

Learn Python the Fun Way

Liana Bakradze

liana.bakradze@jetbrains.com
@med_vector

About Me



Saint Petersburg

Learn Python Resources

codecademy



pythonanywhere



trinket



Why Games?



No fear of failure



Reward



Competition



Visualisation

CODE COMBAT

✓ Dodge the fireballs forever.
✓ Under 4 statements.

GOALS: **SUCCESS!**



```
1 # Code normally executes in the order it's written.  
2 # Loops repeat a block of code multiple times.  
3 # Use tab or 4 spaces to indent the move lines under the loop.  
4  
5 while true:  
6     hero.moveRight()  
7     # Add a moveLeft command to the loop here  
8     hero.moveLeft()  
9  
10
```

RUN

DONE

moveDown()
moveLeft()
moveRight()
moveUp()

while-true loop
attack(target)

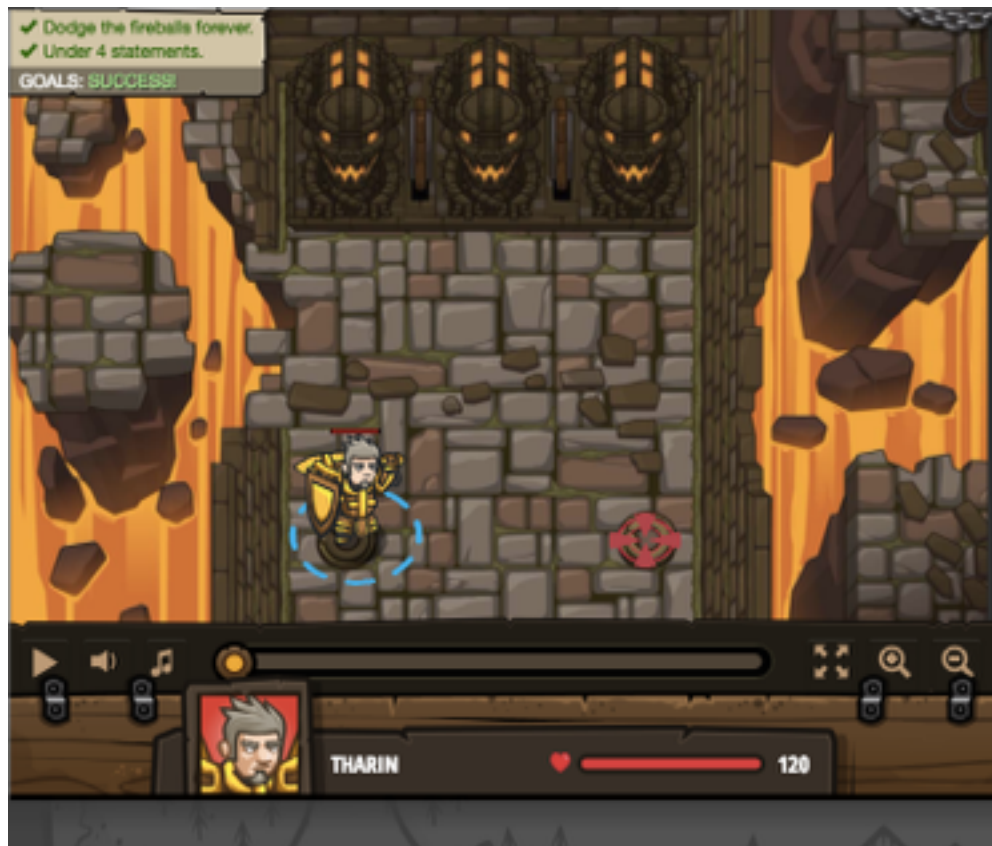
findNearestEnemy(...)



THARIN

❤️  120

CODE COMBAT



CODE COMBAT

✓ Dodge the fireballs forever.
✓ Under 4 statements.

GOALS: SUCCESS!



THARIN 120

```
1 # Code normally executes in the order it's written.
2 # Loops repeat a block of code multiple times.
3 # Use tab or 4 spaces to indent the move lines under the loop.
4
5 while True:
6     hero.moveRight()
7     # Add a moveLeft command to the loop here
8     hero.moveLeft()
9
10
```

RUN DONE

```
moveDown()
moveLeft()
moveRight()
moveUp()
```


```
while-true loop
attack(target)
```

```
findNearestEnemy(...)
```


CODE COMBAT

✓ Dodge the fireballs forever.
✓ Under 4 statements.

GOALS: SUCCESS!



```
1 # Code normally executes in the order it's written.  
2 # Loops repeat a block of code multiple times.  
3 # Use tab or 4 spaces to indent the move lines under the loop.  
4  
5 while true:  
6     hero.moveRight()  
7     # Add a moveLeft command to the loop here  
8     hero.moveLeft()  
9  
10
```

RUN DONE

moveDown() while-true loop findNearestEnemy(...
moveLeft() attack(target)
moveRight()
moveUp()

THARIN 120

CODE COMBAT

✖ Dodge the fireballs forever.
✔ Under 4 statements.
GOALS: **FAILED**



```
1 # Code normally executes in the order it's written.  
2 # Loops repeat a block of code multiple times.  
3 # Use tab or 4 spaces to indent the move lines under the loop.  
4  
5 while True:  
6     hero.moveRight()  
7     # Add a moveLeft command to the loop here  
8  
9
```

RUN

moveDown()
moveLeft()
moveRight()
moveUp()

while-true loop
attack(target)

findNearestEnemy(...)



THARIN

❤️ 120

CODE COMBAT

Defeat the ogres. (2/3)

- ✗ Your hero must survive.
- ✗ Bonus: no code problems.

GOALS: **FAILING**

Ack

Fix Your Code

Line 8: attack's argument target should have type unit, but got string: "Ack1". Attack "Ack", not "Ack1".

Your hero will need more armor to win this fight! Go buy the tarnished bronze breastplate.

Open Item Shop

```
1 # Defeat the ogres.
2 # Remember that they each take two hits.
3
4 ✓ hero.attack("Rig")
5 ✓ hero.attack("Rig")
6 ✓ hero.attack("Gurt")
7 ✓ hero.attack("Gurt")
8 ! hero.attack("Ack1")
9 hero.attack("Ack1")
10
```

RUN

hero.moveDown()
hero.moveLeft()
hero.moveRight()
hero.moveUp()

hero.attack(target)

THARIN

0

CODE COMBAT

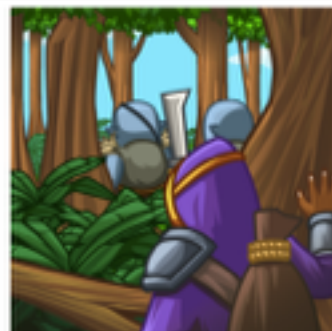
- Archmage Home -
Coders!



- Artisan Home - Builders!



- Adventurer Home -
Testers!



- Scribe Home -
Scribblers!



- Diplomat Home -
Translators!



- Ambassador Home -
Supporters!



CODE COMBAT

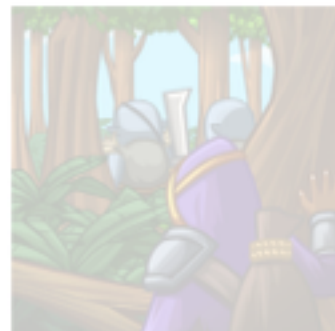
- Archmage Home -
Coders!



- Artisan Home - Builders!



- Adventurer Home -
Testers!



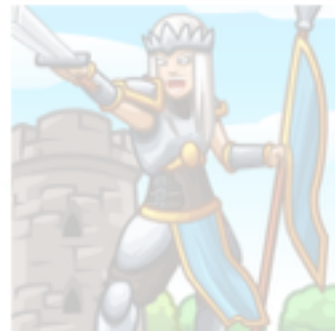
- Scribe Home -
Scribblers!



- Diplomat Home -
Translators!



- Ambassador Home -
Supporters!



CODE COMBAT

- Archmage Home -
Coders!



- Artisan Home - Builders!



- Adventurer Home -
Testers!



- Scribe Home -
Scribblers!



- **Diplomat Home** -
Translators!



- Ambassador Home -
Supporters!



CODE COMBAT

- Archmage Home -
Coders!



- **Artisan Home** - Builders!



- Adventurer Home -
Testers!



- Scribe Home -
Scribblers!



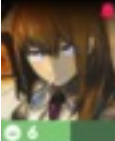
- Diplomat Home -
Translators!



- Ambassador Home -
Supporters!



