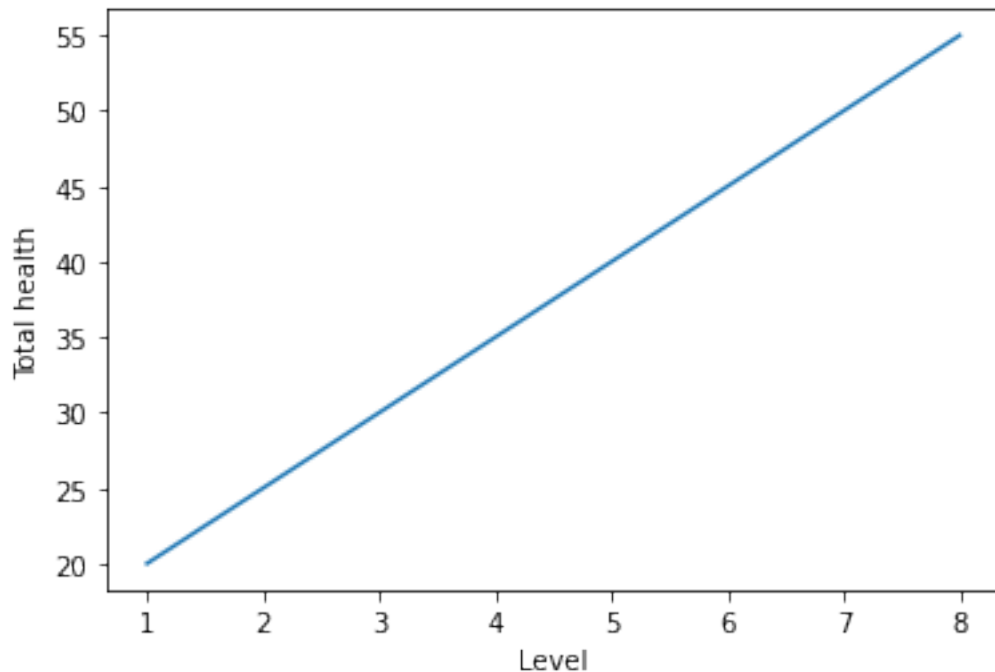


GameBalanceFinal

December 5, 2021

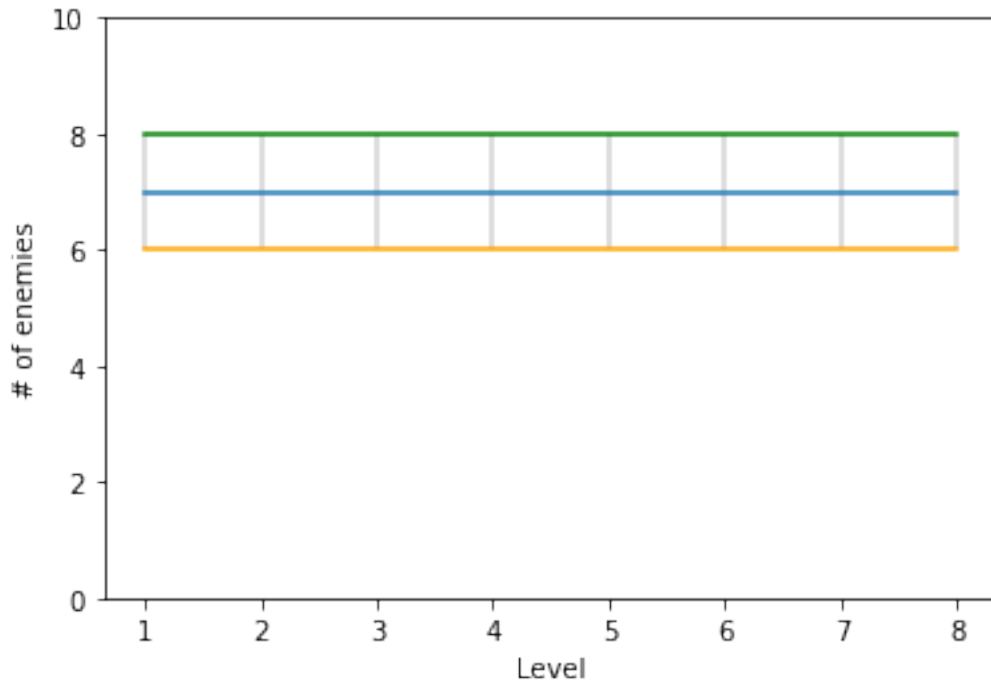
```
[ ]: import numpy as np
import matplotlib.pyplot as plt
from matplotlib.ticker import MaxNLocator

[ ]: # data
levels = np.array([1,2,3,4,5,6,7,8])
plt.xlabel("Level")
plt.ylabel("Total health")
plt.plot(levels, [20 + 5 * (x-1) for x in levels])
plt.show()
```



