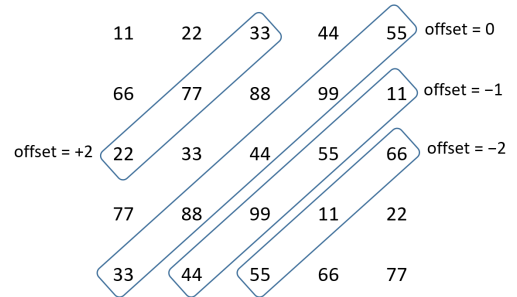# ICA-4 Solutions

Problem 1

```
def sum_diag_UR_LL(grid, offset):
    sum = 0
    if offset > 0:
        i,j = 0,(len(grid)-1)-offset
    else:
        i,j = -offset,(len(grid)-1)

    while 0 <= i < len(grid) and 0 <= j <
len(grid):
        sum += grid[i][j]
        i, j = i+1, j-1

    return sum
```

The diagram to the right shows a grid of numbers with diagonal groupings labeled:

11	22	33	44	55	offset = 0
66	77	88	99	11	offset = –1
22	33	44	55	66	offset = –2 (offset = +2 on left)
77	88	99	11	22	
33	44	55	66	77	

Problem 2

```
def num_keys(d):
    keylist = list(d.keys())
    nkeys = len(keylist)
    for k in keylist:
        nkeys += len(list(d[k].keys()))

    return nkeys
```

Problem 3

```
def process_input(fname):
    sfile = open(fname)
    students = {}
    for line in sfile:
        if line[0] == "#":                  # ignore the header line
            continue
        student = line.strip().split(",")
        name = student[0]
        course_grades = []
        for c_pair in student[1:]:          # loop thru course/grades
            parts = c_pair.split(":")       # separate on colon
            course_grades.append([parts[0], parts[1]])
        students[name] = course_grades
    return students
```

Problem 4

```python
def create_rdict(rlist):
    rdict = {}
    for rec in rlist:
        rdict[rec[0]] =[rec[1], rec[2], rec[1] * rec[2]]
    return rdict

def print_greatest_area(rdict):
    area_l = []
    max = 0

    for key in rdict:
        rvalues = rdict[key]
        if rvalues[2] > max:
            area_l = [key]
            max = rvalues[2]
        elif rvalues[2] == max:
            area_l.append(key)

    for rec in area_l:
        rvalues = rdict[rec]
        print(rec, ":", rvalues)
```

Problem 5

```python
def pair_frequencies(word_list):
    freq_dict = {}
    for word in word_list:
        if len(word) < 2:   #skip words of length 1
            continue
        for i in range(len(word)-1):
            pair = word[i:i+2]
            print(pair)
            if not pair in freq_dict:
                freq_dict[pair] = 0
            freq_dict[pair] += 1
            print(freq_dict)
    for pair in freq_dict:
        print(pair, ":", freq_dict[pair])
```