4. Working with Lists

Looping through an entire list

- How to loop with lists
- looping = always the same action, with every item in the list

In [1]:

```
magicians = ['alice', 'david', 'carolina']
for magician in magicians:
    print(magician)
```

```
alice
david
carolina
```

- this **for loop** tells
  - drop a name from the *list magicians*
  - and store it in the *variable magician*
  - the list will be repeated once for each element in the list
  - “For every magician in the list of magicians, print the magician’s name.”
  - the variable name should be meaningful e.g. "for dog in dogs:"
  - the loop will repeat for each element in the list, until the last item

In [4]:

```
magicians = ['alice', 'david', 'carolina']
for magician in magicians:
    print(magician.title() + " , that was a gret trick!")
```

```
Alice , that was a gret trick!
David , that was a gret trick!
Carolina , that was a gret trick!
```

In [10]:

```
magicians = ['alice', 'david', 'carolina']
for magician in magicians:
    print(magician.title() + " , that was a gret trick!")
    print("I can't wait to see the next trick, " + magician.title() + ".\n")
```

```
Alice , that was a gret trick!
I can't wait to see the next trick, Alice.

David , that was a gret trick!
I can't wait to see the next trick, David.

Carolina , that was a gret trick!
I can't wait to see the next trick, Carolina.
```

- see you can also add more "tasks" to the loop after the intented line

Avoiding Indentation errors

In [11]:

```
magicians = ['alice', 'david', 'carolina']
for magician in magicians:
    print(magician.title() + " , that was a gret trick!")
    print("I can't wait to see the next trick, " + magician.title() + ".\n")

print("Thank you everyone, that was a nice show!")
```

```
Alice , that was a gret trick!
I can't wait to see the next trick, Alice.

David , that was a gret trick!
I can't wait to see the next trick, David.

Carolina , that was a gret trick!
I can't wait to see the next trick, Carolina.

Thank you everyone, that was a nice show!
```

- what is not after the intented line, will be not repeated!

In [13]:

```
magicians = ['alice', 'david', 'carolina']
for magician in magicians:
    print(magician.title() + " , that was a gret trick!")
print("I can't wait to see the next trick, " + magician.title() + ".\n")
```

```
Alice , that was a gret trick!
David , that was a gret trick!
Carolina , that was a gret trick!
I can't wait to see the next trick, Carolina.
```

- this is a *logical error*
- the syntax is correct but the last print statement is not intented, so it will be not in the **for loop**

In [14]:

```
message = "Hello Python"
    print(message)
```

```
  File "<ipython-input-14-5aea29e09cc4>", line 2
    print(message)
    ^
IndentationError: unexpected indent
```

- this intented line is unnecessary, so it will be an error
- the print satement doesn't belong to the line before so it hasn't be intented

In [15]:

```
magicians = ['alice', 'david', 'carolina']
for magician in magicians
    print(magician.title() + " , that was a gret trick!")
```

```
  File "<ipython-input-15-658fc365585b>", line 2
    for magician in magicians
                             ^
SyntaxError: invalid syntax
```

- don't forget the **colon" (at the end of the for loop)**

Tasks

- 4-1. Pizzas: Think of at least three kinds of your favorite pizza. Store these pizza names in a list, and then use a for loop to print the name of each pizza.
- Modify your for loop to print a sentence using the name of the pizza instead of printing just the name of the pizza For each pizza you should have one line of output containing a simple statement like I like pepperoni pizza. Add a line at the end of your program, outside the for loop, that states how much you like pizza. The output should consist of three or more lines about the kinds of pizza you like and then an additional sentence, such as I really love pizza!
- 4-2. Animals: Think of at least three different animals that have a common characteristic. Store the names of these animals in a list, and then use a for loop to print out the name of each animal.
- Modify your program to print a statement about each animal, such as A dog would make a great pet.
- Add a line at the end of your program stating what these animals have in common . You could print a sentence such as Any of these animals would make a great pet!

In [6]:

```
#4.1
pizzas = ['sucuk', 'extra extra cheese', 'tuna']
for pizza in pizzas:
    print("My favorite Pizza is: " + pizza.title() + ".")

print("\nYour pizza is ready!!")
```

```
My favorite Pizza is: Sucuk.
My favorite Pizza is: Extra Extra Cheese.
My favorite Pizza is: Tuna.

Your pizza is ready!!
```

In [9]:

```
#4.2
animals = ['dog', 'cat', 'bird']
for animal in animals:
    print("A " + animal.title() + " would make a great pet!")

print("\nAll theese animals are pets!")
```

```
A Dog would make a great pet!
A Cat would make a great pet!
A Bird would make a great pet!

All theese animals are pets!
```

Making Numerical Lists

In [16]:

```
for value in range(1,5):
    print(value)
```

```
1
2
3
4
```

- **range() function** generates a series of numbers
- the loop creates a range of number between 1 and 5 (1, 2, 3, 4) **not 5** because 5 is not in the range!!!
- the **range() function** starts in the number you tell him (in this case 1)

In [17]:

```
for value in range(0,6):
    print(value)
```

```
0
1
2
3
4
5
```

- you can also start with 0

In [18]:

```
numbers = list(range(1,6))
print(numbers)
```

```
[1, 2, 3, 4, 5]
```

- you can convert the result of **range()** into a **list() function**
- the output will be a *list of numbers*
- in the previous examples we just printed out a series of numbers (it was not a list)

In [19]:

```
even_numbers = list(range(2,11,2))
print(even_numbers)
```

```
[2, 4, 6, 8, 10]
```

- in this example the **range() function** starts with the number 2 and than adds 2 to the value
- the range is until 11
- you can create any number with the **range() function:**

In [20]:

```
squares = []
for value in range(1,11):
    square = value**2
    squares.append(square)

print(squares)
```

```
[1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
```

- first we created a empty list *squares = []*
- then we have a *for loop*
- our variable name is *value*
- our range starts from 1 and ends with 10
- the operation is
  - we store the square of each value (** means square) in the variable name square
  - we append the variable *square* into the list named *squares* with the *append() function*
- we print the list
- **we have for each item the square in a list**

In [21]:

```
squares = []
for value in range(1,11):
    squares.append(value**2)

print(squares)
```

```
[1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
```

- does pretty much the same
- we don't have an extra variable *square*
- we put it directly into the *append() function*

In [23]:

```
digits = [1, 2, 3, 4, 5, 6, 7, 8, 9, 0]
min(digits)
```

Out[23]:

```
0
```

In [24]:

```
digits = [1, 2, 3, 4, 5, 6, 7, 8, 9, 0]
max(digits)
```

Out[24]:

```
9
```

In [25]:

```
digits = [1, 2, 3, 4, 5, 6, 7, 8, 9, 0]
sum(digits)
```

Out[25]:

```
45
```

- some statistics with numbers
- give me the
  - minimum of the list
  - maximum of the list
  - sum of the list

In [26]:

```
squares = [value**2 for value in range(1,11)]
print(squares)
```

```
[1, 4, 9, 16, 25, 36, 49, 64, 81, 100]
```

- one line of code does the same as above
- we have a list named *squares*
- *value**2* will reach value to the second power
- then we write a *for loop* with the wished range()
- no colon is used
- this is called ***list comprehensions***

Tasks

- 4-3. Counting to Twenty: Use a for loop to print the numbers from 1 to 20, inclusive.
- 4-4. One Million: Make a list of the numbers from one to one million, and then use a for loop to print the numbers. (If the output is taking too long, stop it by pressing ctrl-C or by closing the output window.)
- 4-5. Summing a Million: Make a list of the numbers from one to one million, and then use min() and max() to make sure your list actually starts at one and ends at one million. Also, use the sum() function to see how quickly Python can add a million numbers.
- 4-6. Odd Numbers: Use the third argument of the range() function to make a list of the odd numbers from 1 to 20. Use a for loop to print each number.
- 4-7. Threes: Make a list of the multiples of 3 from 3 to 30 . Use a for loop to print the numbers in your list.
- 4-8. Cubes: A number raised to the third power is called a cube. For example, the cube of 2 is written as 2**3 in Python. Make a list of the first 10 cubes (that is, the cube of each integer from 1 through 10), and use a for loop to print out the value of each cube.
- 4-9. Cube Comprehension: Use a list comprehension to generate a list of the first 10 cubes .

In [12]:

```
#4.3
numbers = []
for value in range(1,21):
    numbers.append(value)

print(numbers)
```

```
[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19,
20]
```

In [26]:

```
#4.4
numbers = []
for value in range(1, 10000):
    numbers.append(value)
print(numbers)
```

[1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19,
20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36,
37, 38, 39, 40, 41, 42, 43, 44, 45, 46, 47, 48, 49, 50, 51, 52, 53,
54, 55, 56, 57, 58, 59, 60, 61, 62, 63, 64, 65, 66, 67, 68, 69, 70,
71, 72, 73, 74, 75, 76, 77, 78, 79, 80, 81, 82, 83, 84, 85, 86, 87,
88, 89, 90, 91, 92, 93, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103,
104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116, 11
7, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128, 129, 130,
131, 132, 133, 134, 135, 136, 137, 138, 139, 140, 141, 142, 143, 14
4, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157,
158, 159, 160, 161, 162, 163, 164, 165, 166, 167, 168, 169, 170, 17
1, 172, 173, 174, 175, 176, 177, 178, 179, 180, 181, 182, 183, 184,
185, 186, 187, 188, 189, 190, 191, 192, 193, 194, 195, 196, 197, 19
8, 199, 200, 201, 202, 203, 204, 205, 206, 207, 208, 209, 210, 211,
212, 213, 214, 215, 216, 217, 218, 219, 220, 221, 222, 223, 224, 22
5, 226, 227, 228, 229, 230, 231, 232, 233, 234, 235, 236, 237, 238,
239, 240, 241, 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 25
2, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265,
266, 267, 268, 269, 270, 271, 272, 273, 274, 275, 276, 277, 278, 27
9, 280, 281, 282, 283, 284, 285, 286, 287, 288, 289, 290, 291, 292,
293, 294, 295, 296, 297, 298, 299, 300, 301, 302, 303, 304, 305, 30
6, 307, 308, 309, 310, 311, 312, 313, 314, 315, 316, 317, 318, 319,
320, 321, 322, 323, 324, 325, 326, 327, 328, 329, 330, 331, 332, 33
3, 334, 335, 336, 337, 338, 339, 340, 341, 342, 343, 344, 345, 346,
347, 348, 349, 350, 351, 352, 353, 354, 355, 356, 357, 358, 359, 36
0, 361, 362, 363, 364, 365, 366, 367, 368, 369, 370, 371, 372, 373,
374, 375, 376, 377, 378, 379, 380, 381, 382, 383, 384, 385, 386, 38
7, 388, 389, 390, 391, 392, 393, 394, 395, 396, 397, 398, 399, 400,
401, 402, 403, 404, 405, 406, 407, 408, 409, 410, 411, 412, 413, 41
4, 415, 416, 417, 418, 419, 420, 421, 422, 423, 424, 425, 426, 427,
428, 429, 430, 431, 432, 433, 434, 435, 436, 437, 438, 439, 440, 44
1, 442, 443, 444, 445, 446, 447, 448, 449, 450, 451, 452, 453, 454,
455, 456, 457, 458, 459, 460, 461, 462, 463, 464, 465, 466, 467, 46
8, 469, 470, 471, 472, 473, 474, 475, 476, 477, 478, 479, 480, 481,
482, 483, 484, 485, 486, 487, 488, 489, 490, 491, 492, 493, 494, 49
5, 496, 497, 498, 499, 500, 501, 502, 503, 504, 505, 506, 507, 508,
509, 510, 511, 512, 513, 514, 515, 516, 517, 518, 519, 520, 521, 52
2, 523, 524, 525, 526, 527, 528, 529, 530, 531, 532, 533, 534, 535,
536, 537, 538, 539, 540, 541, 542, 543, 544, 545, 546, 547, 548, 54
9, 550, 551, 552, 553, 554, 555, 556, 557, 558, 559, 560, 561, 562,
563, 564, 565, 566, 567, 568, 569, 570, 571, 572, 573, 574, 575, 57
6, 577, 578, 579, 580, 581, 582, 583, 584, 585, 586, 587, 588, 589,
590, 591, 592, 593, 594, 595, 596, 597, 598, 599, 600, 601, 602, 60
3, 604, 605, 606, 607, 608, 609, 610, 611, 612, 613, 614, 615, 616,
617, 618, 619, 620, 621, 622, 623, 624, 625, 626, 627, 628, 629, 63
0, 631, 632, 633, 634, 635, 636, 637, 638, 639, 640, 641, 642, 643,
644, 645, 646, 647, 648, 649, 650, 651, 652, 653, 654, 655, 656, 65
7, 658, 659, 660, 661, 662, 663, 664, 665, 666, 667, 668, 669, 670,
671, 672, 673, 674, 675, 676, 677, 678, 679, 680, 681, 682, 683, 68
4, 685, 686, 687, 688, 689, 690, 691, 692, 693, 694, 695, 696, 697,
698, 699, 700, 701, 702, 703, 704, 705, 706, 707, 708, 709, 710, 71
1, 712, 713, 714, 715, 716, 717, 718, 719, 720, 721, 722, 723, 724,
725, 726, 727, 728, 729, 730, 731, 732, 733, 734, 735, 736, 737, 73
8, 739, 740, 741, 742, 743, 744, 745, 746, 747, 748, 749, 750, 751,
752, 753, 754, 755, 756, 757, 758, 759, 760, 761, 762, 763, 764, 76
5, 766, 767, 768, 769, 770, 771, 772, 773, 774, 775, 776, 777, 778,
779, 780, 781, 782, 783, 784, 785, 786, 787, 788, 789, 790, 791, 79
2, 793, 794, 795, 796, 797, 798, 799, 800, 801, 802, 803, 804, 805,
806, 807, 808, 809, 810, 811, 812, 813, 814, 815, 816, 817, 818, 81
9, 820, 821, 822, 823, 824, 825, 826, 827, 828, 829, 830, 831, 832,
833, 834, 835, 836, 837, 838, 839, 840, 841, 842, 843, 844, 845, 84

