

```
In [2]: co_prefs = { 'A' : "abcd", 'B': "cadb", 'C': "acdb", 'D': "adbc"}
st_prefs = { 'a' : "CDAB", 'b': 'CDAB', 'c' : "DBAC" , 'd': "ABCD"}
```

```
In [4]: def stable(matching):
    for co in co_prefs.keys():
        for st in st_prefs.keys():
            # if st prefers co to its current match
            # and co prefers st to its current match
            if st_prefs[st].index(co) < st_prefs[st].index(matching[st
]) and \
                co_prefs[co].index(st) < co_prefs[co].index(matching[c
o]):
                print("unstable", co, st)
            return False
    return True
```

```
In [5]: stable({'A':'a', 'B':'b', 'C':'c', 'D':'d', 'a':'A', 'b':'B', 'c':'C', 'd':'D'
'})
```

unstable B c

```
Out[5]: False
```

```
In [14]: def gale_shapley():
# start out with empty matching
matching = {}
offers = { co : set() for co in co_prefs.keys() }
# loop until not everyone is matched
while len(matching) < len(co_prefs) + len(st_prefs):
# find an unmatched company
unmatched_co = next(co for co in co_prefs.keys()
                    if co not in matching)
# offer to the first student who
# we have not offered to before
offer_st = next(st for st in co_prefs[unmatched_co]
                if st not in offers[unmatched_co])
print(f"Offer from {unmatched_co} to {offer_st}")
offers[unmatched_co].add(offer_st)
if offer_st not in matching:
# student is unmatched, accept
print("Accepted (unmatched)")
matching[unmatched_co] = offer_st
matching[offer_st] = unmatched_co
else:
current_co = matching[offer_st]
# is this an improvement for the student?
if st_prefs[offer_st].index(current_co) > \
   st_prefs[offer_st].index(unmatched_co):
print(f"Accepted (rejecting {current_co})")
# current_co is now unmatched
del matching[current_co]
matching[unmatched_co] = offer_st
matching[offer_st] = unmatched_co
else:
print(f"Rejected, {offer_st} likes {current_co}
better")
# do nothing
return matching
```

```
In [2]: co_prefs = { 'A' : "abcd", 'B' : "cadb", 'C' : "acdb", 'D' : "adbc" }
st_prefs = { 'a' : "CDAB", 'b' : 'CDAB', 'c' : "DBAC" , 'd' : "ABCD" }
```

```
In [15]: m = gale_shapley()  
m
```

```
Offer from A to a  
Accepted (unmatched)  
Offer from B to c  
Accepted (unmatched)  
Offer from C to a  
Accepted (rejecting A)  
Offer from A to b  
Accepted (unmatched)  
Offer from D to a  
Rejected, a likes C better  
Offer from D to d  
Accepted (unmatched)
```

```
Out[15]: {'a': 'C',  
          'B': 'c',  
          'c': 'B',  
          'C': 'a',  
          'A': 'b',  
          'b': 'A',  
          'D': 'd',  
          'd': 'D'}
```

```
In [16]: stable(m)
```

```
Out[16]: True
```