CodinGame



Onboarding

Best score 100%

SETTINGS

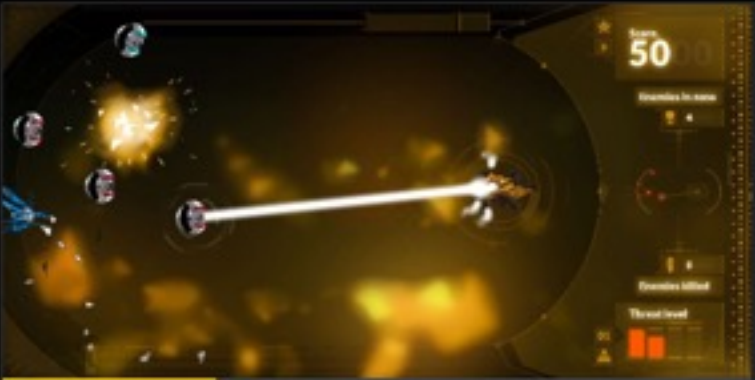
RESULTS

HINTS

FORUM

BACK

FRIENDS



Score 50

Enemies in range 4

Enemies killed 0

Threat level

6/16

The Goal

Your program must destroy the enemy ships by shooting the closest enemy on each

Console output

Game information, Deb...

Standard Output Stream:

> Sectoid

Game information:

Sectoid has been targeted

Threats within range:

HardHat 50m

Buzz 59m

85/16

Standard Output Stream:

> HardHat

Game information:

HardHat has been targeted

86/16

Python3

```
1 import sys
2 import math
3
4 # CodinGame planet is being attacked by slimy insectoid aliens.
5 # <---
6 # Hint: To protect the planet, you can implement the pseudo-code provided in the
7
8
9 # game loop
10 while True:
11     enemy_1 = input() # name of enemy 1
12     dist_1 = int(input()) # distance to enemy 1
13     enemy_2 = input() # name of enemy 2
14     dist_2 = int(input()) # distance to enemy 2
15
16     # Write an action using print
17     # To debug: print("Debug messages...", file=sys.stderr)
18
19     # You have to output a correct ship name to shoot ("Buzz", enemy1, enemy2,
20     if dist_1 < dist_2:
21         print(enemy_1)
22     else:
23         print(enemy_2)
24
```

Test cases

01 Imminent danger

PLAY TESTCASE

Actions

PLAY ALL TESTCASES

SUBMIT

CodinGame

Onboarding

Best score 100%

SETTINGS


RESULTS

HINTS

FORUM

BACK

FRIENDS



The Goal

Your program must destroy the enemy ships by shooting the closest enemy on each

Console output

Game information, Deb...

Standard Output Stream:
> Sectoid
Game information:
Sectoid has been targeted
Threats within range:
HardHat 50e
Buzz 50e
85/16

Standard Output Stream:
> HardHat
Game information:
HardHat has been targeted
86/16

Python3

```
1 import sys
2 import math
3
4 # CodinGame planet is being attacked by alien insectoid aliens.
5 # <---
6 # Hint: To protect the planet, you can implement the pseudo-code provided in the
7
8 # game loop
9 while True:
10     enemy_1 = input() # name of enemy 1
11     dist_1 = int(input()) # distance to enemy 1
12     enemy_2 = input() # name of enemy 2
13     dist_2 = int(input()) # distance to enemy 2
14
15     # Write an action using print
16     # To debug: print("Debug messages...", file=sys.stderr)
17
18     # You have to output a correct ship name to shoot ("Buzz", enemy1, enemy2,
19       print(enemy_1)
20       print(enemy_2)
```

Test cases

01 Imminent danger PLAY TESTCASE

Actions

PLAY ALL TESTCASES

SUBMIT

CodinGame

Onboarding

Best score 100%

SETTINGS

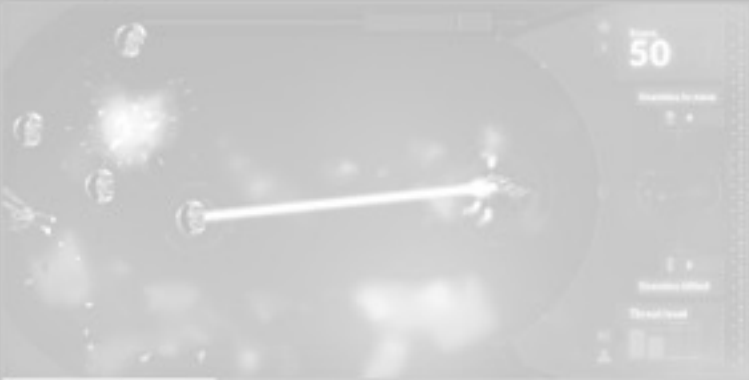
RESULTS

HINTS

FORUM

BACK

FRIENDS



50

6/16

The Goal

Your program must destroy the enemy ships by shooting the closest enemy on each

Console output

Game information, Deb...

