu, pp31@iu.edu
ovadeley@iu.edu, etprince@iu.edu
alscarr@iu.edu, emmmccle@iu.edu
avulas@iu.edu, impofujr@iu.edu
nharkins@iu.edu, aledminc@iu.edu
garcied@iu.edu, ragmahaj@iu.edu
saganna@iu.edu, meyerli@iu.edu
fu7@iu.edu, snyderjk@iu.edu
bcdutka@iu.edu, keasandl@iu.edu
oingogl@iu.edu, aptarin@iu.edu
jwdrew@iu.edu, emluplet@iu.edu
tychid@iu.edu, ntuhl@iu.edu
jacobben@iu.edu, cinivo@iu.edu
aleliffe@iu.edu, aamathew@iu.edu
arklonow@iu.edu, btasa@iu.edu
ohostet@iu.edu, wwtang@iu.edu
howardbw@iu.edu, wlyzun@iu.edu
hh35@iu.edu, rtrammel@iu.edu
gracarmi@iu.edu, gs29@iu.edu
jdc6@iu.edu, lmeldgin@iu.edu
harmonnc@iu.edu, lpelaez@iu.edu
ruqchen@iu.edu, dmmullin@iu.edu
earuland@iu.edu, drsnid@iu.edu
kdembla@iu.edu, ysanghi@iu.edu
anlego@iu.edu, megapaul@iu.edu
apbabu@iu.edu, violuway@iu.edu

zguising@iu.edu, madymcsh@iu.edu
cpkerns@iu.edu, skp2@iu.edu
flynncj@iu.edu, brayrump@iu.edu
agawrys@iu.edu, nniranj@iu.edu
eghough@iu.edu, zshamo@iu.edu
josespos@iu.edu, samrile@iu.edu
ajeeju@iu.edu, iperine@iu.edu
achordi@iu.edu, ryarram@iu.edu
schinitz@iu.edu, jwmullis@iu.edu
austdeck@iu.edu, pw18@iu.edu
brhint@iu.edu, adsize@iu.edu
makinap@iu.edu, gmichna@iu.edu
deombeas@iu.edu, antando@iu.edu
maxklei@iu.edu, gmpierce@iu.edu
fkeele@iu.edu, wardjohn@iu.edu
cannan@iu.edu, aditpate@iu.edu
phjhess@iu.edu, dksanapa@iu.edu
linjaso@iu.edu, reedkier@iu.edu
tfreson@iu.edu, bdyiga@iu.edu
diebarro@iu.edu, clscheum@iu.edu
eakanle@iu.edu, anemlunc@iu.edu
stkimani@iu.edu, moralemd@iu.edu
zacbutle@iu.edu, jomeaghe@iu.edu
aaamoako@iu.edu, kvpriede@iu.edu
twfine@iu.edu, tzuyyen@iu.edu
kaneai@iu.edu, ksadiq@iu.edu
nfelici@iu.edu, rt11@iu.edu
kekchoe@iu.edu, sahaan@iu.edu
scbik@iu.edu, patekek@iu.edu
pyphealy@iu.edu, jdw14@iu.edu
fdonfrio@iu.edu, savebhat@iu.edu
abellah@iu.edu, utwade@iu.edu
jkielcz@iu.edu, owysmit@iu.edu
ceub@iu.edu, lukastef@iu.edu
sharor@iu.edu, tarturnm@iu.edu
sydecook@iu.edu, justyou@iu.edu
sfuneno@iu.edu, krbpatel@iu.edu
clearle@iu.edu, ryminer@iu.edu
grafe@iu.edu, joshroc@iu.edu
willhowa@iu.edu, gepearcy@iu.edu