6, 847, 848, 849, 850, 851, 852, 853, 854, 855, 856, 857, 858, 859,
860, 861, 862, 863, 864, 865, 866, 867, 868, 869, 870, 871, 872, 87
3, 874, 875, 876, 877, 878, 879, 880, 881, 882, 883, 884, 885, 886,
887, 888, 889, 890, 891, 892, 893, 894, 895, 896, 897, 898, 899, 90
0, 901, 902, 903, 904, 905, 906, 907, 908, 909, 910, 911, 912, 913,
914, 915, 916, 917, 918, 919, 920, 921, 922, 923, 924, 925, 926, 92
7, 928, 929, 930, 931, 932, 933, 934, 935, 936, 937, 938, 939, 940,
941, 942, 943, 944, 945, 946, 947, 948, 949, 950, 951, 952, 953, 95
4, 955, 956, 957, 958, 959, 960, 961, 962, 963, 964, 965, 966, 967,
968, 969, 970, 971, 972, 973, 974, 975, 976, 977, 978, 979, 980, 98
1, 982, 983, 984, 985, 986, 987, 988, 989, 990, 991, 992, 993, 994,
995, 996, 997, 998, 999, 1000, 1001, 1002, 1003, 1004, 1005, 1006, 1
007, 1008, 1009, 1010, 1011, 1012, 1013, 1014, 1015, 1016, 1017, 101
8, 1019, 1020, 1021, 1022, 1023, 1024, 1025, 1026, 1027, 1028, 1029,
1030, 1031, 1032, 1033, 1034, 1035, 1036, 1037, 1038, 1039, 1040, 10
41, 1042, 1043, 1044, 1045, 1046, 1047, 1048, 1049, 1050, 1051, 105
2, 1053, 1054, 1055, 1056, 1057, 1058, 1059, 1060, 1061, 1062, 1063,
1064, 1065, 1066, 1067, 1068, 1069, 1070, 1071, 1072, 1073, 1074, 10
75, 1076, 1077, 1078, 1079, 1080, 1081, 1082, 1083, 1084, 1085, 108
6, 1087, 1088, 1089, 1090, 1091, 1092, 1093, 1094, 1095, 1096, 1097,
1098, 1099, 1100, 1101, 1102, 1103, 1104, 1105, 1106, 1107, 1108, 11
09, 1110, 1111, 1112, 1113, 1114, 1115, 1116, 1117, 1118, 1119, 112
0, 1121, 1122, 1123, 1124, 1125, 1126, 1127, 1128, 1129, 1130, 1131,
1132, 1133, 1134, 1135, 1136, 1137, 1138, 1139, 1140, 1141, 1142, 11
43, 1144, 1145, 1146, 1147, 1148, 1149, 1150, 1151, 1152, 1153, 115
4, 1155, 1156, 1157, 1158, 1159, 1160, 1161, 1162, 1163, 1164, 1165,
1166, 1167, 1168, 1169, 1170, 1171, 1172, 1173, 1174, 1175, 1176, 11
77, 1178, 1179, 1180, 1181, 1182, 1183, 1184, 1185, 1186, 1187, 118
8, 1189, 1190, 1191, 1192, 1193, 1194, 1195, 1196, 1197, 1198, 1199,
1200, 1201, 1202, 1203, 1204, 1205, 1206, 1207, 1208, 1209, 1210, 12
11, 1212, 1213, 1214, 1215, 1216, 1217, 1218, 1219, 1220, 1221, 122
2, 1223, 1224, 1225, 1226, 1227, 1228, 1229, 1230, 1231, 1232, 1233,
1234, 1235, 1236, 1237, 1238, 1239, 1240, 1241, 1242, 1243, 1244, 12
45, 1246, 1247, 1248, 1249, 1250, 1251, 1252, 1253, 1254, 1255, 125
6, 1257, 1258, 1259, 1260, 1261, 1262, 1263, 1264, 1265, 1266, 1267,
1268, 1269, 1270, 1271, 1272, 1273, 1274, 1275, 1276, 1277, 1278, 12
79, 1280, 1281, 1282, 1283, 1284, 1285, 1286, 1287, 1288, 1289, 129
0, 1291, 1292, 1293, 1294, 1295, 1296, 1297, 1298, 1299, 1300, 1301,
1302, 1303, 1304, 1305, 1306, 1307, 1308, 1309, 1310, 1311, 1312, 13
13, 1314, 1315, 1316, 1317, 1318, 1319, 1320, 1321, 1322, 1323, 132
4, 1325, 1326, 1327, 1328, 1329, 1330, 1331, 1332, 1333, 1334, 1335,
1336, 1337, 1338, 1339, 1340, 1341, 1342, 1343, 1344, 1345, 1346, 13
47, 1348, 1349, 1350, 1351, 1352, 1353, 1354, 1355, 1356, 1357, 135
8, 1359, 1360, 1361, 1362, 1363, 1364, 1365, 1366, 1367, 1368, 1369,
1370, 1371, 1372, 1373, 1374, 1375, 1376, 1377, 1378, 1379, 1380, 13
81, 1382, 1383, 1384, 1385, 1386, 1387, 1388, 1389, 1390, 1391, 139
2, 1393, 1394, 1395, 1396, 1397, 1398, 1399, 1400, 1401, 1402, 1403,
1404, 1405, 1406, 1407, 1408, 1409, 1410, 1411, 1412, 1413, 1414, 14
15, 1416, 1417, 1418, 1419, 1420, 1421, 1422, 1423, 1424, 1425, 142
6, 1427, 1428, 1429, 1430, 1431, 1432, 1433, 1434, 1435, 1436, 1437,
1438, 1439, 1440, 1441, 1442, 1443, 1444, 1445, 1446, 1447, 1448, 14
49, 1450, 1451, 1452, 1453, 1454, 1455, 1456, 1457, 1458, 1459, 146
0, 1461, 1462, 1463, 1464, 1465, 1466, 1467, 1468, 1469, 1470, 1471,
1472, 1473, 1474, 1475, 1476, 1477, 1478, 1479, 1480, 1481, 1482, 14
83, 1484, 1485, 1486, 1487, 1488, 1489, 1490, 1491, 1492, 1493, 149
4, 1495, 1496, 1497, 1498, 1499, 1500, 1501, 1502, 1503, 1504, 1505,
1506, 1507, 1508, 1509, 1510, 1511, 1512, 1513, 1514, 1515, 1516, 15
17, 1518, 1519, 1520, 1521, 1522, 1523, 1524, 1525, 1526, 1527, 152
8, 1529, 1530, 1531, 1532, 1533, 1534, 1535, 1536, 1537, 1538, 1539,
1540, 1541, 1542, 1543, 1544, 1545, 1546, 1547, 1548, 1549, 1550, 15
51, 1552, 1553, 1554, 1555, 1556, 1557, 1558, 1559, 1560, 1561, 156

2, 1563, 1564, 1565, 1566, 1567, 1568, 1569, 1570, 1571, 1572, 1573,
1574, 1575, 1576, 1577, 1578, 1579, 1580, 1581, 1582, 1583, 1584, 15
85, 1586, 1587, 1588, 1589, 1590, 1591, 1592, 1593, 1594, 1595, 159
6, 1597, 1598, 1599, 1600, 1601, 1602, 1603, 1604, 1605, 1606, 1607,
1608, 1609, 1610, 1611, 1612, 1613, 1614, 1615, 1616, 1617, 1618, 16
19, 1620, 1621, 1622, 1623, 1624, 1625, 1626, 1627, 1628, 1629, 163
0, 1631, 1632, 1633, 1634, 1635, 1636, 1637, 1638, 1639, 1640, 1641,
1642, 1643, 1644, 1645, 1646, 1647, 1648, 1649, 1650, 1651, 1652, 16
53, 1654, 1655, 1656, 1657, 1658, 1659, 1660, 1661, 1662, 1663, 166
4, 1665, 1666, 1667, 1668, 1669, 1670, 1671, 1672, 1673, 1674, 1675,
1676, 1677, 1678, 1679, 1680, 1681, 1682, 1683, 1684, 1685, 1686, 16
87, 1688, 1689, 1690, 1691, 1692, 1693, 1694, 1695, 1696, 1697, 169
8, 1699, 1700, 1701, 1702, 1703, 1704, 1705, 1706, 1707, 1708, 1709,
1710, 1711, 1712, 1713, 1714, 1715, 1716, 1717, 1718, 1719, 1720, 17
21, 1722, 1723, 1724, 1725, 1726, 1727, 1728, 1729, 1730, 1731, 173
2, 1733, 1734, 1735, 1736, 1737, 1738, 1739, 1740, 1741, 1742, 1743,
1744, 1745, 1746, 1747, 1748, 1749, 1750, 1751, 1752, 1753, 1754, 17
55, 1756, 1757, 1758, 1759, 1760, 1761, 1762, 1763, 1764, 1765, 176
6, 1767, 1768, 1769, 1770, 1771, 1772, 1773, 1774, 1775, 1776, 1777,
1778, 1779, 1780, 1781, 1782, 1783, 1784, 1785, 1786, 1787, 1788, 17
89, 1790, 1791, 1792, 1793, 1794, 1795, 1796, 1797, 1798, 1799, 180
0, 1801, 1802, 1803, 1804, 1805, 1806, 1807, 1808, 1809, 1810, 1811,
1812, 1813, 1814, 1815, 1816, 1817, 1818, 1819, 1820, 1821, 1822, 18
23, 1824, 1825, 1826, 1827, 1828, 1829, 1830, 1831, 1832, 1833, 183
4, 1835, 1836, 1837, 1838, 1839, 1840, 1841, 1842, 1843, 1844, 1845,
1846, 1847, 1848, 1849, 1850, 1851, 1852, 1853, 1854, 1855, 1856, 18
57, 1858, 1859, 1860, 1861, 1862, 1863, 1864, 1865, 1866, 1867, 186
8, 1869, 1870, 1871, 1872, 1873, 1874, 1875, 1876, 1877, 1878, 1879,
1880, 1881, 1882, 1883, 1884, 1885, 1886, 1887, 1888, 1889, 1890, 18
91, 1892, 1893, 1894, 1895, 1896, 1897, 1898, 1899, 1900, 1901, 190
2, 1903, 1904, 1905, 1906, 1907, 1908, 1909, 1910, 1911, 1912, 1913,
1914, 1915, 1916, 1917, 1918, 1919, 1920, 1921, 1922, 1923, 1924, 19
25, 1926, 1927, 1928, 1929, 1930, 1931, 1932, 1933, 1934, 1935, 193
6, 1937, 1938, 1939, 1940, 1941, 1942, 1943, 1944, 1945, 1946, 1947,
1948, 1949, 1950, 1951, 1952, 1953, 1954, 1955, 1956, 1957, 1958, 19
59, 1960, 1961, 1962, 1963, 1964, 1965, 1966, 1967, 1968, 1969, 197
0, 1971, 1972, 1973, 1974, 1975, 1976, 1977, 1978, 1979, 1980, 1981,
1982, 1983, 1984, 1985, 1986, 1987, 1988, 1989, 1990, 1991, 1992, 19
93, 1994, 1995, 1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003, 200
4, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015,
2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023, 2024, 2025, 2026, 20
27, 2028, 2029, 2030, 2031, 2032, 2033, 2034, 2035, 2036, 2037, 203
8, 2039, 2040, 2041, 2042, 2043, 2044, 2045, 2046, 2047, 2048, 2049,
2050, 2051, 2052, 2053, 2054, 2055, 2056, 2057, 2058, 2059, 2060, 20
61, 2062, 2063, 2064, 2065, 2066, 2067, 2068, 2069, 2070, 2071, 207
2, 2073, 2074, 2075, 2076, 2077, 2078, 2079, 2080, 2081, 2082, 2083,
2084, 2085, 2086, 2087, 2088, 2089, 2090, 2091, 2092, 2093, 2094, 20
95, 2096, 2097, 2098, 2099, 2100, 2101, 2102, 2103, 2104, 2105, 210
6, 2107, 2108, 2109, 2110, 2111, 2112, 2113, 2114, 2115, 2116, 2117,
2118, 2119, 2120, 2121, 2122, 2123, 2124, 2125, 2126, 2127, 2128, 21
29, 2130, 2131, 2132, 2133, 2134, 2135, 2136, 2137, 2138, 2139, 214
0, 2141, 2142, 2143, 2144, 2145, 2146, 2147, 2148, 2149, 2150, 2151,
2152, 2153, 2154, 2155, 2156, 2157, 2158, 2159, 2160, 2161, 2162, 21
63, 2164, 2165, 2166, 2167, 2168, 2169, 2170, 2171, 2172, 2173, 217
4, 2175, 2176, 2177, 2178, 2179, 2180, 2181, 2182, 2183, 2184, 2185,
2186, 2187, 2188, 2189, 2190, 2191, 2192, 2193, 2194, 2195, 2196, 21
97, 2198, 2199, 2200, 2201, 2202, 2203, 2204, 2205, 2206, 2207, 220
8, 2209, 2210, 2211, 2212, 2213, 2214, 2215, 2216, 2217, 2218, 2219,
2220, 2221, 2222, 2223, 2224, 2225, 2226, 2227, 2228, 2229, 2230, 22
31, 2232, 2233, 2234, 2235, 2236, 2237, 2238, 2239, 2240, 2241, 224
2, 2243, 2244, 2245, 2246, 2247, 2248, 2249, 2250, 2251, 2252, 2253,