Standard Output Stream:
> Sectoid
Game information:
Sectoid has been targeted
Threats within range:
HardHat 50e
Buzz 50e
Standard Output Stream:
> HardHat
Game information:
HardHat has been targeted

85/16

86/16

Python3

```
1 import sys
2 import math
3
4 # CodinGame planet is being attacked by slimy insectoid aliens.
5 # <---
6 # Hint: To protect the planet, you can implement the pseudo-code provided in the
7
8
9 # game loop
10 while True:
11     enemy_1 = input() # name of enemy 1
12     dist_1 = int(input()) # distance to enemy 1
13     enemy_2 = input() # name of enemy 2
14     dist_2 = int(input()) # distance to enemy 2
15
16     # Write an action using print
17     # To debug: print("Debug messages...", file=sys.stderr)
18
19     # You have to output a correct ship name to shoot ("Buzz", enemy1, enemy2,
20     if dist_1 < dist_2:
21         print(enemy_1)
22     else:
23         print(enemy_2)
24
```


Test cases

01 Imminent danger + PLAY TESTCASE

PLAY ALL TESTCASES

SUBMIT

CodinGame



SETTINGS

RESULTS

HINTS

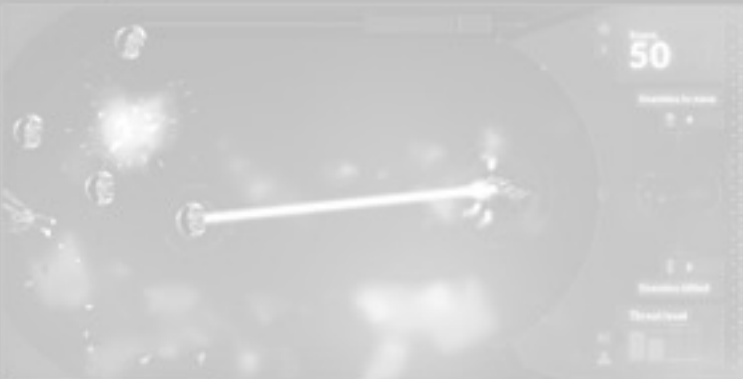
FORUM

BACK

FRIENDS

Onboarding

Best score 100%



6/16

The Goal

Your program must destroy the enemy ships by shooting the closest enemy on each

Console output

Game information, Deb...

Standard Output Stream:

> Sectoid

Game information:

Sectoid has been targeted

Threats within range:

HardHat 50e

Buzz 50e

85/10

Standard Output Stream:

> HardHat

Game information:

HardHat has been targeted

86/10

Python3

```
1 import sys
2 import math
3
4 # CodinGame planet is being attacked by alien insectoid aliens.
5 # <---
6 # Hint: To protect the planet, you can implement the pseudo-code provided in the
7
8
9 # game loop
10 while True:
11     enemy_1 = input() # name of enemy 1
12     dist_1 = int(input()) # distance to enemy 1
13     enemy_2 = input() # name of enemy 2
14     dist_2 = int(input()) # distance to enemy 2
15
16     # Write an action using print
17     # To debug: print("Debug messages...", file=sys.stderr)
18
19     # You have to output a correct ship name to shoot ("Buzz", enemy1, enemy2,
20     if dist_1 < dist_2:
21         print(enemy_1)
22     else:
23         print(enemy_2)
24
```


Test cases

01 Imminent danger ▶ PLAY TESTCASE

▶ PLAY ALL TESTCASES

✓ SUBMIT

CodinGame



SETTINGS

RESULTS


HINTS

FORUM

FRIENDS

Onboarding

Best score 100%



16/16

The Goal

Your program must destroy the enemy ships by shooting the closest enemy on each

Console output

Game Information, Act...

> Hitbot

Game information:

Hitbot has been targeted

Threats within range:

DangerDart 35m

15/16

Standard Output Stream:

> DangerDart

Game information:

Success: You eliminated the threat

No ships remaining.

