

L2 PROBLEM 11 (5/5 points)

Assume that two variables, `varA` and `varB`, are assigned values, either numbers or strings.

Write a piece of Python code that prints out **one** of the following messages:

- `"string involved"` if either `varA` or `varB` are strings
- `"bigger"` if `varA` is larger than `varB`
- `"equal"` if `varA` is equal to `varB`
- `"smaller"` if `varA` is smaller than `varB`

For problems such as these, do not include `raw_input` statements or define the variable `varA` or `varB`. Our automating testing will provide values of `varA` and `varB` for you - so write your code in the following box assuming `varA` and `varB` are already defined.

```
1 if type(varA) == str or type(varB) == str:
2     print 'string involved'
3 elif varA > varB:
4     print 'bigger'
5 elif varA == varB:
6     print 'equal'
7 else:
8     # If none of the above conditions are true,
9     # it must be the case that varA < varB
10    print 'smaller'
```

Correct

```
if type(varA) == str or type(varB) == str:
    print 'string involved'
elif varA > varB:
    print 'bigger'
elif varA == varB:
    print 'equal'
else:
    # If none of the above conditions are true,
    # it must be the case that varA < varB
    print 'smaller'
```

Test results

[See full output](#)

CORRECT

[See full output](#)

Explanation

Is `type(varA) == str or type(varB) == str` equivalent to `type(varA) or type(varB) == str` ? Those are not equivalent because of Python precedence (some operations have higher precedence than others). The `==` has higher precedence than the `or` so it will get evaluated first. Therefore:

```
type(varA) or type(varB) == str
```

Will evaluate to the following, if we explicitly put parentheses:

```
type(varA) or ( type(varB) == str )
```

```
True or ( type(varB) == str )
```

```
( True )
```

Because "anything" or "True" will just take the value of "True" (by boolean algebra).

And the other expression:

```
type(varA) == str or type(varB) == str
```

Will evaluate to the following, if we explicitly put parentheses:

```
( type(varA) == str ) or ( type(varB) == str )
```

So you will have to check each of the expressions in the parentheses to see whether they are true or not to determine the final result. So these two are not equal.

Hint: Remember the `type` function?

Hint: Recall that if you want to check whether the type of a variable is a specific kind, you can compare the type of the variable to the type of a known object.

[Check](#)[Hide Answer](#)[Show Discussion](#)[New Post](#)

EdX offers interactive online classes and MOOCs from the world's best universities. Online courses from MITx, HarvardX, BerkeleyX, UTx and many other universities. Topics include biology, business, chemistry, computer science, economics, finance, electronics, engineering, food and nutrition, history, humanities, law, literature, math, medicine, music, philosophy, physics, science, statistics and more. EdX is a non-profit online initiative created by founding partners Harvard and MIT.

© 2015 edX, some rights reserved

About edX

[About](#)[News](#)[Contact](#)[FAQ](#)[edX Blog](#)[Donate to edX](#)[Jobs at edX](#)

Follow Us

[!\[\]\(c1168d6a8b365d11e842ece304635fa7_img.jpg\) Twitter](#)[!\[\]\(cbd8541a32dfc32f356f5c6c994b0a21_img.jpg\) Facebook](#)[!\[\]\(d3e32d099174a7c248ec1f564ee4f69c_img.jpg\) Meetup](#)[!\[\]\(40770d9ed6ed4f1222ebf89a1396e8b2_img.jpg\) LinkedIn](#)[!\[\]\(ccd39a0dc6d5afcc151e1371f9462f58_img.jpg\) Google+](#)

[Terms of Service and Honor Code](#)

[Privacy Policy \(Revised 4/16/2014\)](#)