2254, 2255, 2256, 2257, 2258, 2259, 2260, 2261, 2262, 2263, 2264, 22
65, 2266, 2267, 2268, 2269, 2270, 2271, 2272, 2273, 2274, 2275, 227
6, 2277, 2278, 2279, 2280, 2281, 2282, 2283, 2284, 2285, 2286, 2287,
2288, 2289, 2290, 2291, 2292, 2293, 2294, 2295, 2296, 2297, 2298, 22
99, 2300, 2301, 2302, 2303, 2304, 2305, 2306, 2307, 2308, 2309, 231
0, 2311, 2312, 2313, 2314, 2315, 2316, 2317, 2318, 2319, 2320, 2321,
2322, 2323, 2324, 2325, 2326, 2327, 2328, 2329, 2330, 2331, 2332, 23
33, 2334, 2335, 2336, 2337, 2338, 2339, 2340, 2341, 2342, 2343, 234
4, 2345, 2346, 2347, 2348, 2349, 2350, 2351, 2352, 2353, 2354, 2355,
2356, 2357, 2358, 2359, 2360, 2361, 2362, 2363, 2364, 2365, 2366, 23
67, 2368, 2369, 2370, 2371, 2372, 2373, 2374, 2375, 2376, 2377, 237
8, 2379, 2380, 2381, 2382, 2383, 2384, 2385, 2386, 2387, 2388, 2389,
2390, 2391, 2392, 2393, 2394, 2395, 2396, 2397, 2398, 2399, 2400, 24
01, 2402, 2403, 2404, 2405, 2406, 2407, 2408, 2409, 2410, 2411, 241
2, 2413, 2414, 2415, 2416, 2417, 2418, 2419, 2420, 2421, 2422, 2423,
2424, 2425, 2426, 2427, 2428, 2429, 2430, 2431, 2432, 2433, 2434, 24
35, 2436, 2437, 2438, 2439, 2440, 2441, 2442, 2443, 2444, 2445, 244
6, 2447, 2448, 2449, 2450, 2451, 2452, 2453, 2454, 2455, 2456, 2457,
2458, 2459, 2460, 2461, 2462, 2463, 2464, 2465, 2466, 2467, 2468, 24
69, 2470, 2471, 2472, 2473, 2474, 2475, 2476, 2477, 2478, 2479, 248
0, 2481, 2482, 2483, 2484, 2485, 2486, 2487, 2488, 2489, 2490, 2491,
2492, 2493, 2494, 2495, 2496, 2497, 2498, 2499, 2500, 2501, 2502, 25
03, 2504, 2505, 2506, 2507, 2508, 2509, 2510, 2511, 2512, 2513, 251
4, 2515, 2516, 2517, 2518, 2519, 2520, 2521, 2522, 2523, 2524, 2525,
2526, 2527, 2528, 2529, 2530, 2531, 2532, 2533, 2534, 2535, 2536, 25
37, 2538, 2539, 2540, 2541, 2542, 2543, 2544, 2545, 2546, 2547, 254
8, 2549, 2550, 2551, 2552, 2553, 2554, 2555, 2556, 2557, 2558, 2559,
2560, 2561, 2562, 2563, 2564, 2565, 2566, 2567, 2568, 2569, 2570, 25
71, 2572, 2573, 2574, 2575, 2576, 2577, 2578, 2579, 2580, 2581, 258
2, 2583, 2584, 2585, 2586, 2587, 2588, 2589, 2590, 2591, 2592, 2593,
2594, 2595, 2596, 2597, 2598, 2599, 2600, 2601, 2602, 2603, 2604, 26
05, 2606, 2607, 2608, 2609, 2610, 2611, 2612, 2613, 2614, 2615, 261
6, 2617, 2618, 2619, 2620, 2621, 2622, 2623, 2624, 2625, 2626, 2627,
2628, 2629, 2630, 2631, 2632, 2633, 2634, 2635, 2636, 2637, 2638, 26
39, 2640, 2641, 2642, 2643, 2644, 2645, 2646, 2647, 2648, 2649, 265
0, 2651, 2652, 2653, 2654, 2655, 2656, 2657, 2658, 2659, 2660, 2661,
2662, 2663, 2664, 2665, 2666, 2667, 2668, 2669, 2670, 2671, 2672, 26
73, 2674, 2675, 2676, 2677, 2678, 2679, 2680, 2681, 2682, 2683, 268
4, 2685, 2686, 2687, 2688, 2689, 2690, 2691, 2692, 2693, 2694, 2695,
2696, 2697, 2698, 2699, 2700, 2701, 2702, 2703, 2704, 2705, 2706, 27
07, 2708, 2709, 2710, 2711, 2712, 2713, 2714, 2715, 2716, 2717, 271
8, 2719, 2720, 2721, 2722, 2723, 2724, 2725, 2726, 2727, 2728, 2729,
2730, 2731, 2732, 2733, 2734, 2735, 2736, 2737, 2738, 2739, 2740, 27
41, 2742, 2743, 2744, 2745, 2746, 2747, 2748, 2749, 2750, 2751, 275
2, 2753, 2754, 2755, 2756, 2757, 2758, 2759, 2760, 2761, 2762, 2763,
2764, 2765, 2766, 2767, 2768, 2769, 2770, 2771, 2772, 2773, 2774, 27
75, 2776, 2777, 2778, 2779, 2780, 2781, 2782, 2783, 2784, 2785, 278
6, 2787, 2788, 2789, 2790, 2791, 2792, 2793, 2794, 2795, 2796, 2797,
2798, 2799, 2800, 2801, 2802, 2803, 2804, 2805, 2806, 2807, 2808, 28
09, 2810, 2811, 2812, 2813, 2814, 2815, 2816, 2817, 2818, 2819, 282
0, 2821, 2822, 2823, 2824, 2825, 2826, 2827, 2828, 2829, 2830, 2831,
2832, 2833, 2834, 2835, 2836, 2837, 2838, 2839, 2840, 2841, 2842, 28
43, 2844, 2845, 2846, 2847, 2848, 2849, 2850, 2851, 2852, 2853, 285
4, 2855, 2856, 2857, 2858, 2859, 2860, 2861, 2862, 2863, 2864, 2865,
2866, 2867, 2868, 2869, 2870, 2871, 2872, 2873, 2874, 2875, 2876, 28
77, 2878, 2879, 2880, 2881, 2882, 2883, 2884, 2885, 2886, 2887, 288
8, 2889, 2890, 2891, 2892, 2893, 2894, 2895, 2896, 2897, 2898, 2899,
2900, 2901, 2902, 2903, 2904, 2905, 2906, 2907, 2908, 2909, 2910, 29
11, 2912, 2913, 2914, 2915, 2916, 2917, 2918, 2919, 2920, 2921, 292
2, 2923, 2924, 2925, 2926, 2927, 2928, 2929, 2930, 2931, 2932, 2933,
2934, 2935, 2936, 2937, 2938, 2939, 2940, 2941, 2942, 2943, 2944, 29

45, 2946, 2947, 2948, 2949, 2950, 2951, 2952, 2953, 2954, 2955, 295
6, 2957, 2958, 2959, 2960, 2961, 2962, 2963, 2964, 2965, 2966, 2967,
2968, 2969, 2970, 2971, 2972, 2973, 2974, 2975, 2976, 2977, 2978, 29
79, 2980, 2981, 2982, 2983, 2984, 2985, 2986, 2987, 2988, 2989, 299
0, 2991, 2992, 2993, 2994, 2995, 2996, 2997, 2998, 2999, 3000, 3001,
3002, 3003, 3004, 3005, 3006, 3007, 3008, 3009, 3010, 3011, 3012, 30
13, 3014, 3015, 3016, 3017, 3018, 3019, 3020, 3021, 3022, 3023, 302
4, 3025, 3026, 3027, 3028, 3029, 3030, 3031, 3032, 3033, 3034, 3035,
3036, 3037, 3038, 3039, 3040, 3041, 3042, 3043, 3044, 3045, 3046, 30
47, 3048, 3049, 3050, 3051, 3052, 3053, 3054, 3055, 3056, 3057, 305
8, 3059, 3060, 3061, 3062, 3063, 3064, 3065, 3066, 3067, 3068, 3069,
3070, 3071, 3072, 3073, 3074, 3075, 3076, 3077, 3078, 3079, 3080, 30
81, 3082, 3083, 3084, 3085, 3086, 3087, 3088, 3089, 3090, 3091, 309
2, 3093, 3094, 3095, 3096, 3097, 3098, 3099, 3100, 3101, 3102, 3103,
3104, 3105, 3106, 3107, 3108, 3109, 3110, 3111, 3112, 3113, 3114, 31
15, 3116, 3117, 3118, 3119, 3120, 3121, 3122, 3123, 3124, 3125, 312
6, 3127, 3128, 3129, 3130, 3131, 3132, 3133, 3134, 3135, 3136, 3137,
3138, 3139, 3140, 3141, 3142, 3143, 3144, 3145, 3146, 3147, 3148, 31
49, 3150, 3151, 3152, 3153, 3154, 3155, 3156, 3157, 3158, 3159, 316
0, 3161, 3162, 3163, 3164, 3165, 3166, 3167, 3168, 3169, 3170, 3171,
3172, 3173, 3174, 3175, 3176, 3177, 3178, 3179, 3180, 3181, 3182, 31
83, 3184, 3185, 3186, 3187, 3188, 3189, 3190, 3191, 3192, 3193, 319
4, 3195, 3196, 3197, 3198, 3199, 3200, 3201, 3202, 3203, 3204, 3205,
3206, 3207, 3208, 3209, 3210, 3211, 3212, 3213, 3214, 3215, 3216, 32
17, 3218, 3219, 3220, 3221, 3222, 3223, 3224, 3225, 3226, 3227, 322
8, 3229, 3230, 3231, 3232, 3233, 3234, 3235, 3236, 3237, 3238, 3239,
3240, 3241, 3242, 3243, 3244, 3245, 3246, 3247, 3248, 3249, 3250, 32
51, 3252, 3253, 3254, 3255, 3256, 3257, 3258, 3259, 3260, 3261, 326
2, 3263, 3264, 3265, 3266, 3267, 3268, 3269, 3270, 3271, 3272, 3273,
3274, 3275, 3276, 3277, 3278, 3279, 3280, 3281, 3282, 3283, 3284, 32
85, 3286, 3287, 3288, 3289, 3290, 3291, 3292, 3293, 3294, 3295, 329
6, 3297, 3298, 3299, 3300, 3301, 3302, 3303, 3304, 3305, 3306, 3307,
3308, 3309, 3310, 3311, 3312, 3313, 3314, 3315, 3316, 3317, 3318, 33
19, 3320, 3321, 3322, 3323, 3324, 3325, 3326, 3327, 3328, 3329, 333
0, 3331, 3332, 3333, 3334, 3335, 3336, 3337, 3338, 3339, 3340, 3341,
3342, 3343, 3344, 3345, 3346, 3347, 3348, 3349, 3350, 3351, 3352, 33
53, 3354, 3355, 3356, 3357, 3358, 3359, 3360, 3361, 3362, 3363, 336
4, 3365, 3366, 3367, 3368, 3369, 3370, 3371, 3372, 3373, 3374, 3375,
3376, 3377, 3378, 3379, 3380, 3381, 3382, 3383, 3384, 3385, 3386, 33
87, 3388, 3389, 3390, 3391, 3392, 3393, 3394, 3395, 3396, 3397, 339
8, 3399, 3400, 3401, 3402, 3403, 3404, 3405, 3406, 3407, 3408, 3409,
3410, 3411, 3412, 3413, 3414, 3415, 3416, 3417, 3418, 3419, 3420, 34
21, 3422, 3423, 3424, 3425, 3426, 3427, 3428, 3429, 3430, 3431, 343
2, 3433, 3434, 3435, 3436, 3437, 3438, 3439, 3440, 3441, 3442, 3443,
3444, 3445, 3446, 3447, 3448, 3449, 3450, 3451, 3452, 3453, 3454, 34
55, 3456, 3457, 3458, 3459, 3460, 3461, 3462, 3463, 3464, 3465, 346
6, 3467, 3468, 3469, 3470, 3471, 3472, 3473, 3474, 3475, 3476, 3477,
3478, 3479, 3480, 3481, 3482, 3483, 3484, 3485, 3486, 3487, 3488, 34
89, 3490, 3491, 3492, 3493, 3494, 3495, 3496, 3497, 3498, 3499, 350
0, 3501, 3502, 3503, 3504, 3505, 3506, 3507, 3508, 3509, 3510, 3511,
3512, 3513, 3514, 3515, 3516, 3517, 3518, 3519, 3520, 3521, 3522, 35
23, 3524, 3525, 3526, 3527, 3528, 3529, 3530, 3531, 3532, 3533, 353
4, 3535, 3536, 3537, 3538, 3539, 3540, 3541, 3542, 3543, 3544, 3545,
3546, 3547, 3548, 3549, 3550, 3551, 3552, 3553, 3554, 3555, 3556, 35
57, 3558, 3559, 3560, 3561, 3562, 3563, 3564, 3565, 3566, 3567, 356
8, 3569, 3570, 3571, 3572, 3573, 3574, 3575, 3576, 3577, 3578, 3579,
3580, 3581, 3582, 3583, 3584, 3585, 3586, 3587, 3588, 3589, 3590, 35
91, 3592, 3593, 3594, 3595, 3596, 3597, 3598, 3599, 3600, 3601, 360
2, 3603, 3604, 3605, 3606, 3607, 3608, 3609, 3610, 3611, 3612, 3613,
3614, 3615, 3616, 3617, 3618, 3619, 3620, 3621, 3622, 3623, 3624, 36
25, 3626, 3627, 3628, 3629, 3630, 3631, 3632, 3633, 3634, 3635, 363

