

First day assessment, CIS 122-01 and 04, spr 22

In any programming language that you know, write a function or method called `print_index_of_max(alist)`. This function should take as its argument a list or array (your choice) of floating-point numbers, and should print the index of the maximum of those numbers. In other words, it should print the position in the array of the biggest number in the array. If you cannot remember any programming notation at all, use pseudocode.

My rough grading scale was:

- A = "Good" (4 students). Completely or nearly-correct code (only working for positive numbers is ok).
- B = "On the right track" (14 students). Typically had the right idea and wrote a complete method, but used the wrong loop pattern, or solved a related problem (e.g., printing the maximum).
- C = "Writes wrong code" (13 students). A mixed bag. Some had the right intent but could only write half a method, or only solved it for 3 elements. Some wrote code to solve a completely different problem. However, all showed they COULD string a thought together in code.
- D = "Can't write code" (18 students). Some just had a couple of header lines. Many wrote code where the intent seemed to change line by line. Several said they realized they'd forgotten a lot.
- F = "Didn't try" (2 students). Submitted blank bits of paper, even though I told them to just write the algorithm in English.

This sums to 51 students, but only 47 answers are shown, as 4 were submitted via email. For the purposes of this document, student answers have been cleaned up as follows:

- All indentation has been changed to 4-space tab stops
- I used Python syntax-coloring by default
- Written notes are shown as comments
- Crossed-out code and workings are not included
- Otherwise, all code is rendered "as-is", including typos

1

```
print_index_of_max(input list)
  for I in (length of list-1)
    x = I
    if I < I + 1
```

2

```
Print_index_of_max
{
  public static void main(String[] args)
  {
    Sys.out.print(4+2);
    Sys.out.print(2-2);
    Sys.out.print("42+5");
```

```

        Sys.out.print(4 . 5);
    }
}

```

3

```

public class alist {
    public static void main(String[] args) {

```

4

```

Class main {
    public static void(String[] args) {
        Public static int print_index_of_max(alist) {
            int alist = [1,2,3,4];
            System.out.println(alist.max);
        }
    }
}

```

5

```

system void() print_index_of_max(ref alist) {
    max = alist[0];
    index = 0;
    for (i = 1; alist.length; i++) {
        if alist[i] > max {
            max = alist[i];
            index = i;
        }
    }
    Console.write(index);
}

```

6

```

def (print_index_of_max(alist))
alist = [1,2,3,4,5]
for I in alist
    print(alist.max)

```

7

```

a_list_of_numbers = {1, 2, 3, 4, 5}
max_index = 0
for i in a_list_of_numbers
    while index > 0
        if i > max_index
            max_index = i
print(max_index)

```

8

```

def(print_index_of_max)
alist=(1, 2, 3, 4, 5, 6)
flot(alist)
for I in alist
    print max(alist)

```

9

```

a = float(int(input("Enter the first number:")))
b = float(int(input("Enter the second number:")))
c = float(int(input("Enter the third number:")))
If a > b and a > c;
    print("The first number is the biggest.")
else if b > a and b > c;
    print("The second number is the biggest.")
else;
    print("The third number is the biggest.")

```

10

```

def alist:
    alist = [4, 16, 9, 10, 12]
    print alist
    if alist[0] > alist[1]
        print alist[0]
    elseif
        print alist[1]

```

11

```

list = [1.1, 1.2, 1.3]
x = 0
def p_i_o_m(list):
    newList = list.sort()

```

```
x = list[0]
x = list.find(x) # I believe .find gets the index
return x
```

12

1. define the function print_index_of_max(alist)
2. write an if statement that takes each number in the list and

13

```
Def print_index_of_max(alist):
    x = alist.Max()
    for i in range(len(alist)):
        y += 1
        if i == x:
            Break
    return print(y-1)
```

14

```
def Print_index_of_max(alist):
    max = 0
    for i in range:
        if i > max:
            max = i
    return max
print max
```

15

```
print_index_of_max(alist)
value = alist[0]
for i in range(0, len(alist) - 1):
    if alist[i+1] > value:
        value == alist[i+1]
print alist[value]
```

16

```
x = 5;
y = 10;
z = 20;
```

```

{
    if z > y then
        print z;
    else if y > x then
        print y;
    else print z;
}

```

We need to print a number in a position of the other so when we add a big number we can use the algebra great than or less than in a loop to show the hierarchy of the number

17

```

include<stdio.h>
void main()
{
    int i, max, k
    float A[i]
    printf("Enter the number of indexes")
    cin >> i;
    {
        for 0 <= j <= i
            printf("Enter the number ")
            cin >> A[j]
            j++
        }
        for 0 <= k <= i
            if A[k] > A[k-1]
                man = A[k]
                k++
        }
    }
    printf(man)
    return()
}

```

18

```

x = [1, 2, 3, 4, 5]
def print_index_of_max(alist)
    i = 0
    while i > len(x):
        if alist[i] > alist[i+1]:
            c = list[i]
        else:
            c = list[i+1]
    return c
print(print_index_of_max(x))