16/16

Python3

```
1 import sys
2 import math
3
4 # CodinGame planet is being attacked by slimy insectoid aliens.
5 # <====
6 # Hint:To protect the planet, you can implement the pseudo-code provided in the
7
8
9 # game loop
10 while True:
11     enemy_1 = input() # name of enemy 1
12     dist_1 = int(input()) # distance to enemy 1
13     enemy_2 = input() # name of enemy 2
14     dist_2 = int(input()) # distance to enemy 2
15
16     # Write an action using print
17     # To debug: print("Debug messages...", file=sys.stderr)
18
19
20     # You have to output a correct ship name to shoot ("Buzz", enemy1, enemy2,
21     if dist_1 < dist_2:
22         print(enemy_1)
23     else:
24         print(enemy_2)
25
```

Test cases

01 Imminent danger

PLAY TESTCASE

PLAY ALL TESTCASES

SUBMIT



COMMUNITY PUZZLES

1% DONE

Create your own puzzle!

The puzzles below were created by the CodinGame Community. Why don't you create yours? :)



CREATE A PUZZLE NOW



Quarternion Multiplication by TheNinja

SOLVE IT

SOLUTIONS



35 CodinGamers have completed this game
















0%



Queneau Numbers by [CG]VonRickroll






CheckiO




HOME 1%


Welcome home! Take your time getting all pumped up and ready for the journey that's about to begin.

>> Collect points to unlock new stations



You're off to a great start, keep it up and you'll be able to unlock a new station shortly!


10  30


**Non-unique Elements**

Trim an array down to its non-unique elements

structures

10 SOLVED • PUBLISHED • REVIEWED


 Elementary


**Median**

Find the mathematical median in a list of numbers

numbers statistics

SOLVED • PUBLISHED • REVIEWED


 Elementary


**House password**

Check the strength of your favorite password


text

SOLVED • PUBLISHED • REVIEWED

 Elementary











**Moore Neighbourhood**


SOLVED • PUBLISHED • REVIEWED





CheckiO






Median

English (original) ▾

SOLVED • PUBLISHED • REVIEWED



A median is a numerical value separating the upper half of a **sorted** array of numbers from the lower half. In a list where there are an odd number of entities, the median is the number found in the middle of the array. If the array contains an even number of entities, then there is no single middle value, instead the median becomes the average of the two numbers found in the middle. For this mission, you are given a non-empty array of natural numbers (X). With it, you must separate the upper half of the numbers from the lower half and find the median.

Input: An array as a list of integers.

Output: The median as a float or an integer.





Example:

HOME

0 0 0

Story

- ✓ Solve it
- Discuss
- Timeline
- All Solutions
- Random
- Python 3.3
- ➔ Get next task



<http://checkio.org/mission/mv>

Users attempted:	31578
Users succeeded:	25767



CheckiO

Python 3.3

Median

help

Save Run & Check Try it Reset code Task... Use local editor

```
1 def checkio(data):
2
3     #replace this for solution
4     return data[0]
5
6 #These "asserts" using only for self-checking and not necessary for auto-testing
7 if __name__ == '__main__':
8     assert checkio([1, 2, 3, 4, 5]) == 3, "Sorted list"
9     assert checkio([1, 1, 2, 3, 3]) == 3, "Not sorted list"
10    assert checkio([1, 100, 2, 200, 1]) == 2, "It's not an average"
11    assert checkio([3, 6, 20, 99, 10, 15]) == 12.5, "Even length"
12    print("Start the long test")
13    assert checkio(list(range(1000000))) == 499999.5, "Long."
14    print("The local tests are done.")
```

Check results

A median is a numerical value separating the upper half of a **sorted** array of numbers from the lower half. In a list where there are an odd number of entries, the median is the number found in the middle of the array. If the array contains an even number of entries, then there is no single middle value, instead the median becomes the average of the two numbers found in the middle. For this mission, you are given a non-empty array of natural numbers (x). With it, you must separate the upper half of the numbers from the lower half and find the median.

Input: An array as a list of integers.

Output: The median as a float or an integer.

Precondition:

$1 < \text{len}(\text{data}) \leq 1000$

$\text{all}(0 \leq x < 10^{**6} \text{ for } x \text{ in data})$

LianaBakradze
I have no idea how to start solving this mission

bryukh
As first you need to define the length of the array. You can use the built-in function `len()`.

LianaBakradze
I need some help to proceed with the mission

bryukh
With the builtin operator "modulus" you can define if a number even or odd.

```
if n % 2 == 1: # odd
```

LianaBakradze
I am gone half way through. Need help

bryukh
If you are using python3, then don't forget that "/" is a real division, "//" is an integer division.

LianaBakradze
I am stuck. I need a small hint.

bryukh
Don't forget to sort the array.

Output (Python console)

Click on "Run Code" to view results or Ctrl + /
Click on "Save" to save your code or Ctrl + S

Find median

1 10 2 9 3 8 4 7 5 6



CheckiO

Python 3.3

Median

help

Save Run & Check Try it Reset code Task... Use local editor

```
1 def checkio(data):
2
3     #replace this for solution
4     return data[0]
5
6 #These "asserts" using only for self-checking and not necessary for auto-testing
7 if __name__ == '__main__':
8     assert checkio([3, 2, 3, 4, 5]) == 3, "Sorted list"
9     assert checkio([3, 1, 2, 5, 3]) == 3, "Not sorted list"
10    assert checkio([3, 300, 2, 200, 1]) == 2, "It's not on average"
11    assert checkio([3, 6, 20, 99, 10, 15]) == 12.5, "Even length"
12    print("Start the long test")
13    assert checkio(list(range(1000000))) == 499999.5, "Long."
14    print("The local tests are done.")
```

Check results

A median is a numerical value separating the upper half of a **sorted** array of numbers from the lower half. In a list where there are an odd number of entries, the median is the number found in the middle of the array. If the array contains an even number of entries, then there is no single middle value, instead the median becomes the average of the two numbers found in the middle. For this mission, you are given a non-empty array of natural numbers (x). With it, you must separate the upper half of the numbers from the lower half and find the median.

Input: An array as a list of integers.

Output: The median as a float or an integer.

Precondition:

$1 < \text{len}(\text{data}) \leq 1000$

$\forall 0 \leq x < 10^6 \text{ for } x \text{ in data}$

Output (Python console)

Click on "Run Code" to view results or Ctrl + /

Click on "Save" to save your code or Ctrl + S

1 10 2 9 3 8 4 7 5 6

Find median

LianaBakradze

I have no idea how to start solving this mission.

bryukh

As first you need to define the length of the array. You can use the built-in function `len()`.

LianaBakradze

I need some help to proceed with the mission.

bryukh

With the builtin operator "modulus" you can define if a number even or odd.

```
if n % 2 == 1: # odd
```

LianaBakradze

I am gone half way through. Need help!

bryukh

If you are using python3, then don't forget that "/" is a real division, "//" is an integer division.

LianaBakradze

I am stuck. I need a small hint.

bryukh

Don't forget to sort the array.



CheckiO

CheckiO Python 3.3 Median ? Help LianaBakradze

Save Run & Check Try It Reset code Task... Use local editor

```
1 def checkio(data):
2
3     #replace this for solution
4     return data[0]
5
6 #These "asserts" using only for self-checking and not necessary for auto-testing
7 if __name__ == '__main__':
8     assert checkio([1, 2, 3, 4, 5]) == 3, "Sorted list"
9     assert checkio([3, 1, 2, 5, 3]) == 3, "Not sorted list"
10    assert checkio([1, 300, 2, 200, 1]) == 2, "It's not an average"
11    assert checkio([3, 6, 20, 99, 10, 15]) == 12.5, "Even length"
12    print("Start the long test")
13    assert checkio(list(range(1000000))) == 499999.5, "Long."
14    print("The local tests are done.")
```

Check results

A median is a numerical value separating the upper half of a **sorted** array of numbers from the lower half. In a list where there are an odd number of entities, the median is the number found in the middle of the array. If the array contains an even number of entities, then there is no single middle value, instead the median becomes the average of the two numbers found in the middle. For this mission, you are given a non-empty array of natural numbers (X). With it, you must separate the upper half of the numbers from the lower half and find the median.

Input: An array as a list of integers.

Output: The median as a float or an integer.

Preconditions:

- 1 < len(data) ≤ 1000
- all(0 ≤ x < 10 ** 6 for x in data)

Output (Python console)

Click on "Run Code" to view results or Ctrl + /

Click on "Save" to save your code or Ctrl + S

1 10 2 9 3 8 4 7 5 6

Find median

LianaBakradze

I have no idea how to start solving this mission

bryukh

As first you need to define the length of the array. You can use the built-in function `len()`.

