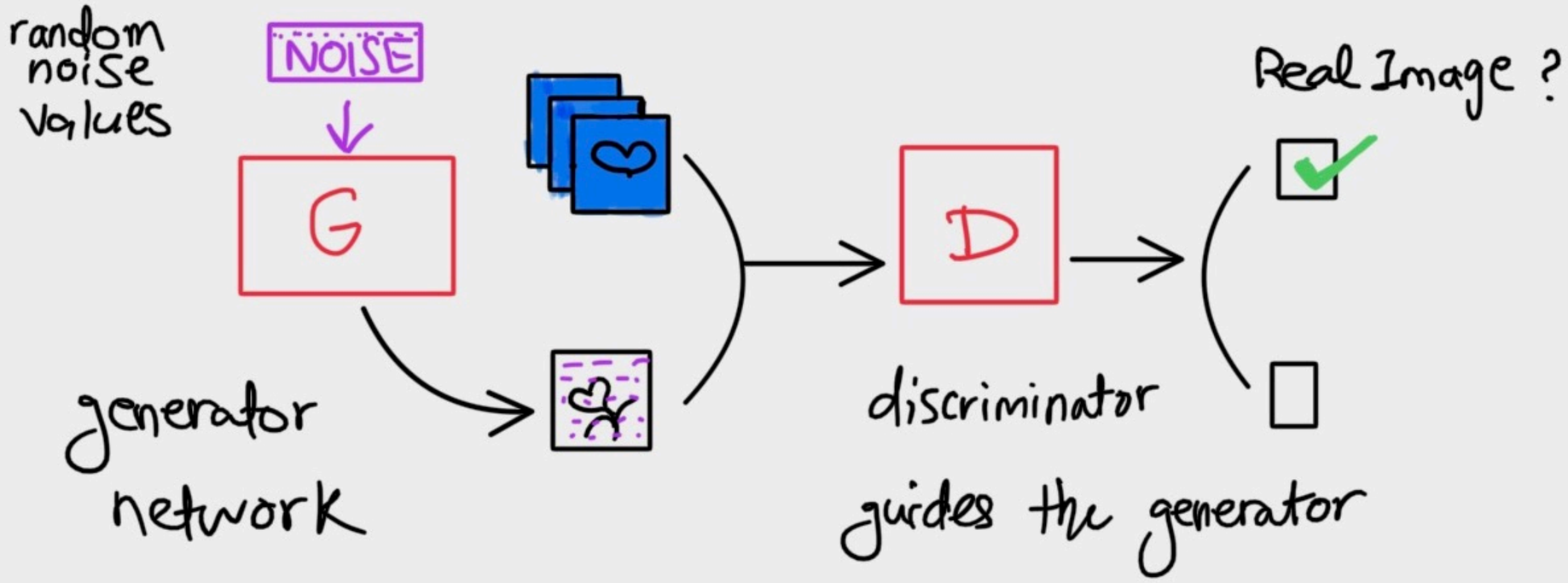
Generative Adversarial Networks — Overview

- How do GANs work?



* "D" is shown real images half of the time, and fake "gen" images the other half. "D" is trained to output close to 1 prob. for real images.
Meanwhile, "G" tries to generate images that "D" will classify as real.
So, overtime, "G" learns to produce more real images in order to fool "D".

G and D are at a competition.

* choosing a good architecture

* lrelu = leaky relu

«a simple GAN»

D and G should have at least 1 hidden layer

- In GANs there are two optimizations:

```
optim.Adam(D.parameters(), lr)
optim.Adam(G.parameters(), lr)
```

 → D → logits → sigmoid → prob 0/1 (binary classification task)

```
d_loss = nn.BCEWithLogitsLoss(logits, label * 0.9)
```

numerically stable cross-entropy

* it help to generalize better

```
g_loss = nn.BCEWithLogitsLoss(logits, flipped_labels)
```

Alternative GAN Architecture