6, 3637, 3638, 3639, 3640, 3641, 3642, 3643, 3644, 3645, 3646, 3647,
3648, 3649, 3650, 3651, 3652, 3653, 3654, 3655, 3656, 3657, 3658, 36
59, 3660, 3661, 3662, 3663, 3664, 3665, 3666, 3667, 3668, 3669, 367
0, 3671, 3672, 3673, 3674, 3675, 3676, 3677, 3678, 3679, 3680, 3681,
3682, 3683, 3684, 3685, 3686, 3687, 3688, 3689, 3690, 3691, 3692, 36
93, 3694, 3695, 3696, 3697, 3698, 3699, 3700, 3701, 3702, 3703, 370
4, 3705, 3706, 3707, 3708, 3709, 3710, 3711, 3712, 3713, 3714, 3715,
3716, 3717, 3718, 3719, 3720, 3721, 3722, 3723, 3724, 3725, 3726, 37
27, 3728, 3729, 3730, 3731, 3732, 3733, 3734, 3735, 3736, 3737, 373
8, 3739, 3740, 3741, 3742, 3743, 3744, 3745, 3746, 3747, 3748, 3749,
3750, 3751, 3752, 3753, 3754, 3755, 3756, 3757, 3758, 3759, 3760, 37
61, 3762, 3763, 3764, 3765, 3766, 3767, 3768, 3769, 3770, 3771, 377
2, 3773, 3774, 3775, 3776, 3777, 3778, 3779, 3780, 3781, 3782, 3783,
3784, 3785, 3786, 3787, 3788, 3789, 3790, 3791, 3792, 3793, 3794, 37
95, 3796, 3797, 3798, 3799, 3800, 3801, 3802, 3803, 3804, 3805, 380
6, 3807, 3808, 3809, 3810, 3811, 3812, 3813, 3814, 3815, 3816, 3817,
3818, 3819, 3820, 3821, 3822, 3823, 3824, 3825, 3826, 3827, 3828, 38
29, 3830, 3831, 3832, 3833, 3834, 3835, 3836, 3837, 3838, 3839, 384
0, 3841, 3842, 3843, 3844, 3845, 3846, 3847, 3848, 3849, 3850, 3851,
3852, 3853, 3854, 3855, 3856, 3857, 3858, 3859, 3860, 3861, 3862, 38
63, 3864, 3865, 3866, 3867, 3868, 3869, 3870, 3871, 3872, 3873, 387
4, 3875, 3876, 3877, 3878, 3879, 3880, 3881, 3882, 3883, 3884, 3885,
3886, 3887, 3888, 3889, 3890, 3891, 3892, 3893, 3894, 3895, 3896, 38
97, 3898, 3899, 3900, 3901, 3902, 3903, 3904, 3905, 3906, 3907, 390
8, 3909, 3910, 3911, 3912, 3913, 3914, 3915, 3916, 3917, 3918, 3919,
3920, 3921, 3922, 3923, 3924, 3925, 3926, 3927, 3928, 3929, 3930, 39
31, 3932, 3933, 3934, 3935, 3936, 3937, 3938, 3939, 3940, 3941, 394
2, 3943, 3944, 3945, 3946, 3947, 3948, 3949, 3950, 3951, 3952, 3953,
3954, 3955, 3956, 3957, 3958, 3959, 3960, 3961, 3962, 3963, 3964, 39
65, 3966, 3967, 3968, 3969, 3970, 3971, 3972, 3973, 3974, 3975, 397
6, 3977, 3978, 3979, 3980, 3981, 3982, 3983, 3984, 3985, 3986, 3987,
3988, 3989, 3990, 3991, 3992, 3993, 3994, 3995, 3996, 3997, 3998, 39
99, 4000, 4001, 4002, 4003, 4004, 4005, 4006, 4007, 4008, 4009, 401
0, 4011, 4012, 4013, 4014, 4015, 4016, 4017, 4018, 4019, 4020, 4021,
4022, 4023, 4024, 4025, 4026, 4027, 4028, 4029, 4030, 4031, 4032, 40
33, 4034, 4035, 4036, 4037, 4038, 4039, 4040, 4041, 4042, 4043, 404
4, 4045, 4046, 4047, 4048, 4049, 4050, 4051, 4052, 4053, 4054, 4055,
4056, 4057, 4058, 4059, 4060, 4061, 4062, 4063, 4064, 4065, 4066, 40
67, 4068, 4069, 4070, 4071, 4072, 4073, 4074, 4075, 4076, 4077, 407
8, 4079, 4080, 4081, 4082, 4083, 4084, 4085, 4086, 4087, 4088, 4089,
4090, 4091, 4092, 4093, 4094, 4095, 4096, 4097, 4098, 4099, 4100, 41
01, 4102, 4103, 4104, 4105, 4106, 4107, 4108, 4109, 4110, 4111, 411
2, 4113, 4114, 4115, 4116, 4117, 4118, 4119, 4120, 4121, 4122, 4123,
4124, 4125, 4126, 4127, 4128, 4129, 4130, 4131, 4132, 4133, 4134, 41
35, 4136, 4137, 4138, 4139, 4140, 4141, 4142, 4143, 4144, 4145, 414
6, 4147, 4148, 4149, 4150, 4151, 4152, 4153, 4154, 4155, 4156, 4157,
4158, 4159, 4160, 4161, 4162, 4163, 4164, 4165, 4166, 4167, 4168, 41
69, 4170, 4171, 4172, 4173, 4174, 4175, 4176, 4177, 4178, 4179, 418
0, 4181, 4182, 4183, 4184, 4185, 4186, 4187, 4188, 4189, 4190, 4191,
4192, 4193, 4194, 4195, 4196, 4197, 4198, 4199, 4200, 4201, 4202, 42
03, 4204, 4205, 4206, 4207, 4208, 4209, 4210, 4211, 4212, 4213, 421
4, 4215, 4216, 4217, 4218, 4219, 4220, 4221, 4222, 4223, 4224, 4225,
4226, 4227, 4228, 4229, 4230, 4231, 4232, 4233, 4234, 4235, 4236, 42
37, 4238, 4239, 4240, 4241, 4242, 4243, 4244, 4245, 4246, 4247, 424
8, 4249, 4250, 4251, 4252, 4253, 4254, 4255, 4256, 4257, 4258, 4259,
4260, 4261, 4262, 4263, 4264, 4265, 4266, 4267, 4268, 4269, 4270, 42
71, 4272, 4273, 4274, 4275, 4276, 4277, 4278, 4279, 4280, 4281, 428
2, 4283, 4284, 4285, 4286, 4287, 4288, 4289, 4290, 4291, 4292, 4293,
4294, 4295, 4296, 4297, 4298, 4299, 4300, 4301, 4302, 4303, 4304, 43
05, 4306, 4307, 4308, 4309, 4310, 4311, 4312, 4313, 4314, 4315, 431
6, 4317, 4318, 4319, 4320, 4321, 4322, 4323, 4324, 4325, 4326, 4327,

4328, 4329, 4330, 4331, 4332, 4333, 4334, 4335, 4336, 4337, 4338, 43
39, 4340, 4341, 4342, 4343, 4344, 4345, 4346, 4347, 4348, 4349, 435
0, 4351, 4352, 4353, 4354, 4355, 4356, 4357, 4358, 4359, 4360, 4361,
4362, 4363, 4364, 4365, 4366, 4367, 4368, 4369, 4370, 4371, 4372, 43
73, 4374, 4375, 4376, 4377, 4378, 4379, 4380, 4381, 4382, 4383, 438
4, 4385, 4386, 4387, 4388, 4389, 4390, 4391, 4392, 4393, 4394, 4395,
4396, 4397, 4398, 4399, 4400, 4401, 4402, 4403, 4404, 4405, 4406, 44
07, 4408, 4409, 4410, 4411, 4412, 4413, 4414, 4415, 4416, 4417, 441
8, 4419, 4420, 4421, 4422, 4423, 4424, 4425, 4426, 4427, 4428, 4429,
4430, 4431, 4432, 4433, 4434, 4435, 4436, 4437, 4438, 4439, 4440, 44
41, 4442, 4443, 4444, 4445, 4446, 4447, 4448, 4449, 4450, 4451, 445
2, 4453, 4454, 4455, 4456, 4457, 4458, 4459, 4460, 4461, 4462, 4463,
4464, 4465, 4466, 4467, 4468, 4469, 4470, 4471, 4472, 4473, 4474, 44
75, 4476, 4477, 4478, 4479, 4480, 4481, 4482, 4483, 4484, 4485, 448
6, 4487, 4488, 4489, 4490, 4491, 4492, 4493, 4494, 4495, 4496, 4497,
4498, 4499, 4500, 4501, 4502, 4503, 4504, 4505, 4506, 4507, 4508, 45
09, 4510, 4511, 4512, 4513, 4514, 4515, 4516, 4517, 4518, 4519, 452
0, 4521, 4522, 4523, 4524, 4525, 4526, 4527, 4528, 4529, 4530, 4531,
4532, 4533, 4534, 4535, 4536, 4537, 4538, 4539, 4540, 4541, 4542, 45
43, 4544, 4545, 4546, 4547, 4548, 4549, 4550, 4551, 4552, 4553, 455
4, 4555, 4556, 4557, 4558, 4559, 4560, 4561, 4562, 4563, 4564, 4565,
4566, 4567, 4568, 4569, 4570, 4571, 4572, 4573, 4574, 4575, 4576, 45
77, 4578, 4579, 4580, 4581, 4582, 4583, 4584, 4585, 4586, 4587, 458
8, 4589, 4590, 4591, 4592, 4593, 4594, 4595, 4596, 4597, 4598, 4599,
4600, 4601, 4602, 4603, 4604, 4605, 4606, 4607, 4608, 4609, 4610, 46
11, 4612, 4613, 4614, 4615, 4616, 4617, 4618, 4619, 4620, 4621, 462
2, 4623, 4624, 4625, 4626, 4627, 4628, 4629, 4630, 4631, 4632, 4633,
4634, 4635, 4636, 4637, 4638, 4639, 4640, 4641, 4642, 4643, 4644, 46
45, 4646, 4647, 4648, 4649, 4650, 4651, 4652, 4653, 4654, 4655, 465
6, 4657, 4658, 4659, 4660, 4661, 4662, 4663, 4664, 4665, 4666, 4667,
4668, 4669, 4670, 4671, 4672, 4673, 4674, 4675, 4676, 4677, 4678, 46
79, 4680, 4681, 4682, 4683, 4684, 4685, 4686, 4687, 4688, 4689, 469
0, 4691, 4692, 4693, 4694, 4695, 4696, 4697, 4698, 4699, 4700, 4701,
4702, 4703, 4704, 4705, 4706, 4707, 4708, 4709, 4710, 4711, 4712, 47
13, 4714, 4715, 4716, 4717, 4718, 4719, 4720, 4721, 4722, 4723, 472
4, 4725, 4726, 4727, 4728, 4729, 4730, 4731, 4732, 4733, 4734, 4735,
4736, 4737, 4738, 4739, 4740, 4741, 4742, 4743, 4744, 4745, 4746, 47
47, 4748, 4749, 4750, 4751, 4752, 4753, 4754, 4755, 4756, 4757, 475
8, 4759, 4760, 4761, 4762, 4763, 4764, 4765, 4766, 4767, 4768, 4769,
4770, 4771, 4772, 4773, 4774, 4775, 4776, 4777, 4778, 4779, 4780, 47
81, 4782, 4783, 4784, 4785, 4786, 4787, 4788, 4789, 4790, 4791, 479
2, 4793, 4794, 4795, 4796, 4797, 4798, 4799, 4800, 4801, 4802, 4803,
4804, 4805, 4806, 4807, 4808, 4809, 4810, 4811, 4812, 4813, 4814, 48
15, 4816, 4817, 4818, 4819, 4820, 4821, 4822, 4823, 4824, 4825, 482
6, 4827, 4828, 4829, 4830, 4831, 4832, 4833, 4834, 4835, 4836, 4837,
4838, 4839, 4840, 4841, 4842, 4843, 4844, 4845, 4846, 4847, 4848, 48
49, 4850, 4851, 4852, 4853, 4854, 4855, 4856, 4857, 4858, 4859, 486
0, 4861, 4862, 4863, 4864, 4865, 4866, 4867, 4868, 4869, 4870, 4871,
4872, 4873, 4874, 4875, 4876, 4877, 4878, 4879, 4880, 4881, 4882, 48
83, 4884, 4885, 4886, 4887, 4888, 4889, 4890, 4891, 4892, 4893, 489
4, 4895, 4896, 4897, 4898, 4899, 4900, 4901, 4902, 4903, 4904, 4905,
4906, 4907, 4908, 4909, 4910, 4911, 4912, 4913, 4914, 4915, 4916, 49
17, 4918, 4919, 4920, 4921, 4922, 4923, 4924, 4925, 4926, 4927, 492
8, 4929, 4930, 4931, 4932, 4933, 4934, 4935, 4936, 4937, 4938, 4939,
4940, 4941, 4942, 4943, 4944, 4945, 4946, 4947, 4948, 4949, 4950, 49
51, 4952, 4953, 4954, 4955, 4956, 4957, 4958, 4959, 4960, 4961, 496
2, 4963, 4964, 4965, 4966, 4967, 4968, 4969, 4970, 4971, 4972, 4973,
4974, 4975, 4976, 4977, 4978, 4979, 4980, 4981, 4982, 4983, 4984, 49
85, 4986, 4987, 4988, 4989, 4990, 4991, 4992, 4993, 4994, 4995, 499
6, 4997, 4998, 4999, 5000, 5001, 5002, 5003, 5004, 5005, 5006, 5007,
5008, 5009, 5010, 5011, 5012, 5013, 5014, 5015, 5016, 5017, 5018, 50