LianaBakradze

I need some help to proceed with the mission

bryukh

With the builtin operator "modulus" you can define if a number even or odd.

```
if n % 2 == 1: # odd
```

LianaBakradze

I am gone half way through. Need help!

bryukh

If you are using python3, then don't forget that "/" is a real division, "//" is an integer division.

LianaBakradze

I am stuck. I need a small hint.

bryukh

Don't forget to sort the array.



CheckiO

Python 3.3 - Median - ? help


Save Run & Check Try it Reset code Task... Use local editor

```
1 def checkio(data):
2
3     #replace this for solution
4     return data[0]
5
6 #These "asserts" using only for self-checking and not necessary for auto-testing
7 if __name__ == '__main__':
8     assert checkio([1, 2, 3, 4, 5]) == 3, "Sorted list"
9     assert checkio([3, 1, 2, 5, 3]) == 3, "Not sorted list"
10    assert checkio([1, 300, 2, 200, 1]) == 2, "It's not on average"
11    assert checkio([3, 6, 20, 99, 10, 15]) == 12.5, "Even length"
12    print("Start the long test")
13    assert checkio(list(range(1000000))) == 499999.5, "Long."
14    print("The local tests are done.")
```

Output (Python console)

Click on "Run Code" to view results or Ctrl + /
Click on "Save" to save your code or Ctrl + S

Find median



Check results

A median is a numerical value separating the upper half of a **sorted** array of numbers from the lower half. In a list where there are an odd number of entries, the median is the number found in the middle of the array. If the array contains an even number of entries, then there is no single middle value, instead the median becomes the average of the two numbers found in the middle. For this mission, you are given a non-empty array of natural numbers (X). With it, you must separate the upper half of the numbers from the lower half and find the median.

Input: An array as a list of integers.
Output: The median as a float or an integer.

Precondition:
 $1 < \text{len}(\text{data}) \leq 1000$
 $0 \leq x < 10^6$ for x in data

LianaBekradze
I have no idea how to start solving this mission

bryukh
As first you need to define the length of the array. You can use the built-in function `len()`.

LianaBekradze
I need some help to proceed with the mission

bryukh
With the builtin operator "modulus" you can define if a number even or odd.

```
if n % 2 == 1: # odd
```

LianaBekradze
I am gone half way through. Need help!

bryukh
If you are using python3, then don't forget that "/" is a real division, "//" is an integer division.

LianaBekradze
I am stuck. I need a small hint.

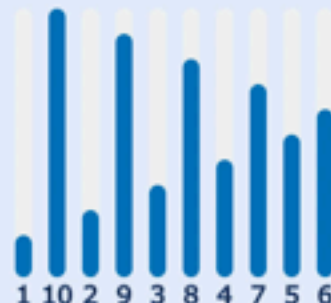
bryukh
Don't forget to sort the array.

```

1 def checkio(data):
2     #replace this for solution
3     return data[0]
4
5 #These "asserts" using only for self-checking and not necessary for auto-testing
6 if __name__ == '__main__':
7     assert checkio([1, 2, 3, 4, 5]) == 3, "Sorted list"
8     assert checkio([3, 1, 2, 5, 3]) == 3, "Not sorted list"
9     assert checkio([1, 300, 2, 200, 1]) == 2, "It's not an average"
10    assert checkio([3, 6, 20, 99, 10, 15]) == 12.5, "Even length"
11    print("Start the long test")
12    assert checkio(list(range(1000000))) == 499999.5, "Long."
13    print("The local tests are done.")
    
```

Output (Python console)

Click on "Run Code" to view results or Ctrl + /
 Click on "Save" to save your code or Ctrl + S



Find median



CheckiO

CheckiO / checkio-mission-template

Watch

6

Star

9

Fork

42

Code

Issues

Pull requests

Pulse

Graphs

Base CheckiO template for users task

79 commits

1 branch

0 releases

3 contributors

Branch: master

New pull request

Find file

Clone or download



Bryukh Change jscolors

Latest commit 459991f on Jan 14, 2015

editor

Change jscolors

2 years ago

hints

typo

2 years ago

info

Change jscolors

2 years ago

verification

small referee tests adjustments

3 years ago

.gitignore

Fix error handling with new data

3 years ago

README.md

Update README.md

3 years ago

README.md

checkio-task-template

Conclusion

- 3 great projects
- try yourself
- contribute
- invent something cool



Thank You!

Liana Bakradze

liana.bakradze@jetbrains.com
@med_vector