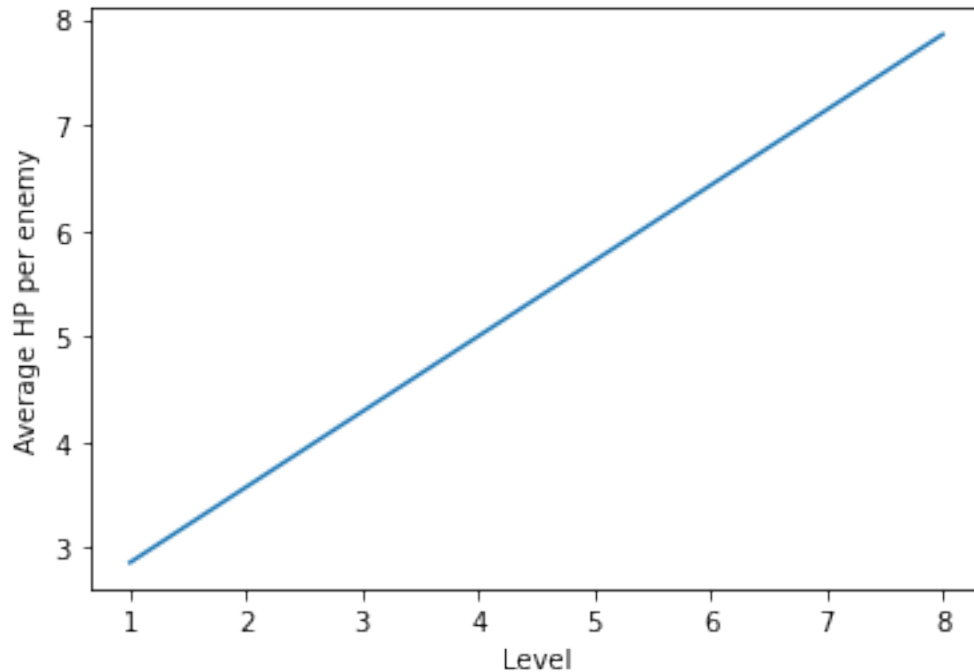
Balancing total health with respect to level. We find that twenty hp for the first level seems appropriate. After each level, the players are capable of purchasing two power up's. To balance the player's increase in power level, we add five hp per subsequent level.

```
[ ]: levels = np.array([1,2,3,4,5,6,7,8])
plt.xlabel("Level")
plt.ylabel("# of enemies")
plt.ylim(0,10)
plt.errorbar(levels, [7 for x in levels], yerr = -1, ecolor='lightgray')
plt.plot(levels, [8 for x in levels], color='green')
plt.plot(levels, [6 for x in levels], color='orange')
plt.show()
```



We want to keep number of enemies per level to be around six to eight. This decision is made mainly from the size of our battle room and the size of the players and enemies, seven seems like an appropriate number of enemies after qualitative discussion. We want the freedom of the range of six to eight because enemies don't have the same difficulty.

```
[ ]: !apt-get install texlive texlive-xetex texlive-latex-extra pandoc
      !pip install py pandoc
levels = np.array([1,2,3,4,5,6,7,8])
plt.xlabel("Level")
plt.ylabel("Average HP per enemy")
plt.plot(levels, [(20 + 5 * (x-1))/7 for x in levels])
plt.show()
```



Here we have the average hp per enemy, assuming we have seven enemies. This is to help us with balancing the hp of each individual enemy. We have many types of enemy and we want to set each to a set hp and not change it. So this helps us balance each level with a combination of easy and hard enemies.

Other types of balancing include tweaking the projectile speed of players and enemies to account for the zoom delay. Increased players' projectile speed to reduce aiming precision and decreased enemies' projectile speed to make it easier for player to dodge.

We also reduced enemies' movement speed to make it easier for players to dodge the enemies.

We balanced each of our power up's to have roughly the same power level and all cost five. We did this to simplify the process of incrementally increasing level difficulty. The players' power level grows just as the level's difficulty grow.

We balance level layout by having both easy enemies with low hp and hard enemies with high hp. We balanced each of our level individually to make sure the difficulty feels right.

The final boss is balanced differently from the other levels, we did a lot of play testing to make sure the boss feels both challenging and fair.

```
[8]: !jupyter nbconvert --to pdf "GameBalance.ipynb"
```

```
[NbConvertApp] Converting notebook GameBalance.ipynb to pdf
[NbConvertApp] Support files will be in GameBalance_files/
[NbConvertApp] Making directory ./GameBalance_files
[NbConvertApp] Making directory ./GameBalance_files
[NbConvertApp] Making directory ./GameBalance_files
[NbConvertApp] Writing 54947 bytes to ./notebook.tex
[NbConvertApp] Building PDF
[NbConvertApp] Running xelatex 3 times: [u'xelatex', u'./notebook.tex',
```

```
'-quiet']  
[NbConvertApp] Running bibtex 1 time: [u'bibtex', u'./notebook']  
[NbConvertApp] WARNING | bibtex had problems, most likely because there were no  
citations  
[NbConvertApp] PDF successfully created  
[NbConvertApp] Writing 59799 bytes to GameBalance.pdf
```

[]: