

## السؤال الأول:

في البداية قمنا بتعريف قائمة بأسماء الطلاب المتخرجين، في السطر الثاني طلبنا من المستخدم إدخال اسم الطالب الذي يريد معرفة حالته (متخرج أم لا).

في حلقة for يتم جلب جميع العناصر بالقائمة ثم اختبارها مع قيمة اسم الطالب، في حال تساوت قيمة الاسم من القائمة مع الاسم المدخل يطبع عبارة تدل على أن الطالب متخرج، في حال انتهاء الحلقة دون تنفيذ break يتم تنفيذ التعليمة في السطر الأخير وهي طباعة عبارة تدل على أن الطالب غير متخرج.

Q1:

```
graduated_students = ['Mohammad', 'Anas', 'Khaled', 'Ali', 'Osama']
student_name = input('Enter student name:')
for student in graduated_students:
    if student == student_name:
        print('Yes, you are graduated.')
        break
    else:
        print('No, you are not graduated.')
```

عند التنفيذ:

```
Enter student name:Anas
Yes, you are graduated.
```

```
Enter student name:Amjad
No, you are not graduated.
```

Q2:

التعليمة الأولى تولد أعداد ضمن المجال [1,1001] والشرط هو لتوليد فقط الأعداد التي باقي قسمتها على 2 لا يساوي الصفر (أي الأعداد الفردية فقط).

```
odd_list = [num for num in range(1,1001) if num % 2 != 0]
print(odd_list)
```

عند التنفيذ:

```
[1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39, 41, 43, 45, 47, 49, 51, 53, 55, 57, 59, 61, 63, 65, 67, 69, 71, 73, 75, 77, 79, 81, 83, 85, 87, 89, 91, 93, 95, 97, 99, 101, 103, 105, 107, 109, 111, 113, 115, 117, 119, 121, 123, 125, 127, 129, 131, 133, 135, 137, 139, 141, 143, 145, 147, 149, 151, 153, 155, 157, 159, 161, 163, 165, 167, 169, 171, 173, 175, 177, 179, 181, 183, 185, 187, 189, 191, 193, 195, 197, 199, 201, 203, 205, 207, 209, 211, 213, 215, 217, 219, 221, 223, 225, 227, 229, 231, 233, 235, 237, 239, 241, 243, 245, 247, 249, 251, 253, 255, 257, 259, 261, 263, 265, 267, 269, 271, 273, 275, 277, 279, 281, 283, 285, 287, 289, 291, 293, 295, 297, 299, 301, 303, 305, 307, 309, 311, 313, 315, 317, 319, 321, 323, 325, 327, 329, 331, 333, 335, 337, 339, 341, 343, 345, 347, 349, 351, 353, 355, 357, 359, 361, 363, 365, 367, 369, 371, 373, 375, 377, 379, 381, 383, 385, 387, 389, 391, 393, 395, 397, 399, 401, 403, 405, 407, 409, 411, 413, 415, 417, 419, 421, 423, 425, 427, 429, 431, 433, 435, 437, 439, 441, 443, 445, 447, 449, 451, 453, 455, 457, 459, 461, 463, 465, 467, 469, 471, 473, 475, 477, 479, 481, 483, 485, 487, 489, 491, 493, 495, 497, 499, 501, 503, 505, 507, 509, 511, 513, 515, 517, 519, 521, 523, 525, 527, 529, 531, 533, 535, 537, 539, 541, 543, 545, 547, 549, 551, 553, 555, 557, 559, 561, 563, 565, 567, 569, 571, 573, 575, 577, 579, 581, 583, 585, 587, 589, 591, 593, 595, 597, 599, 601, 603, 605, 607, 609, 611, 613, 615, 617, 619, 621, 623, 625, 627, 629, 631, 633, 635, 637, 639, 641, 643, 645, 647, 649, 651, 653, 655, 657, 659, 661, 663, 665, 667, 669, 671, 673, 675, 677, 679, 681, 683, 685, 687, 689, 691, 693, 695, 697, 699, 701, 703, 705, 707, 709, 711, 713, 715, 717, 719, 721, 723, 725, 727, 729, 731, 733, 735, 737, 739, 741, 743, 745, 747, 749, 751, 753, 755, 757, 759, 761, 763, 765, 767, 769, 771, 773, 775, 777, 779, 781, 783, 785, 787, 789, 791, 793, 795, 797, 799, 801, 803, 805, 807, 809, 811, 813, 815, 817, 819, 821, 823, 825, 827, 829, 831, 833, 835, 837, 839, 841, 843, 845, 847, 849, 851, 853, 855, 857, 859, 861, 863, 865, 867, 869, 871, 873, 875, 877, 879, 881, 883, 885, 887, 889, 891, 893, 895, 897, 899, 901, 903, 905, 907, 909, 911, 913, 915, 917, 919, 921, 923, 925, 927, 929, 931, 933, 935, 937, 939, 941, 943, 945, 947, 949, 951, 953, 955, 957, 959, 961, 963, 965, 967, 969, 971, 973, 975, 977, 979, 981, 983, 985, 987, 989, 991, 993, 995, 997, 999]
```

```
821, 823, 825, 827, 829, 831, 833, 835, 837, 839, 841, 843, 845, 847, 849, 851, 853, 855, 857, 859, 861, 863, 865, 867, 869, 871, 873, 875, 877, 879, 881, 883, 885, 887, 889, 891, 893, 895, 897, 899, 901, 903, 905, 907, 909, 911, 913, 915, 917, 919, 921, 923, 925, 927, 929, 931, 933, 935, 937, 939, 941, 943, 945, 947, 949, 951, 953, 955, 957, 959, 961, 963, 965, 967, 969, 971, 973, 975, 977, 979, 981, 983, 985, 987, 989, 991, 993, 995, 997, 999]
```

Q3:

التعليمة الأولى لإنشاء قائمة بأسماء المواد المطلوبة، التعليمة الثانية لإنشاء قائمة فارغة لنضع فيها المواد التي تبدأ بحرف P، الحلقة تدور على كل العناصر في المصفوفة L ويتم اختبار أول حرف من كل مادة، في حال كان P نضيفه إلى المصفوفة words ثم نطبع القائمة الناتجة.

```
L = ['Network', 'Math', 'Programming', 'Physics', 'Music']
words = []
for x in L:
    if(x[0] == 'P'):
        words.append(x)
print(words)
```

عند التنفيذ:

```
['Programming', 'Physics']
```

Q4:

التعليمة التالية يتم إنشاء قاموس كل عنصر فيه num يساوي إلى قيمته للتربيع  $num*num$  والعناصر فيه هي الأعداد ضمن المجال [1,11].

```
my_dict = {num:num*num for num in range(1,11)}
print(my_dict)
```

عند التنفيذ:

```
{1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7: 49, 8: 64, 9: 81, 10: 100}
```

## السؤال الثاني:

في التعليم الأولى نطلب من المستخدم إدخال عدد عشري، ثم نقصر هذا العدد لعدد صحيح (لأنه عند الإدخال يعتبر سلسلة محرفية)، ثم عرفنا قائمة فارغة لنضع فيها نتيجة تحويل العدد من نظام عد عشري إلى نظام عد ثنائي، ثم ضمن حلقة while نختبر طالما هذا العدد أكبر من صفر، نضيف إلى قائمة النتيجة باقي قسمة العدد على 2 ونقسم العدد نفسه على 2 (خوارزمية تحويل عدد من نظام عشري إلى ثنائي)، بعد انتهاء الحلقة نعكس القائمة حتى نقرأ من اليسار إلى اليمين، ونضع عناصر القائمة في سلسلة محرفية ونعرضها للمستخدم.

```
dec_num = int(input('Enter a Dicemal number:'))
bin_num_list = []
res = dec_num
while res > 0:
    bin_num_list.append(res%2)
    res//=2
bin_num_list.reverse()
bin_num = ""
for bit in bin_num_list:
    bin_num += str(bit)
print('The binary number:',bin_num)
```

عند التنفيذ:

```
Enter a Dicemal number:125
The binary number: 1111101
```

```
Enter a Dicemal number:9
The binary number: 1001
```

### السؤال الثالث:

في البداية نفتح ملف الأسئلة والأجوبة ونحزن الملف ف f، ثم عرفنا متحول content يدل على محتوى الملف، وخرزنا كل سطر بقائمة اسمها lines وأغلقتنا الملف.

في الحلقة الأولى كل سطر يتألف من سؤال وجواب يفصل بينهما ":" فصلنا الأسئلة عن الأجوبة وتم تخزينهم في قائمتين منفصلتين، بعد انتهاء الحلقة وإدخال اسم المستخدم وتعريف متحول يدل على عدد الأسئلة وتهيئ العلامة بقيمة أولية صفر يتم تنفيذ الحلقة الثانية لتطبع الأسئلة وتطلب من المستخدم إدخال إجابته، في حال تطابق جواب المستخدم مع الإجابة الصحيحة من الملف يتم زيادة العلامات بمقدار علامة، في النهاية يتم كتابة النتيجة في ملف جديد.

```
questions = []
answers = []
f = open('questions_answers.txt')
content = f.read()
lines = content.split('\n')
f.close()

for line in lines:
    q_a = line.split(':')
    questions.append(q_a[0])
    answers.append(q_a[1])

count = len(questions)
```

```

name = input('Enter Your name: ')
print('Quiz started.\nFor each question, Please Answer Y or N')
mark = 0

for i in range(count):
    question = questions[i]
    answer = answers[i]
    student_answer = input(question+". (Y or N)? ")
    if student_answer==answer:
        mark+=1

print('Quiz Finished.\nyour mark:', mark,'/',count)

f = open('results.txt', 'a')
f.write(name + " -> " + str(mark) + "/" + str(count) + "\n")

```

ملف الأسئلة:

```

≡ questions_answers.txt
1  Damascus is the capital of Syria:Y
2  Iskandaria is the capital of Egypt:N
3  Baghdad is the capital of Iraq:Y
4  Damascus is the capital of Syria:Y
5  Iskandaria is the capital of Egypt:N
6  Baghdad is the capital of Iraq:Y

```

عند التنفيذ:

```

Enter Your name: Mohammad Nour
Quiz started.
For each question, Please Answer Y or N
Damascus is the capital of Syria. (Y or N)? Y
Iskandaria is the capital of Egypt. (Y or N)? N
Baghdad is the capital of Iraq. (Y or N)? Y
Damascus is the capital of Syria. (Y or N)? Y
Iskandaria is the capital of Egypt. (Y or N)? Y
Baghdad is the capital of Iraq. (Y or N)? Y
Quiz Finished.
your mark: 5 / 6

```

ملف النتيجة:

```
results.txt
1 | Mohammad Nour -> 5/6
2
```