

Brute force solution:

→ Have nested loop that loops n^2 times in worst case

```
for i in range(nums.length)
  for j in range(nums.length)
    if nums[i] + nums[j] == target
      return [i, j]
```

Efficient solution:

```
h = {}
```

```
for i in range(nums.length)
```

```
  h[nums[i]] = i
```

↳ creates a hashmap that links
the value to the index

i.e. [3, 3, 8] → {(3:0), (3:1), (8:2)}

```
for i in range(nums.length)
```

```
  comp = target - nums[i]
```

```
  if comp in h and h[comp] != i
```

```
    return [i, h[comp]]
```

```
return [inf, inf]
```