```

19

20

21

```
alist = {1, 2, 3, 5, 6, 7}
def print_index_of_max(alist):
    for i in range(len(alist)+1):
        if alist[i] < alist[i+1]:
            return i += 1
        else:
            return i
```

22

```
array alist;
int print_index_of_max(alist);
for (i = 0; i > 1; i++)
    int u;
    int great;
    u = i + 11
    if alist(i > u);
        great = i;
println(int great, "is the greatest");
# missing how to use array positions
```

23

```
print_index_of_max = float([list+])
list = 1,3
if list < 3:
    print("None")
else
    print(alist)
```

24

```
# I don't know how it's done but I have a guess
# You put the values in a list.
```

```
# Then you use an if/else to figure out which number is the largest (I don't
remember how to write the code)
# There is a built-in function to show you indexes of the values
# You use that to show you the index of the greatest number
```

25

```
L1 = [1.1, 1.2, 1.4, 1.8, 2.9, 4.6, 19.7, 197.26]
L1max = len(L1)
# because the last index of this list is the maximum value
# and I can't remember the function for getting the index of the max
print(L1max)
```

26

```
print_index_of_max(alist)
index[ ]
foreach(i in alist)
    if i > 0
console.writeline(i)
```

27

```
print_index_of_max(2list):
    2list = temp[ ]
    Print("The max index is " . max(2list))
```

28

```
def print_index_of_max(alist)
    x = 0
    for i in range(0, alist)
        if (i > i + 1)
            x = i
    print(alist.index(x))
```

29

```
{
    aListOfNumbers =
    # declare our variables
    # our list of numbers
}
```

30

```
print("index of max")
sum("index of max")
# So, I don't get the question at all but I guess I know the basics of coding
```

31

```
print_index_of_max(alist) = [ ]
for i in print_index_of_max(alist)
    print_index_of_max = max
    print(max)
```

32

```
# The function has to do with identifying the biggest number within a range of
multiple numbers. I do not recall any programming notation at this exact moment
due to how long I haven't done any form of programming. I am certain that with a
visual reminder and practice I will quickly remember what to do.
```

33

```
alist = [1.1, 1.2, 2.0, 3.5, 1.4]
for i in alist:
    max = 0
    if i > max:
        max = i
print(alist.index(max))
```

34

```
# I really don't remember much of anything from CIS 121, unfortunately. Trying to
learn it fully online wasn't great for me. Anything that I do remember is
fragmented in my brain. So I can't put it into anything that makes sense
```

35

```
def print_index_of_max(alist):
    alist = (1, 2, 3, 4, 5, 6) # list of numbers
    max = get.maximum(alist) # I think that's how you get max?
```



```
index_of_max = idx(max)
return
```

36

```
Def print_index_of_max(alist):
    alist.sort()
    total = len(alist)
    print("The position of the largest number after sorting the list is", total)
```

37

```
positive_
    if 0 > 1
        0 = +1
    return positive_
negative_
    if 0 < -1
        0 = -1
    return negative_
# I don't remember some of the codes that I learned in my last class
```

38

```
some_list = [1,2,3,4,5]
def print_index_of_max(some_list):
    for elem in some_list:
# I forgot a command which prints an index of the elements
# The next step would be to compare elements and when the maximum is found to call
the funct
```

39

```
def print_index_of_max(alist):
# I can't remember any things. What should I do??
```

40

```
def print_index_of_max(alist):
    biggest_number = 0      # set variable
    for i in range(alist):  # examines each number
        if alist[i] > biggest_number:  # compares number
            biggest_number = alist[i]
```

```
        index = i
    print(index)
```

41

```
# alist should contain the list of floating-point numbers then
def index_of_max( ):
# You for sure need something that will know the position of numbers for indexing.
I'm thinking len with something I forgot
# The main function will know the maximum for the list as well.
# You need to put those together to use this function
# like with a for loop or if statement they go to each position and ask is that
number the max, yes or no, if yes print it, if no go to next position and ask
again
```

42

```
alist = input("Enter the list of numbers:")
alist = [alist]
def print_index_of_max(alist):
    alist = alist.sort()
    return "The biggest number is ", alist[0]
```

43

```
print("Hello world")
name = input("Enter name:")
user_name = print("my name is" + name)
calc_later = 2+2
class(employees):
    __init__class
        name = init.name
        role = init.role
        email = init.email
print(calc_later)
print(user_name)
```

44

```
x = int(input(" "))
for i in range of x:
    y == x+i
    x = 0
print(y)
B = int(input(" "))
```

```
B = B+1
print([B])
print(B%2)
```

45

```
alist = (1,2,4,12,19,9)
def print_index_of_max(alist):
    max = 0
    index = 0
    for i in range len(alist):
        if alist[i-1] > max
            max = alist[i+1]
            index = i-1
    return print(index)
```

46

```
def print_index_of_max(alist):
    n = length(alist)
    max = 0
    for i in range n:
        if alist[i] > max:
            max = alist[i]
        i +=
    return max
```

47

```
def print_index_max(alist)
    alist.sort()
    print(alist[0])
```