19, 5020, 5021, 5022, 5023, 5024, 5025, 5026, 5027, 5028, 5029, 503
0, 5031, 5032, 5033, 5034, 5035, 5036, 5037, 5038, 5039, 5040, 5041,
5042, 5043, 5044, 5045, 5046, 5047, 5048, 5049, 5050, 5051, 5052, 50
53, 5054, 5055, 5056, 5057, 5058, 5059, 5060, 5061, 5062, 5063, 506
4, 5065, 5066, 5067, 5068, 5069, 5070, 5071, 5072, 5073, 5074, 5075,
5076, 5077, 5078, 5079, 5080, 5081, 5082, 5083, 5084, 5085, 5086, 50
87, 5088, 5089, 5090, 5091, 5092, 5093, 5094, 5095, 5096, 5097, 509
8, 5099, 5100, 5101, 5102, 5103, 5104, 5105, 5106, 5107, 5108, 5109,
5110, 5111, 5112, 5113, 5114, 5115, 5116, 5117, 5118, 5119, 5120, 51
21, 5122, 5123, 5124, 5125, 5126, 5127, 5128, 5129, 5130, 5131, 513
2, 5133, 5134, 5135, 5136, 5137, 5138, 5139, 5140, 5141, 5142, 5143,
5144, 5145, 5146, 5147, 5148, 5149, 5150, 5151, 5152, 5153, 5154, 51
55, 5156, 5157, 5158, 5159, 5160, 5161, 5162, 5163, 5164, 5165, 516
6, 5167, 5168, 5169, 5170, 5171, 5172, 5173, 5174, 5175, 5176, 5177,
5178, 5179, 5180, 5181, 5182, 5183, 5184, 5185, 5186, 5187, 5188, 51
89, 5190, 5191, 5192, 5193, 5194, 5195, 5196, 5197, 5198, 5199, 520
0, 5201, 5202, 5203, 5204, 5205, 5206, 5207, 5208, 5209, 5210, 5211,
5212, 5213, 5214, 5215, 5216, 5217, 5218, 5219, 5220, 5221, 5222, 52
23, 5224, 5225, 5226, 5227, 5228, 5229, 5230, 5231, 5232, 5233, 523
4, 5235, 5236, 5237, 5238, 5239, 5240, 5241, 5242, 5243, 5244, 5245,
5246, 5247, 5248, 5249, 5250, 5251, 5252, 5253, 5254, 5255, 5256, 52
57, 5258, 5259, 5260, 5261, 5262, 5263, 5264, 5265, 5266, 5267, 526
8, 5269, 5270, 5271, 5272, 5273, 5274, 5275, 5276, 5277, 5278, 5279,
5280, 5281, 5282, 5283, 5284, 5285, 5286, 5287, 5288, 5289, 5290, 52
91, 5292, 5293, 5294, 5295, 5296, 5297, 5298, 5299, 5300, 5301, 530
2, 5303, 5304, 5305, 5306, 5307, 5308, 5309, 5310, 5311, 5312, 5313,
5314, 5315, 5316, 5317, 5318, 5319, 5320, 5321, 5322, 5323, 5324, 53
25, 5326, 5327, 5328, 5329, 5330, 5331, 5332, 5333, 5334, 5335, 533
6, 5337, 5338, 5339, 5340, 5341, 5342, 5343, 5344, 5345, 5346, 5347,
5348, 5349, 5350, 5351, 5352, 5353, 5354, 5355, 5356, 5357, 5358, 53
59, 5360, 5361, 5362, 5363, 5364, 5365, 5366, 5367, 5368, 5369, 537
0, 5371, 5372, 5373, 5374, 5375, 5376, 5377, 5378, 5379, 5380, 5381,
5382, 5383, 5384, 5385, 5386, 5387, 5388, 5389, 5390, 5391, 5392, 53
93, 5394, 5395, 5396, 5397, 5398, 5399, 5400, 5401, 5402, 5403, 540
4, 5405, 5406, 5407, 5408, 5409, 5410, 5411, 5412, 5413, 5414, 5415,
5416, 5417, 5418, 5419, 5420, 5421, 5422, 5423, 5424, 5425, 5426, 54
27, 5428, 5429, 5430, 5431, 5432, 5433, 5434, 5435, 5436, 5437, 543
8, 5439, 5440, 5441, 5442, 5443, 5444, 5445, 5446, 5447, 5448, 5449,
5450, 5451, 5452, 5453, 5454, 5455, 5456, 5457, 5458, 5459, 5460, 54
61, 5462, 5463, 5464, 5465, 5466, 5467, 5468, 5469, 5470, 5471, 547
2, 5473, 5474, 5475, 5476, 5477, 5478, 5479, 5480, 5481, 5482, 5483,
5484, 5485, 5486, 5487, 5488, 5489, 5490, 5491, 5492, 5493, 5494, 54
95, 5496, 5497, 5498, 5499, 5500, 5501, 5502, 5503, 5504, 5505, 550
6, 5507, 5508, 5509, 5510, 5511, 5512, 5513, 5514, 5515, 5516, 5517,
5518, 5519, 5520, 5521, 5522, 5523, 5524, 5525, 5526, 5527, 5528, 55
29, 5530, 5531, 5532, 5533, 5534, 5535, 5536, 5537, 5538, 5539, 554
0, 5541, 5542, 5543, 5544, 5545, 5546, 5547, 5548, 5549, 5550, 5551,
5552, 5553, 5554, 5555, 5556, 5557, 5558, 5559, 5560, 5561, 5562, 55
63, 5564, 5565, 5566, 5567, 5568, 5569, 5570, 5571, 5572, 5573, 557
4, 5575, 5576, 5577, 5578, 5579, 5580, 5581, 5582, 5583, 5584, 5585,
5586, 5587, 5588, 5589, 5590, 5591, 5592, 5593, 5594, 5595, 5596, 55
97, 5598, 5599, 5600, 5601, 5602, 5603, 5604, 5605, 5606, 5607, 560
8, 5609, 5610, 5611, 5612, 5613, 5614, 5615, 5616, 5617, 5618, 5619,
5620, 5621, 5622, 5623, 5624, 5625, 5626, 5627, 5628, 5629, 5630, 56
31, 5632, 5633, 5634, 5635, 5636, 5637, 5638, 5639, 5640, 5641, 564
2, 5643, 5644, 5645, 5646, 5647, 5648, 5649, 5650, 5651, 5652, 5653,
5654, 5655, 5656, 5657, 5658, 5659, 5660, 5661, 5662, 5663, 5664, 56
65, 5666, 5667, 5668, 5669, 5670, 5671, 5672, 5673, 5674, 5675, 567
6, 5677, 5678, 5679, 5680, 5681, 5682, 5683, 5684, 5685, 5686, 5687,
5688, 5689, 5690, 5691, 5692, 5693, 5694, 5695, 5696, 5697, 5698, 56
99, 5700, 5701, 5702, 5703, 5704, 5705, 5706, 5707, 5708, 5709, 571

0, 5711, 5712, 5713, 5714, 5715, 5716, 5717, 5718, 5719, 5720, 5721,
5722, 5723, 5724, 5725, 5726, 5727, 5728, 5729, 5730, 5731, 5732, 57
33, 5734, 5735, 5736, 5737, 5738, 5739, 5740, 5741, 5742, 5743, 574
4, 5745, 5746, 5747, 5748, 5749, 5750, 5751, 5752, 5753, 5754, 5755,
5756, 5757, 5758, 5759, 5760, 5761, 5762, 5763, 5764, 5765, 5766, 57
67, 5768, 5769, 5770, 5771, 5772, 5773, 5774, 5775, 5776, 5777, 577
8, 5779, 5780, 5781, 5782, 5783, 5784, 5785, 5786, 5787, 5788, 5789,
5790, 5791, 5792, 5793, 5794, 5795, 5796, 5797, 5798, 5799, 5800, 58
01, 5802, 5803, 5804, 5805, 5806, 5807, 5808, 5809, 5810, 5811, 581
2, 5813, 5814, 5815, 5816, 5817, 5818, 5819, 5820, 5821, 5822, 5823,
5824, 5825, 5826, 5827, 5828, 5829, 5830, 5831, 5832, 5833, 5834, 58
35, 5836, 5837, 5838, 5839, 5840, 5841, 5842, 5843, 5844, 5845, 584
6, 5847, 5848, 5849, 5850, 5851, 5852, 5853, 5854, 5855, 5856, 5857,
5858, 5859, 5860, 5861, 5862, 5863, 5864, 5865, 5866, 5867, 5868, 58
69, 5870, 5871, 5872, 5873, 5874, 5875, 5876, 5877, 5878, 5879, 588
0, 5881, 5882, 5883, 5884, 5885, 5886, 5887, 5888, 5889, 5890, 5891,
5892, 5893, 5894, 5895, 5896, 5897, 5898, 5899, 5900, 5901, 5902, 59
03, 5904, 5905, 5906, 5907, 5908, 5909, 5910, 5911, 5912, 5913, 591
4, 5915, 5916, 5917, 5918, 5919, 5920, 5921, 5922, 5923, 5924, 5925,
5926, 5927, 5928, 5929, 5930, 5931, 5932, 5933, 5934, 5935, 5936, 59
37, 5938, 5939, 5940, 5941, 5942, 5943, 5944, 5945, 5946, 5947, 594
8, 5949, 5950, 5951, 5952, 5953, 5954, 5955, 5956, 5957, 5958, 5959,
5960, 5961, 5962, 5963, 5964, 5965, 5966, 5967, 5968, 5969, 5970, 59
71, 5972, 5973, 5974, 5975, 5976, 5977, 5978, 5979, 5980, 5981, 598
2, 5983, 5984, 5985, 5986, 5987, 5988, 5989, 5990, 5991, 5992, 5993,
5994, 5995, 5996, 5997, 5998, 5999, 6000, 6001, 6002, 6003, 6004, 60
05, 6006, 6007, 6008, 6009, 6010, 6011, 6012, 6013, 6014, 6015, 601
6, 6017, 6018, 6019, 6020, 6021, 6022, 6023, 6024, 6025, 6026, 6027,
6028, 6029, 6030, 6031, 6032, 6033, 6034, 6035, 6036, 6037, 6038, 60
39, 6040, 6041, 6042, 6043, 6044, 6045, 6046, 6047, 6048, 6049, 605
0, 6051, 6052, 6053, 6054, 6055, 6056, 6057, 6058, 6059, 6060, 6061,
6062, 6063, 6064, 6065, 6066, 6067, 6068, 6069, 6070, 6071, 6072, 60
73, 6074, 6075, 6076, 6077, 6078, 6079, 6080, 6081, 6082, 6083, 608
4, 6085, 6086, 6087, 6088, 6089, 6090, 6091, 6092, 6093, 6094, 6095,
6096, 6097, 6098, 6099, 6100, 6101, 6102, 6103, 6104, 6105, 6106, 61
07, 6108, 6109, 6110, 6111, 6112, 6113, 6114, 6115, 6116, 6117, 611
8, 6119, 6120, 6121, 6122, 6123, 6124, 6125, 6126, 6127, 6128, 6129,
6130, 6131, 6132, 6133, 6134, 6135, 6136, 6137, 6138, 6139, 6140, 61
41, 6142, 6143, 6144, 6145, 6146, 6147, 6148, 6149, 6150, 6151, 615
2, 6153, 6154, 6155, 6156, 6157, 6158, 6159, 6160, 6161, 6162, 6163,
6164, 6165, 6166, 6167, 6168, 6169, 6170, 6171, 6172, 6173, 6174, 61
75, 6176, 6177, 6178, 6179, 6180, 6181, 6182, 6183, 6184, 6185, 618
6, 6187, 6188, 6189, 6190, 6191, 6192, 6193, 6194, 6195, 6196, 6197,
6198, 6199, 6200, 6201, 6202, 6203, 6204, 6205, 6206, 6207, 6208, 62
09, 6210, 6211, 6212, 6213, 6214, 6215, 6216, 6217, 6218, 6219, 622
0, 6221, 6222, 6223, 6224, 6225, 6226, 6227, 6228, 6229, 6230, 6231,
6232, 6233, 6234, 6235, 6236, 6237, 6238, 6239, 6240, 6241, 6242, 62
43, 6244, 6245, 6246, 6247, 6248, 6249, 6250, 6251, 6252, 6253, 625
4, 6255, 6256, 6257, 6258, 6259, 6260, 6261, 6262, 6263, 6264, 6265,
6266, 6267, 6268, 6269, 6270, 6271, 6272, 6273, 6274, 6275, 6276, 62
77, 6278, 6279, 6280, 6281, 6282, 6283, 6284, 6285, 6286, 6287, 628
8, 6289, 6290, 6291, 6292, 6293, 6294, 6295, 6296, 6297, 6298, 6299,
6300, 6301, 6302, 6303, 6304, 6305, 6306, 6307, 6308, 6309, 6310, 63
11, 6312, 6313, 6314, 6315, 6316, 6317, 6318, 6319, 6320, 6321, 632
2, 6323, 6324, 6325, 6326, 6327, 6328, 6329, 6330, 6331, 6332, 6333,
6334, 6335, 6336, 6337, 6338, 6339, 6340, 6341, 6342, 6343, 6344, 63
45, 6346, 6347, 6348, 6349, 6350, 6351, 6352, 6353, 6354, 6355, 635
6, 6357, 6358, 6359, 6360, 6361, 6362, 6363, 6364, 6365, 6366, 6367,
6368, 6369, 6370, 6371, 6372, 6373, 6374, 6375, 6376, 6377, 6378, 63
79, 6380, 6381, 6382, 6383, 6384, 6385, 6386, 6387, 6388, 6389, 639
0, 6391, 6392, 6393, 6394, 6395, 6396, 6397, 6398, 6399, 6400, 6401,