mdonato@iu.edu, shrrao@iu.edu
krisgupt@iu.edu, bcmarret@iu.edu
cfampo@iu.edu, singkani@iu.edu
spgreenf@iu.edu, gavilleg@iu.edu
mbekkem@iu.edu, blswing@iu.edu
spgerst@iu.edu, ntorpoco@iu.edu
kaihara@iu.edu, gsilingh@iu.edu
ddrotts@iu.edu, awsaunde@iu.edu
colrkram@iu.edu, sahimann@iu.edu
jdgonzal@iu.edu, jaslnu@iu.edu
ageorgio@iu.edu, gszopin@iu.edu
ridbhan@iu.edu, ltmckinn@iu.edu
zhatfie@iu.edu, tpandey@iu.edu
sg40@iu.edu, mveltri@iu.edu
dja1@iu.edu, tt13@iu.edu
sakalwa@iu.edu, orrostew@iu.edu
abuswell@iu.edu, nsatti@iu.edu
amkhatri@iu.edu, vrradia@iu.edu
jtbland@iu.edu, mimimurp@iu.edu
tchapell@iu.edu, deturne@iu.edu
migriswo@iu.edu, lvansyck@iu.edu
nihanas@iu.edu, jnzheng@iu.edu
hk120@iu.edu, rpoludas@iu.edu
alchatz@iu.edu, nrizvi@iu.edu
nokebark@iu.edu, bmpool@iu.edu
seangarc@iu.edu, mjroelle@iu.edu
lflenoy@iu.edu, samyuan@iu.edu
saecohen@iu.edu, rafir@iu.edu
jabbarke@iu.edu, tolatinw@iu.edu
efritch@iu.edu, mz24@iu.edu
jwcase@iu.edu, vmungara@iu.edu
simadams@iu.edu, schwajaw@iu.edu
adwadash@iu.edu, erschaef@iu.edu
aaragga@iu.edu, dernguye@iu.edu
blacount@iu.edu, nmr1@iu.edu
cgkabedi@iu.edu, fshamrin@iu.edu
hermbrar@iu.edu, mnimmala@iu.edu
laharden@iu.edu, nlippman@iu.edu
adiyer@iu.edu, awolkind@iu.edu
skunduru@iu.edu, jtsuter@iu.edu

mwclawso@iu.edu, vyeruba@iu.edu
hawkjod@iu.edu, smremmer@iu.edu
dce@iu.edu, aranjit@iu.edu
rl29@iu.edu, cstancom@iu.edu
edfran@iu.edu, maklsmit@iu.edu
lcoveney@iu.edu, ajtse@iu.edu
bencalex@iu.edu, mzagotta@iu.edu
bellcol@iu.edu, jacmanto@iu.edu
arnadutt@iu.edu, maglowe@iu.edu
tfleuran@iu.edu, patelsak@iu.edu
wgurley@iu.edu, jwu6@iu.edu
cuizek@iu.edu, jlopezmo@iu.edu
mdiazrey@iu.edu, pateishi@iu.edu
vkommar@iu.edu, anassour@iu.edu
marganey@iu.edu, msorsor@iu.edu
anrkram@iu.edu, cmarcuka@iu.edu
agrevel@iu.edu, emgward@iu.edu
leokurtz@iu.edu, rwan@iu.edu
joehawl@iu.edu, tylewhit@iu.edu
marcchao@iu.edu, ir1@iu.edu
milhavil@iu.edu, rnschroe@iu.edu
ejhaas@iu.edu, ap52@iu.edu
loggreen@iu.edu, woodsky@iu.edu
apchavis@iu.edu, benprohm@iu.edu
ethickma@iu.edu, asaokho@iu.edu
khannni@iu.edu, vpolu@iu.edu
laburkle@iu.edu, nmata@iu.edu
ckdiallo@iu.edu, aveluru@iu.edu
wilcusic@iu.edu, cjwaller@iu.edu
greenpat@iu.edu, gavsteve@iu.edu
ballachm@iu.edu, pricemo@iu.edu
alelefeb@iu.edu, mehtriya@iu.edu
alyjone@iu.edu, jneblett@iu.edu
rcaswel@iu.edu, cmvanhov@iu.edu
nmcastan@iu.edu, myeralli@iu.edu