

---

# 알고리즘 실습

— 181204 - Stable Marriage —

---

# 오늘의 목표

- Stable Marriage Matching

# Feedback

# 지난 시간: LCS 응용

- 제출률: .%(/41)
- 질문: 이메일 건, 연구실 방문 건

# 안정 매칭 문제

# 안정 매칭: 조건과 목표

- A그룹, B그룹 서로에 대해서 모두 알아야 함
- 상대 그룹 멤버에 대하여 '선호 순위'가 있어야 함
- 매칭이 완료된 후에는 변화가 없어야 함
- 현재의 파트너보다 다른 파트너가 더 높은 순위(서로!!)

# 안정 매칭: 구현

- Code wiki: [https://rosettacode.org/wiki/Stable\\_marriage\\_problem](https://rosettacode.org/wiki/Stable_marriage_problem)
- Blog:  
<https://www.vitoshacademy.com/python-algorithms-stable-matching-problem/>
- Stackexchange:  
<https://codereview.stackexchange.com/questions/186181/performance-of-stable-marriage-solution-in-python-3>
- (심지어) 3rd party 도구: <https://pypi.org/project/matching/>

# SMP 구현 예제

- TA가 생각하기에 가장 쉬운 버전

```
56 def matchmaker():
57     guysfree = guys[:]
58     engaged = {}
59     guyprefers2 = copy.deepcopy(guyprefers)
60     galprefers2 = copy.deepcopy(galprefers)
61     while guysfree:
62         guy = guysfree.pop(0)
63         guyslist = guyprefers2[guy]
64         gal = guyslist.pop(0)
65         fiance = engaged.get(gal)
66         if not fiance:
67             # She's free
68             engaged[gal] = guy
69             print(" %s and %s" % (guy, gal))
70         else:
71             # The bounder proposes to an engaged lass!
72             galslist = galprefers2[gal]
73             if galslist.index(fiance) > galslist.index(guy):
74                 # She prefers new guy
75                 engaged[gal] = guy
76                 print(" %s dumped %s for %s" % (gal, fiance, guy))
77                 if guyprefers2[fiance]:
78                     # Ex has more girls to try
79                     guysfree.append(fiance)
80             else:
81                 # She is faithful to old fiance
82                 if guyslist:
83                     # Look again
84                     guysfree.append(guy)
85     return engaged
```



# More...

- csv로 입력을 받아서...

```
1 group,name,pref
2 A,초코,우유-모카-스무디
3 A,카페,모카-스무디-우유
4 A,딸기,우유-스무디-모카
5 B,스무디,딸기-초코-카페
6 B,모카,카페-초코-딸기
7 B,우유,카페-초코-딸기
```

```
1 import csv
2
3 from person import Person
4
5
6 def get_pref_list(file_path):
7     prefs = dict()
8     with open(file_path, 'r') as f:
9         reader = csv.DictReader(f)
10        for row in reader:
11            group = row['group']
12            name = row['name']
13            if group not in prefs.keys():
14                prefs[group] = dict()
15            prefs[group][name] = Person(row)
16        return prefs
```

# Class Person

- 사람을 객체로!
  - 이름, 선호 순위 등을  
멤버 변수로 저장
- list of object를 인자  
(인수)로 받아 SMP  
수행

```
1 class Person(object):
2     def __init__(self, row):
3         self.group = row['group']
4         self.name = row['name']
5         self.pref = row['pref'].split('-')
```

```
43 def main():
44     prefs = get_pref_list(FILE_NAME)
45     smp(prefs['A'], prefs['B'])
46
47     for key in prefs['A']:
48         print(prefs['A'][key])
49     for key in prefs['B']:
50         print(prefs['B'][key])
51
52     print(is_stable(prefs['A'], prefs['B']))
```

```
['초코', '카페', '딸기'] ['스무디', '모카', '우유']
Name: 초코, Partner: 우유
Name: 카페, Partner: 모카
Name: 딸기, Partner: 스무디
Name: 스무디, Partner: 딸기
Name: 모카, Partner: 카페
Name: 우유, Partner: 초코
True
```

# 기타 유용한 정보

# 출석부 및 실습 점수가 궁금하다면?

- 출석부 및 실습 채점표
  - 수업 시작 후 30분까지 지각, 이후 결석
  - 실습 딜레이 1일당: -2점
    - 딜레이 2일까지: -2
    - 이후 -1씩 추가
- 튜터의 테스트 결과

# 질문이 생기면?

- 이름: 문현수
- 전공: 통신및보안
- 과정: 석박사통합과정 8학기
- 연구실: 데이터네트워크연구실(공5633)
- 메일: [munhyunsu@cs-cnu.org](mailto:munhyunsu@cs-cnu.org)
- 알고리즘은 함께 해결해가는 과목이므로 과감하게 연락
- 이메일로 처리가 안 되는 급한일: 문자/전화 등