6402, 6403, 6404, 6405, 6406, 6407, 6408, 6409, 6410, 6411, 6412, 64
13, 6414, 6415, 6416, 6417, 6418, 6419, 6420, 6421, 6422, 6423, 642
4, 6425, 6426, 6427, 6428, 6429, 6430, 6431, 6432, 6433, 6434, 6435,
6436, 6437, 6438, 6439, 6440, 6441, 6442, 6443, 6444, 6445, 6446, 64
47, 6448, 6449, 6450, 6451, 6452, 6453, 6454, 6455, 6456, 6457, 645
8, 6459, 6460, 6461, 6462, 6463, 6464, 6465, 6466, 6467, 6468, 6469,
6470, 6471, 6472, 6473, 6474, 6475, 6476, 6477, 6478, 6479, 6480, 64
81, 6482, 6483, 6484, 6485, 6486, 6487, 6488, 6489, 6490, 6491, 649
2, 6493, 6494, 6495, 6496, 6497, 6498, 6499, 6500, 6501, 6502, 6503,
6504, 6505, 6506, 6507, 6508, 6509, 6510, 6511, 6512, 6513, 6514, 65
15, 6516, 6517, 6518, 6519, 6520, 6521, 6522, 6523, 6524, 6525, 652
6, 6527, 6528, 6529, 6530, 6531, 6532, 6533, 6534, 6535, 6536, 6537,
6538, 6539, 6540, 6541, 6542, 6543, 6544, 6545, 6546, 6547, 6548, 65
49, 6550, 6551, 6552, 6553, 6554, 6555, 6556, 6557, 6558, 6559, 656
0, 6561, 6562, 6563, 6564, 6565, 6566, 6567, 6568, 6569, 6570, 6571,
6572, 6573, 6574, 6575, 6576, 6577, 6578, 6579, 6580, 6581, 6582, 65
83, 6584, 6585, 6586, 6587, 6588, 6589, 6590, 6591, 6592, 6593, 659
4, 6595, 6596, 6597, 6598, 6599, 6600, 6601, 6602, 6603, 6604, 6605,
6606, 6607, 6608, 6609, 6610, 6611, 6612, 6613, 6614, 6615, 6616, 66
17, 6618, 6619, 6620, 6621, 6622, 6623, 6624, 6625, 6626, 6627, 662
8, 6629, 6630, 6631, 6632, 6633, 6634, 6635, 6636, 6637, 6638, 6639,
6640, 6641, 6642, 6643, 6644, 6645, 6646, 6647, 6648, 6649, 6650, 66
51, 6652, 6653, 6654, 6655, 6656, 6657, 6658, 6659, 6660, 6661, 666
2, 6663, 6664, 6665, 6666, 6667, 6668, 6669, 6670, 6671, 6672, 6673,
6674, 6675, 6676, 6677, 6678, 6679, 6680, 6681, 6682, 6683, 6684, 66
85, 6686, 6687, 6688, 6689, 6690, 6691, 6692, 6693, 6694, 6695, 669
6, 6697, 6698, 6699, 6700, 6701, 6702, 6703, 6704, 6705, 6706, 6707,
6708, 6709, 6710, 6711, 6712, 6713, 6714, 6715, 6716, 6717, 6718, 67
19, 6720, 6721, 6722, 6723, 6724, 6725, 6726, 6727, 6728, 6729, 673
0, 6731, 6732, 6733, 6734, 6735, 6736, 6737, 6738, 6739, 6740, 6741,
6742, 6743, 6744, 6745, 6746, 6747, 6748, 6749, 6750, 6751, 6752, 67
53, 6754, 6755, 6756, 6757, 6758, 6759, 6760, 6761, 6762, 6763, 676
4, 6765, 6766, 6767, 6768, 6769, 6770, 6771, 6772, 6773, 6774, 6775,
6776, 6777, 6778, 6779, 6780, 6781, 6782, 6783, 6784, 6785, 6786, 67
87, 6788, 6789, 6790, 6791, 6792, 6793, 6794, 6795, 6796, 6797, 679
8, 6799, 6800, 6801, 6802, 6803, 6804, 6805, 6806, 6807, 6808, 6809,
6810, 6811, 6812, 6813, 6814, 6815, 6816, 6817, 6818, 6819, 6820, 68
21, 6822, 6823, 6824, 6825, 6826, 6827, 6828, 6829, 6830, 6831, 683
2, 6833, 6834, 6835, 6836, 6837, 6838, 6839, 6840, 6841, 6842, 6843,
6844, 6845, 6846, 6847, 6848, 6849, 6850, 6851, 6852, 6853, 6854, 68
55, 6856, 6857, 6858, 6859, 6860, 6861, 6862, 6863, 6864, 6865, 686
6, 6867, 6868, 6869, 6870, 6871, 6872, 6873, 6874, 6875, 6876, 6877,
6878, 6879, 6880, 6881, 6882, 6883, 6884, 6885, 6886, 6887, 6888, 68
89, 6890, 6891, 6892, 6893, 6894, 6895, 6896, 6897, 6898, 6899, 690
0, 6901, 6902, 6903, 6904, 6905, 6906, 6907, 6908, 6909, 6910, 6911,
6912, 6913, 6914, 6915, 6916, 6917, 6918, 6919, 6920, 6921, 6922, 69
23, 6924, 6925, 6926, 6927, 6928, 6929, 6930, 6931, 6932, 6933, 693
4, 6935, 6936, 6937, 6938, 6939, 6940, 6941, 6942, 6943, 6944, 6945,
6946, 6947, 6948, 6949, 6950, 6951, 6952, 6953, 6954, 6955, 6956, 69
57, 6958, 6959, 6960, 6961, 6962, 6963, 6964, 6965, 6966, 6967, 696
8, 6969, 6970, 6971, 6972, 6973, 6974, 6975, 6976, 6977, 6978, 6979,
6980, 6981, 6982, 6983, 6984, 6985, 6986, 6987, 6988, 6989, 6990, 69
91, 6992, 6993, 6994, 6995, 6996, 6997, 6998, 6999, 7000, 7001, 700
2, 7003, 7004, 7005, 7006, 7007, 7008, 7009, 7010, 7011, 7012, 7013,
7014, 7015, 7016, 7017, 7018, 7019, 7020, 7021, 7022, 7023, 7024, 70
25, 7026, 7027, 7028, 7029, 7030, 7031, 7032, 7033, 7034, 7035, 703
6, 7037, 7038, 7039, 7040, 7041, 7042, 7043, 7044, 7045, 7046, 7047,
7048, 7049, 7050, 7051, 7052, 7053, 7054, 7055, 7056, 7057, 7058, 70
59, 7060, 7061, 7062, 7063, 7064, 7065, 7066, 7067, 7068, 7069, 707
0, 7071, 7072, 7073, 7074, 7075, 7076, 7077, 7078, 7079, 7080, 7081,
7082, 7083, 7084, 7085, 7086, 7087, 7088, 7089, 7090, 7091, 7092, 70

93, 7094, 7095, 7096, 7097, 7098, 7099, 7100, 7101, 7102, 7103, 710
4, 7105, 7106, 7107, 7108, 7109, 7110, 7111, 7112, 7113, 7114, 7115,
7116, 7117, 7118, 7119, 7120, 7121, 7122, 7123, 7124, 7125, 7126, 71
27, 7128, 7129, 7130, 7131, 7132, 7133, 7134, 7135, 7136, 7137, 713
8, 7139, 7140, 7141, 7142, 7143, 7144, 7145, 7146, 7147, 7148, 7149,
7150, 7151, 7152, 7153, 7154, 7155, 7156, 7157, 7158, 7159, 7160, 71
61, 7162, 7163, 7164, 7165, 7166, 7167, 7168, 7169, 7170, 7171, 717
2, 7173, 7174, 7175, 7176, 7177, 7178, 7179, 7180, 7181, 7182, 7183,
7184, 7185, 7186, 7187, 7188, 7189, 7190, 7191, 7192, 7193, 7194, 71
95, 7196, 7197, 7198, 7199, 7200, 7201, 7202, 7203, 7204, 7205, 720
6, 7207, 7208, 7209, 7210, 7211, 7212, 7213, 7214, 7215, 7216, 7217,
7218, 7219, 7220, 7221, 7222, 7223, 7224, 7225, 7226, 7227, 7228, 72
29, 7230, 7231, 7232, 7233, 7234, 7235, 7236, 7237, 7238, 7239, 724
0, 7241, 7242, 7243, 7244, 7245, 7246, 7247, 7248, 7249, 7250, 7251,
7252, 7253, 7254, 7255, 7256, 7257, 7258, 7259, 7260, 7261, 7262, 72
63, 7264, 7265, 7266, 7267, 7268, 7269, 7270, 7271, 7272, 7273, 727
4, 7275, 7276, 7277, 7278, 7279, 7280, 7281, 7282, 7283, 7284, 7285,
7286, 7287, 7288, 7289, 7290, 7291, 7292, 7293, 7294, 7295, 7296, 72
97, 7298, 7299, 7300, 7301, 7302, 7303, 7304, 7305, 7306, 7307, 730
8, 7309, 7310, 7311, 7312, 7313, 7314, 7315, 7316, 7317, 7318, 7319,
7320, 7321, 7322, 7323, 7324, 7325, 7326, 7327, 7328, 7329, 7330, 73
31, 7332, 7333, 7334, 7335, 7336, 7337, 7338, 7339, 7340, 7341, 734
2, 7343, 7344, 7345, 7346, 7347, 7348, 7349, 7350, 7351, 7352, 7353,
7354, 7355, 7356, 7357, 7358, 7359, 7360, 7361, 7362, 7363, 7364, 73
65, 7366, 7367, 7368, 7369, 7370, 7371, 7372, 7373, 7374, 7375, 737
6, 7377, 7378, 7379, 7380, 7381, 7382, 7383, 7384, 7385, 7386, 7387,
7388, 7389, 7390, 7391, 7392, 7393, 7394, 7395, 7396, 7397, 7398, 73
99, 7400, 7401, 7402, 7403, 7404, 7405, 7406, 7407, 7408, 7409, 741
0, 7411, 7412, 7413, 7414, 7415, 7416, 7417, 7418, 7419, 7420, 7421,
7422, 7423, 7424, 7425, 7426, 7427, 7428, 7429, 7430, 7431, 7432, 74
33, 7434, 7435, 7436, 7437, 7438, 7439, 7440, 7441, 7442, 7443, 744
4, 7445, 7446, 7447, 7448, 7449, 7450, 7451, 7452, 7453, 7454, 7455,
7456, 7457, 7458, 7459, 7460, 7461, 7462, 7463, 7464, 7465, 7466, 74
67, 7468, 7469, 7470, 7471, 7472, 7473, 7474, 7475, 7476, 7477, 747
8, 7479, 7480, 7481, 7482, 7483, 7484, 7485, 7486, 7487, 7488, 7489,
7490, 7491, 7492, 7493, 7494, 7495, 7496, 7497, 7498, 7499, 7500, 75
01, 7502, 7503, 7504, 7505, 7506, 7507, 7508, 7509, 7510, 7511, 751
2, 7513, 7514, 7515, 7516, 7517, 7518, 7519, 7520, 7521, 7522, 7523,
7524, 7525, 7526, 7527, 7528, 7529, 7530, 7531, 7532, 7533, 7534, 75
35, 7536, 7537, 7538, 7539, 7540, 7541, 7542, 7543, 7544, 7545, 754
6, 7547, 7548, 7549, 7550, 7551, 7552, 7553, 7554, 7555, 7556, 7557,
7558, 7559, 7560, 7561, 7562, 7563, 7564, 7565, 7566, 7567, 7568, 75
69, 7570, 7571, 7572, 7573, 7574, 7575, 7576, 7577, 7578, 7579, 758
0, 7581, 7582, 7583, 7584, 7585, 7586, 7587, 7588, 7589, 7590, 7591,
7592, 7593, 7594, 7595, 7596, 7597, 7598, 7599, 7600, 7601, 7602, 76
03, 7604, 7605, 7606, 7607, 7608, 7609, 7610, 7611, 7612, 7613, 761
4, 7615, 7616, 7617, 7618, 7619, 7620, 7621, 7622, 7623, 7624, 7625,
7626, 7627, 7628, 7629, 7630, 7631, 7632, 7633, 7634, 7635, 7636, 76
37, 7638, 7639, 7640, 7641, 7642, 7643, 7644, 7645, 7646, 7647, 764
8, 7649, 7650, 7651, 7652, 7653, 7654, 7655, 7656, 7657, 7658, 7659,
7660, 7661, 7662, 7663, 7664, 7665, 7666, 7667, 7668, 7669, 7670, 76
71, 7672, 7673, 7674, 7675, 7676, 7677, 7678, 7679, 7680, 7681, 768
2, 7683, 7684, 7685, 7686, 7687, 7688, 7689, 7690, 7691, 7692, 7693,
7694, 7695, 7696, 7697, 7698, 7699, 7700, 7701, 7702, 7703, 7704, 77
05, 7706, 7707, 7708, 7709, 7710, 7711, 7712, 7713, 7714, 7715, 771
6, 7717, 7718, 7719, 7720, 7721, 7722, 7723, 7724, 7725, 7726, 7727,
7728, 7729, 7730, 7731, 7732, 7733, 7734, 7735, 7736, 7737, 7738, 77
39, 7740, 7741, 7742, 7743, 7744, 7745, 7746, 7747, 7748, 7749, 775
0, 7751, 7752, 7753, 7754, 7755, 7756, 7757, 7758, 7759, 7760, 7761,
7762, 7763, 7764, 7765, 7766, 7767, 7768, 7769, 7770, 7771, 7772, 77
73, 7774, 7775, 7776, 7777, 7778, 7779, 7780, 7781, 7782, 7783, 778

