Overview
Deployment
Accounts
Pages
Notifications
Challenges
Flags
Custom Challenges
Custom Challenges
Dynamic Value
Multiple Choice
Manual Verification
Code Challenges
King of the Hill
Application Target
Management
Integrations
Scoring

Custom Challenges › Dynamic Value

Dynamic Value

A dynamic value challenge is a challenge whose point value decreases after each solve. By reducing the value of the challenge on each solve, all users who have previously solved the challenge will have lowered scores. Thus an easier and more solved challenge will naturally have a lower point value than a harder and less solved challenge.

> **INFO**
>
> Hidden users and teams will **not** affect the scoring of dynamic value challenges. If a hidden user or team (e.g. an admin) solves a dynamic value challenge, the value for other participants will remain unchanged.

Within CTFd you are free to mix and match regular and dynamic challenges.

The current implementation requires the challenge to keep track of three values:

- Initial - The original point valuation
- Decay - The amount of solves before the challenge will be at the minimum
- Minimum - The lowest possible point valuation

The value decay logic is implemented with the following math:

$$a = \text{max points}$$

$$b = \text{min points}$$

$$s = \text{solve threshold}$$

$$f(x) = \frac{b-a}{s^2}x^2 + a$$

or in pseudo code:

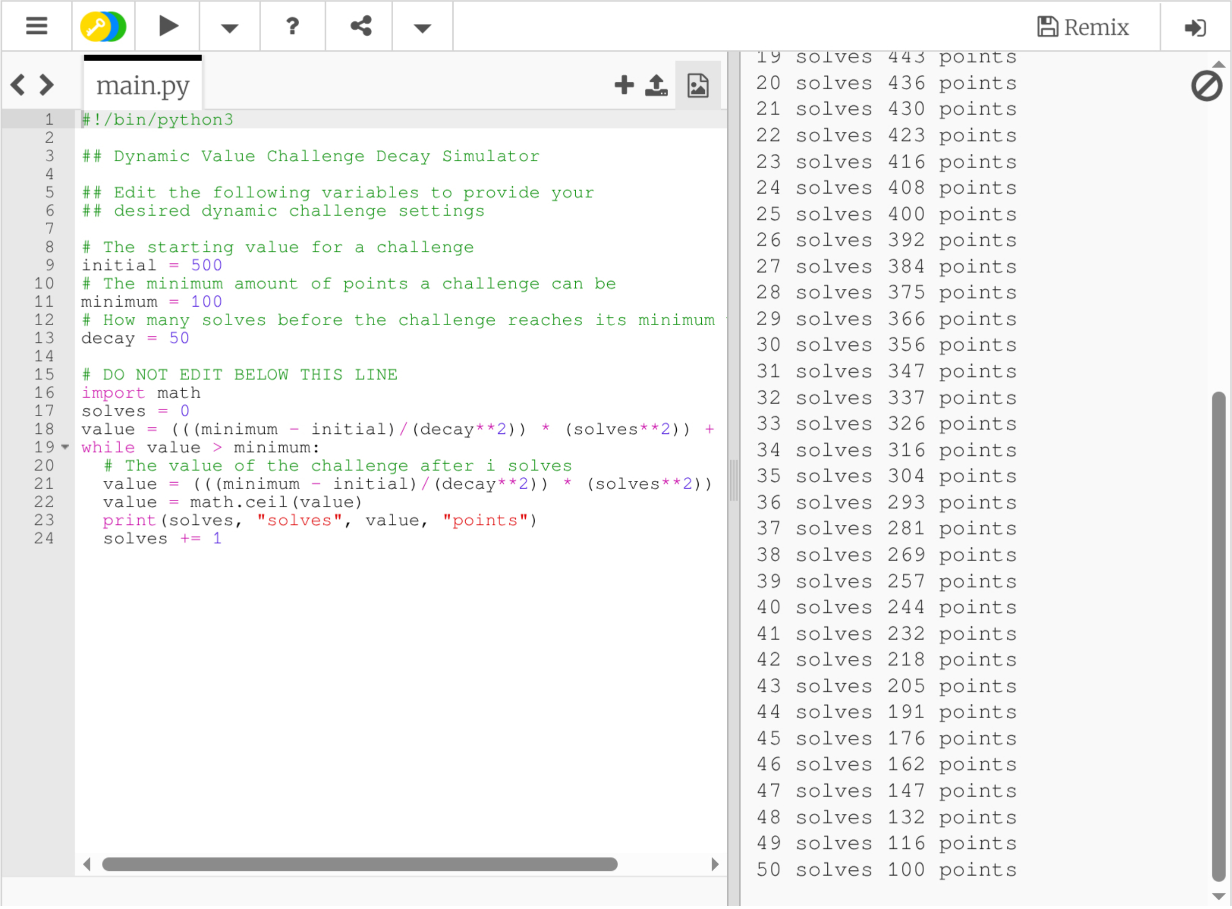
```
value = (((minimum - initial) / (decay ** 2)) * (solve_count ** 2)) + initial
value = math.ceil(value)
```

If the number generated is lower than the minimum, the minimum is chosen instead.

CTFd uses a parabolic function instead of a linear, exponential, or logarithmic decay function so that higher valued challenges have a slower drop from their initial value.

Value Decay Simulator

Below is a simulator that allows you to test out the decay function variables of your dynamic value challenge.



```
#!/bin/python3

## Dynamic Value Challenge Decay Simulator

## Edit the following variables to provide your
## desired dynamic challenge settings

# The starting value for a challenge
initial = 500
# The minimum amount of points a challenge can be
minimum = 100
# How many solves before the challenge reaches its minimum
decay = 50

# DO NOT EDIT BELOW THIS LINE
import math
solves = 0
value = (((minimum - initial)/(decay**2)) * (solves**2)) +
while value > minimum:
  # The value of the challenge after i solves
  value = (((minimum - initial)/(decay**2)) * (solves**2))
  value = math.ceil(value)
  print(solves, "solves", value, "points")
  solves += 1
```

```
19 solves 443 points
20 solves 436 points
21 solves 430 points
22 solves 423 points
23 solves 416 points
24 solves 408 points
25 solves 400 points
26 solves 392 points
27 solves 384 points
28 solves 375 points
29 solves 366 points
30 solves 356 points
31 solves 347 points
32 solves 337 points
33 solves 326 points
34 solves 316 points
35 solves 304 points
36 solves 293 points
37 solves 281 points
38 solves 269 points
39 solves 257 points
40 solves 244 points
41 solves 232 points
42 solves 218 points
43 solves 205 points
44 solves 191 points
45 solves 176 points
46 solves 162 points
47 solves 147 points
48 solves 132 points
49 solves 116 points
50 solves 100 points
```

Direct Link

Previous
« Custom Challenges

Next
Multiple Choice »

Was this page helpful? 👍 👎

Share your feedback