

Results

Shang Xiao — 1st Attempt



10
Out of 10 points

186:34:35
Time for this attempt

This assessment has unlimited attempts. Take Now

Attempt History

Results	Points	Score	(Highest score is kept)
Attempt 1	10 of 10	100%	(Highest score)

Your Answers:

1 3 / 3 points

Put the following in the proper order:
Do stuff inside loop
End of loop stuff
Set up loop stuff

Set up loop stuff

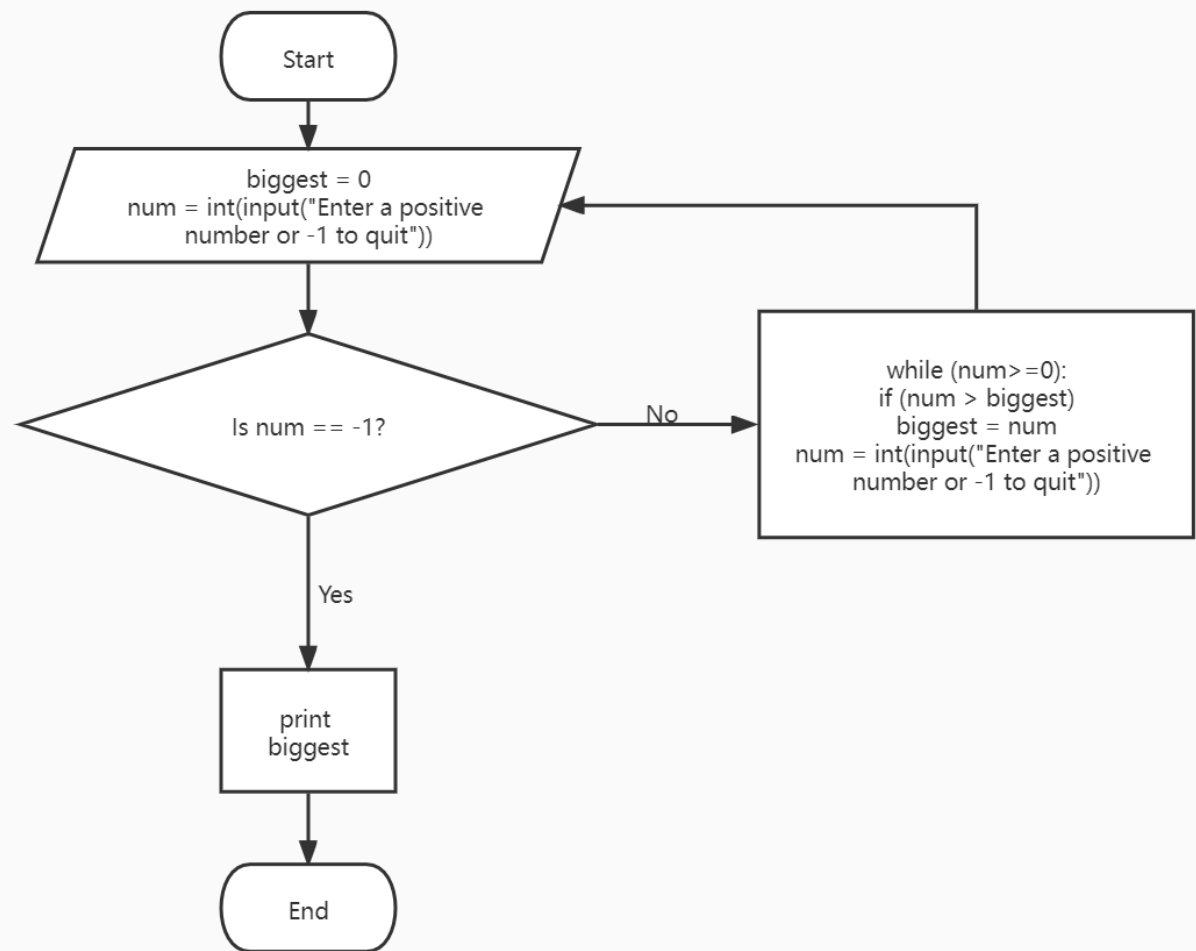
Do stuff inside loop

End of loop stuff

Correct

2 2 / 2 points

Turn in a jpg version of the flow chart from the In Class Activity where you find the biggest number entered in a loop.



Correct

3 2 / 2 points

List one possible loop error and code that demonstrates it in action.

One common mistake I often make is `SyntaxError: 'break' outside loop`. This means I used a break condition in a wrong spot.

Example:

```
def main():
    value1 = int(input("Please enter number 1:\n"))
    value2 = int(input("Please enter number 2:\n"))
    value3 = int(input("Please enter number 3:\n"))

    while True:
        if value1<=0 or value2<=0 or value3<=0:
            print("At least one number was negative, try again")
            value1 = int(input("Please enter number 1:\n"))
            value2 = int(input("Please enter number 2:\n"))
            value3 = int(input("Please enter number 3:\n"))

        max = value1
        if value2 > value1:
            max = value2
        if value3 > value2:
            max = value3
        print("The max number is", max)
    break

main()
```

In this case, `break` is outside loop, which gives me error: `SyntaxError: 'break' outside loop`. Because `break` should be indented, which means: stop the loop after printing result. Thus, the correct code should be:

```
def main():
    value1 = int(input("Please enter number 1:\n"))
    value2 = int(input("Please enter number 2:\n"))
    value3 = int(input("Please enter number 3:\n"))

    while True:
        if value1<=0 or value2<=0 or value3<=0:
            print("At least one number was negative, try again")
            value1 = int(input("Please enter number 1:\n"))
            value2 = int(input("Please enter number 2:\n"))
            value3 = int(input("Please enter number 3:\n"))

        max = value1
        if value2 > value1:
            max = value2
        if value3 > value2:
            max = value3
        print("The max number is", max)
        break

main()
```

Correct

4 2 / 2 points

List a second (different) loop error and code that demonstrates it in action

Another error I find interesting is an infinite loop. This means at some point we failed to ensure we exit the loop after it completes what we wanted it to do.

Considering the following code:

```
def main():
    i = 0
    while i < 3:
        print("Nice")

main()
```

In this case, if we run the code, it will print "Nice" forever, and we have to hit ctrl+c to manually stop the process, which also gives us an error:

Traceback (most recent call last):

File "C:\pythonProject\venv\CS5001\Lecture\L4\l04discussion.py", line 35, in <module>

main()

File "C:\pythonProject\venv\CS5001\Lecture\L4\l04discussion.py", line 33, in main

print("Nice")

KeyboardInterrupt

Process finished with exit code -1073741510 (0xC000013A: interrupted by Ctrl+C)

This is a keyboard interrupt error, this means we have manually stopped the program instead of the loop exiting in the correct way itself.

We then have to debug, by adding a line of code:

```
def main():
    i = 0
    while i < 3:
        print("Nice")
        print("i=", i)
main()
```

If we run again, we can find, "i" was always = 0, and this is why we had an infinite loop, we need to make sure everytime after each execution we update the counter, which is the value assigned to "i":

```
def main():
    i = 0
    while i < 3:
        print("Nice")
        print("i=", i)
        i += 1
main()
```

Now we can exit the loop in the correct way.

Correct

5 1 / 1 point

What was one of the keywords for loops that you are NOT going to use because it's sloppy?

The 'break' statement because programmers often find it is confusing when initialization and update of code happen on the same line, we want to introduce a new variable to control the loop.

Correct