4, 7785, 7786, 7787, 7788, 7789, 7790, 7791, 7792, 7793, 7794, 7795,
7796, 7797, 7798, 7799, 7800, 7801, 7802, 7803, 7804, 7805, 7806, 78
07, 7808, 7809, 7810, 7811, 7812, 7813, 7814, 7815, 7816, 7817, 781
8, 7819, 7820, 7821, 7822, 7823, 7824, 7825, 7826, 7827, 7828, 7829,
7830, 7831, 7832, 7833, 7834, 7835, 7836, 7837, 7838, 7839, 7840, 78
41, 7842, 7843, 7844, 7845, 7846, 7847, 7848, 7849, 7850, 7851, 785
2, 7853, 7854, 7855, 7856, 7857, 7858, 7859, 7860, 7861, 7862, 7863,
7864, 7865, 7866, 7867, 7868, 7869, 7870, 7871, 7872, 7873, 7874, 78
75, 7876, 7877, 7878, 7879, 7880, 7881, 7882, 7883, 7884, 7885, 788
6, 7887, 7888, 7889, 7890, 7891, 7892, 7893, 7894, 7895, 7896, 7897,
7898, 7899, 7900, 7901, 7902, 7903, 7904, 7905, 7906, 7907, 7908, 79
09, 7910, 7911, 7912, 7913, 7914, 7915, 7916, 7917, 7918, 7919, 792
0, 7921, 7922, 7923, 7924, 7925, 7926, 7927, 7928, 7929, 7930, 7931,
7932, 7933, 7934, 7935, 7936, 7937, 7938, 7939, 7940, 7941, 7942, 79
43, 7944, 7945, 7946, 7947, 7948, 7949, 7950, 7951, 7952, 7953, 795
4, 7955, 7956, 7957, 7958, 7959, 7960, 7961, 7962, 7963, 7964, 7965,
7966, 7967, 7968, 7969, 7970, 7971, 7972, 7973, 7974, 7975, 7976, 79
77, 7978, 7979, 7980, 7981, 7982, 7983, 7984, 7985, 7986, 7987, 798
8, 7989, 7990, 7991, 7992, 7993, 7994, 7995, 7996, 7997, 7998, 7999,
8000, 8001, 8002, 8003, 8004, 8005, 8006, 8007, 8008, 8009, 8010, 80
11, 8012, 8013, 8014, 8015, 8016, 8017, 8018, 8019, 8020, 8021, 802
2, 8023, 8024, 8025, 8026, 8027, 8028, 8029, 8030, 8031, 8032, 8033,
8034, 8035, 8036, 8037, 8038, 8039, 8040, 8041, 8042, 8043, 8044, 80
45, 8046, 8047, 8048, 8049, 8050, 8051, 8052, 8053, 8054, 8055, 805
6, 8057, 8058, 8059, 8060, 8061, 8062, 8063, 8064, 8065, 8066, 8067,
8068, 8069, 8070, 8071, 8072, 8073, 8074, 8075, 8076, 8077, 8078, 80
79, 8080, 8081, 8082, 8083, 8084, 8085, 8086, 8087, 8088, 8089, 809
0, 8091, 8092, 8093, 8094, 8095, 8096, 8097, 8098, 8099, 8100, 8101,
8102, 8103, 8104, 8105, 8106, 8107, 8108, 8109, 8110, 8111, 8112, 81
13, 8114, 8115, 8116, 8117, 8118, 8119, 8120, 8121, 8122, 8123, 812
4, 8125, 8126, 8127, 8128, 8129, 8130, 8131, 8132, 8133, 8134, 8135,
8136, 8137, 8138, 8139, 8140, 8141, 8142, 8143, 8144, 8145, 8146, 81
47, 8148, 8149, 8150, 8151, 8152, 8153, 8154, 8155, 8156, 8157, 815
8, 8159, 8160, 8161, 8162, 8163, 8164, 8165, 8166, 8167, 8168, 8169,
8170, 8171, 8172, 8173, 8174, 8175, 8176, 8177, 8178, 8179, 8180, 81
81, 8182, 8183, 8184, 8185, 8186, 8187, 8188, 8189, 8190, 8191, 819
2, 8193, 8194, 8195, 8196, 8197, 8198, 8199, 8200, 8201, 8202, 8203,
8204, 8205, 8206, 8207, 8208, 8209, 8210, 8211, 8212, 8213, 8214, 82
15, 8216, 8217, 8218, 8219, 8220, 8221, 8222, 8223, 8224, 8225, 822
6, 8227, 8228, 8229, 8230, 8231, 8232, 8233, 8234, 8235, 8236, 8237,
8238, 8239, 8240, 8241, 8242, 8243, 8244, 8245, 8246, 8247, 8248, 82
49, 8250, 8251, 8252, 8253, 8254, 8255, 8256, 8257, 8258, 8259, 826
0, 8261, 8262, 8263, 8264, 8265, 8266, 8267, 8268, 8269, 8270, 8271,
8272, 8273, 8274, 8275, 8276, 8277, 8278, 8279, 8280, 8281, 8282, 82
83, 8284, 8285, 8286, 8287, 8288, 8289, 8290, 8291, 8292, 8293, 829
4, 8295, 8296, 8297, 8298, 8299, 8300, 8301, 8302, 8303, 8304, 8305,
8306, 8307, 8308, 8309, 8310, 8311, 8312, 8313, 8314, 8315, 8316, 83
17, 8318, 8319, 8320, 8321, 8322, 8323, 8324, 8325, 8326, 8327, 832
8, 8329, 8330, 8331, 8332, 8333, 8334, 8335, 8336, 8337, 8338, 8339,
8340, 8341, 8342, 8343, 8344, 8345, 8346, 8347, 8348, 8349, 8350, 83
51, 8352, 8353, 8354, 8355, 8356, 8357, 8358, 8359, 8360, 8361, 836
2, 8363, 8364, 8365, 8366, 8367, 8368, 8369, 8370, 8371, 8372, 8373,
8374, 8375, 8376, 8377, 8378, 8379, 8380, 8381, 8382, 8383, 8384, 83
85, 8386, 8387, 8388, 8389, 8390, 8391, 8392, 8393, 8394, 8395, 839
6, 8397, 8398, 8399, 8400, 8401, 8402, 8403, 8404, 8405, 8406, 8407,
8408, 8409, 8410, 8411, 8412, 8413, 8414, 8415, 8416, 8417, 8418, 84
19, 8420, 8421, 8422, 8423, 8424, 8425, 8426, 8427, 8428, 8429, 843
0, 8431, 8432, 8433, 8434, 8435, 8436, 8437, 8438, 8439, 8440, 8441,
8442, 8443, 8444, 8445, 8446, 8447, 8448, 8449, 8450, 8451, 8452, 84
53, 8454, 8455, 8456, 8457, 8458, 8459, 8460, 8461, 8462, 8463, 846
4, 8465, 8466, 8467, 8468, 8469, 8470, 8471, 8472, 8473, 8474, 8475,

8476, 8477, 8478, 8479, 8480, 8481, 8482, 8483, 8484, 8485, 8486, 84
87, 8488, 8489, 8490, 8491, 8492, 8493, 8494, 8495, 8496, 8497, 849
8, 8499, 8500, 8501, 8502, 8503, 8504, 8505, 8506, 8507, 8508, 8509,
8510, 8511, 8512, 8513, 8514, 8515, 8516, 8517, 8518, 8519, 8520, 85
21, 8522, 8523, 8524, 8525, 8526, 8527, 8528, 8529, 8530, 8531, 853
2, 8533, 8534, 8535, 8536, 8537, 8538, 8539, 8540, 8541, 8542, 8543,
8544, 8545, 8546, 8547, 8548, 8549, 8550, 8551, 8552, 8553, 8554, 85
55, 8556, 8557, 8558, 8559, 8560, 8561, 8562, 8563, 8564, 8565, 856
6, 8567, 8568, 8569, 8570, 8571, 8572, 8573, 8574, 8575, 8576, 8577,
8578, 8579, 8580, 8581, 8582, 8583, 8584, 8585, 8586, 8587, 8588, 85
89, 8590, 8591, 8592, 8593, 8594, 8595, 8596, 8597, 8598, 8599, 860
0, 8601, 8602, 8603, 8604, 8605, 8606, 8607, 8608, 8609, 8610, 8611,
8612, 8613, 8614, 8615, 8616, 8617, 8618, 8619, 8620, 8621, 8622, 86
23, 8624, 8625, 8626, 8627, 8628, 8629, 8630, 8631, 8632, 8633, 863
4, 8635, 8636, 8637, 8638, 8639, 8640, 8641, 8642, 8643, 8644, 8645,
8646, 8647, 8648, 8649, 8650, 8651, 8652, 8653, 8654, 8655, 8656, 86
57, 8658, 8659, 8660, 8661, 8662, 8663, 8664, 8665, 8666, 8667, 866
8, 8669, 8670, 8671, 8672, 8673, 8674, 8675, 8676, 8677, 8678, 8679,
8680, 8681, 8682, 8683, 8684, 8685, 8686, 8687, 8688, 8689, 8690, 86
91, 8692, 8693, 8694, 8695, 8696, 8697, 8698, 8699, 8700, 8701, 870
2, 8703, 8704, 8705, 8706, 8707, 8708, 8709, 8710, 8711, 8712, 8713,
8714, 8715, 8716, 8717, 8718, 8719, 8720, 8721, 8722, 8723, 8724, 87
25, 8726, 8727, 8728, 8729, 8730, 8731, 8732, 8733, 8734, 8735, 873
6, 8737, 8738, 8739, 8740, 8741, 8742, 8743, 8744, 8745, 8746, 8747,
8748, 8749, 8750, 8751, 8752, 8753, 8754, 8755, 8756, 8757, 8758, 87
59, 8760, 8761, 8762, 8763, 8764, 8765, 8766, 8767, 8768, 8769, 877
0, 8771, 8772, 8773, 8774, 8775, 8776, 8777, 8778, 8779, 8780, 8781,
8782, 8783, 8784, 8785, 8786, 8787, 8788, 8789, 8790, 8791, 8792, 87
93, 8794, 8795, 8796, 8797, 8798, 8799, 8800, 8801, 8802, 8803, 880
4, 8805, 8806, 8807, 8808, 8809, 8810, 8811, 8812, 8813, 8814, 8815,
8816, 8817, 8818, 8819, 8820, 8821, 8822, 8823, 8824, 8825, 8826, 88
27, 8828, 8829, 8830, 8831, 8832, 8833, 8834, 8835, 8836, 8837, 883
8, 8839, 8840, 8841, 8842, 8843, 8844, 8845, 8846, 8847, 8848, 8849,
8850, 8851, 8852, 8853, 8854, 8855, 8856, 8857, 8858, 8859, 8860, 88
61, 8862, 8863, 8864, 8865, 8866, 8867, 8868, 8869, 8870, 8871, 887
2, 8873, 8874, 8875, 8876, 8877, 8878, 8879, 8880, 8881, 8882, 8883,
8884, 8885, 8886, 8887, 8888, 8889, 8890, 8891, 8892, 8893, 8894, 88
95, 8896, 8897, 8898, 8899, 8900, 8901, 8902, 8903, 8904, 8905, 890
6, 8907, 8908, 8909, 8910, 8911, 8912, 8913, 8914, 8915, 8916, 8917,
8918, 8919, 8920, 8921, 8922, 8923, 8924, 8925, 8926, 8927, 8928, 89
29, 8930, 8931, 8932, 8933, 8934, 8935, 8936, 8937, 8938, 8939, 894
0, 8941, 8942, 8943, 8944, 8945, 8946, 8947, 8948, 8949, 8950, 8951,
8952, 8953, 8954, 8955, 8956, 8957, 8958, 8959, 8960, 8961, 8962, 89
63, 8964, 8965, 8966, 8967, 8968, 8969, 8970, 8971, 8972, 8973, 897
4, 8975, 8976, 8977, 8978, 8979, 8980, 8981, 8982, 8983, 8984, 8985,
8986, 8987, 8988, 8989, 8990, 8991, 8992, 8993, 8994, 8995, 8996, 89
97, 8998, 8999, 9000, 9001, 9002, 9003, 9004, 9005, 9006, 9007, 900
8, 9009, 9010, 9011, 9012, 9013, 9014, 9015, 9016, 9017, 9018, 9019,
9020, 9021, 9022, 9023, 9024, 9025, 9026, 9027, 9028, 9029, 9030, 90
31, 9032, 9033, 9034, 9035, 9036, 9037, 9038, 9039, 9040, 9041, 904
2, 9043, 9044, 9045, 9046, 9047, 9048, 9049, 9050, 9051, 9052, 9053,
9054, 9055, 9056, 9057, 9058, 9059, 9060, 9061, 9062, 9063, 9064, 90
65, 9066, 9067, 9068, 9069, 9070, 9071, 9072, 9073, 9074, 9075, 907
6, 9077, 9078, 9079, 9080, 9081, 9082, 9083, 9084, 9085, 9086, 9087,
9088, 9089, 9090, 9091, 9092, 9093, 9094, 9095, 9096, 9097, 9098, 90
99, 9100, 9101, 9102, 9103, 9104, 9105, 9106, 9107, 9108, 9109, 911
0, 9111, 9112, 9113, 9114, 9115, 9116, 9117, 9118, 9119, 9120, 9121,
9122, 9123, 9124, 9125, 9126, 9127, 9128, 9129, 9130, 9131, 9132, 91
33, 9134, 9135, 9136, 9137, 9138, 9139, 9140, 9141, 9142, 9143, 914
4, 9145, 9146, 9147, 9148, 9149, 9150, 9151, 9152, 9153, 9154, 9155,
9156, 9157, 9158, 9159, 9160, 9161, 9162, 9163, 9164, 9165, 9166, 91

67, 9168, 9169, 9170, 9171, 9172, 9173, 9174, 9175, 9176, 9177, 917
8, 9179, 9180, 9181, 9182, 9183, 9184, 9185, 9186, 9187, 9188, 9189,
9190, 9191, 9192, 9193, 9194, 9195, 9196, 9197, 9198, 9199, 9200, 92
01, 9202, 9203, 9204, 9205, 9206, 9207, 9208, 9209, 9210, 9211, 921
2, 9213, 9214, 9215, 9216, 9217, 9218, 9219, 9220, 9221, 9222, 9223,
9224, 9225, 9226, 9227, 9228, 9229, 9230, 9231, 9232, 9233, 9234, 92
35, 9236, 9237, 9238, 9239, 9240, 9241, 9242, 9243, 9244, 9245, 924
6, 9247, 9248, 9249, 9250, 9251, 9252, 9253, 9254, 9255, 9256, 9257,
9258, 9259, 9260, 9261, 9262, 9263, 9264, 9265, 9266, 9267, 9268, 92
69, 9270, 9271, 9272, 9273, 9274, 9275, 9276, 9277, 9278, 9279, 928
0, 9281, 9282, 9283, 9284, 9285, 9286, 9287, 9288, 9289, 9290, 9291,
9292, 9293, 9294, 9295, 9296, 9297, 9298, 9299, 9300, 9301, 9302, 93
03, 9304, 9305, 9306, 9307, 9308, 9309, 9310, 9311, 9312, 9313, 931
4, 9315, 9316, 9317, 9318, 9319, 9320, 9321, 9322, 9323, 9324, 9325,
9326, 9327, 9328, 9329, 9330, 9331, 9332, 9333, 9334, 9335, 9336, 93
37, 9338, 9339, 9340, 9341, 9342, 9343, 9344, 9345, 9346, 9347, 934
8, 9349, 9350, 9351, 9352, 9353, 9354, 9355, 9356, 9357, 9358, 9359,
9360, 9361, 9362, 9363, 9364, 9365, 9366, 9367, 9368, 9369, 9370, 93
71, 9372, 9373, 9374, 9375, 9376, 9377, 9378, 9379, 9380, 9381, 938
2, 9383, 9384, 9385, 9386, 9387, 9388, 9389, 9390, 9391, 9392, 9393,
9394, 9395, 9396, 9397, 9398, 9399, 9400, 9401, 9402, 9403, 9404, 94
05, 9406, 9407, 9408, 9409, 9410, 9411, 9412, 9413, 9414, 9415, 941
6, 9417, 9418, 9419, 9420, 9421, 9422, 9423, 9424, 9425, 9426, 9427,
9428, 9429, 9430, 9431, 9432, 9433, 9434, 9435, 9436, 9437, 9438, 94
39, 9440, 9441, 9442, 9443, 9444, 9445, 9446, 9447, 9448, 9449, 945
0, 9451, 9452, 9453, 9454, 9455, 9456, 9457, 9458, 9459, 9460, 9461,
9462, 9463, 9464, 9465, 9466, 9467, 9468, 9469, 9470, 9471, 9472, 94
73, 9474, 9475, 9476, 9477, 9478, 9479, 9480, 9481, 9482, 9483, 948
4, 9485, 9486, 9487, 9488, 9489, 9490, 9491, 9492, 9493, 9494, 9495,
9496, 9497, 9498, 9499, 9500, 9501, 9502, 9503, 9504, 9505, 9506, 95
07, 9508, 9509, 9510, 9511, 9512, 9513, 9514, 9515, 9516, 9517, 951
8, 9519, 9520, 9521, 9522, 9523, 9524, 9525, 9526, 9527, 9528, 9529,
9530, 9531, 9532, 9533, 9534, 9535, 9536, 9537, 9538, 9539, 9540, 95
41, 9542, 9543, 9544, 9545, 9546, 9547, 9548, 9549, 9550, 9551, 955
2, 9553, 9554, 9555, 9556, 9557, 9558, 9559, 9560, 9561, 9562, 9563,
9564, 9565, 9566, 9567, 9568, 9569, 9570, 9571, 9572, 9573, 9574, 95
75, 9576, 9577, 9578, 9579, 9580, 9581, 9582, 9583, 9584, 9585, 958
6, 9587, 9588, 9589, 9590, 9591, 9592, 9593, 9594, 9595, 9596, 9597,
9598, 9599, 9600, 9601, 9602, 9603, 9604, 9605, 9606, 9607, 9608, 96
09, 9610, 9611, 9612, 9613, 9614, 9615, 9616, 9617, 9618, 9619, 962
0, 9621, 9622, 9623, 9624, 9625, 9626, 9627, 9628, 9629, 9630, 9631,
9632, 9633, 9634, 9635, 9636, 9637, 9638, 9639, 9640, 9641, 9642, 96
43, 9644, 9645, 9646, 9647, 9648, 9649, 9650, 9651, 9652, 9653, 965
4, 9655, 9656, 9657, 9658, 9659, 9660, 9661, 9662, 9663, 9664, 9665,
9666, 9667, 9668, 9669, 9670, 9671, 9672, 9673, 9674, 9675, 9676, 96
77, 9678, 9679, 9680, 9681, 9682, 9683, 9684, 9685, 9686, 9687, 968
8, 9689, 9690, 9691, 9692, 9693, 9694, 9695, 9696, 9697, 9698, 9699,
9700, 9701, 9702, 9703, 9704, 9705, 9706, 9707, 9708, 9709, 9710, 97
11, 9712, 9713, 9714, 9715, 9716, 9717, 9718, 9719, 9720, 9721, 972
2, 9723, 9724, 9725, 9726, 9727, 9728, 9729, 9730, 9731, 9732, 9733,
9734, 9735, 9736, 9737, 9738, 9739, 9740, 9741, 9742, 9743, 9744, 97
45, 9746, 9747, 9748, 9749, 9750, 9751, 9752, 9753, 9754, 9755, 975
6, 9757, 9758, 9759, 9760, 9761, 9762, 9763, 9764, 9765, 9766, 9767,
9768, 9769, 9770, 9771, 9772, 9773, 9774, 9775, 9776, 9777, 9778, 97
79, 9780, 9781, 9782, 9783, 9784, 9785, 9786, 9787, 9788, 9789, 979
0, 9791, 9792, 9793, 9794, 9795, 9796, 9797, 9798, 9799, 9800, 9801,
9802, 9803, 9804, 9805, 9806, 9807, 9808, 9809, 9810, 9811, 9812, 98
13, 9814, 9815, 9816, 9817, 9818, 9819, 9820, 9821, 9822, 9823, 982
4, 9825, 9826, 9827, 9828, 9829, 9830, 9831, 9832, 9833, 9834, 9835,
9836, 9837, 9838, 9839, 9840, 9841, 9842, 9843, 9844, 9845, 9846, 98
47, 9848, 9849, 9850, 9851, 9852, 9853, 9854, 9855, 9856, 9857, 985

```
8, 9859, 9860, 9861, 9862, 9863, 9864, 9865, 9866, 9867, 9868, 9869,
9870, 9871, 9872, 9873, 9874, 9875, 9876, 9877, 9878, 9879, 9880, 98
81, 9882, 9883, 9884, 9885, 9886, 9887, 9888, 9889, 9890, 9891, 989
2, 9893, 9894, 9895, 9896, 9897, 9898, 9899, 9900, 9901, 9902, 9903,
9904, 9905, 9906, 9907, 9908, 9909, 9910, 9911, 9912, 9913, 9914, 99
15, 9916, 9917, 9918, 9919, 9920, 9921, 9922, 9923, 9924, 9925, 992
6, 9927, 9928, 9929, 9930, 9931, 9932, 9933, 9934, 9935, 9936, 9937,
9938, 9939, 9940, 9941, 9942, 9943, 9944, 9945, 9946, 9947, 9948, 99
49, 9950, 9951, 9952, 9953, 9954, 9955, 9956, 9957, 9958, 9959, 996
0, 9961, 9962, 9963, 9964, 9965, 9966, 9967, 9968, 9969, 9970, 9971,
9972, 9973, 9974, 9975, 9976, 9977, 9978, 9979, 9980, 9981, 9982, 99
83, 9984, 9985, 9986, 9987, 9988, 9989, 9990, 9991, 9992, 9993, 999
4, 9995, 9996, 9997, 9998, 9999]
```

In [28]:

```
min(numbers)
```

Out[28]:

```
1
```

In [30]:

```
max(numbers)
```

Out[30]:

```
9999
```

In [32]:

```
sum(numbers)
```

Out[32]:

```
49995000
```

In [35]:

```
#4.6
numbers = []
for value in range(1, 21, 3):
    numbers.append(value)
print(numbers)
```

```
[1, 4, 7, 10, 13, 16, 19]
```

In [38]:

```
#4.7
multiples = []
for value in range(3,31):
    multiples.append(value*3)

print(multiples)
```

```
[9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 42, 45, 48, 51, 54, 57,
60, 63, 66, 69, 72, 75, 78, 81, 84, 87, 90]
```

In [40]:

```
#4.8
cubes = []
for value in range(1,11):
    cubes.append(value**3)

print(cubes)
```

```
[1, 8, 27, 64, 125, 216, 343, 512, 729, 1000]
```

In [41]:

```
#4.9
cubes = [value**3 for value in range(1,11)]
print(cubes)
```

```
[1, 8, 27, 64, 125, 216, 343, 512, 729, 1000]
```

Working with Part of a List

In [27]:

```
players = ['charles', 'martina', 'michael', 'florence', 'eli']
print(players[0:3])
```

```
['charles', 'martina', 'michael']
```

- **slicing** a list
  - you can slice a list with the index
  - just as witht the *range() function* for the indexing the first item here is 0 and the last one is 3

In [29]:

```
players = ['charles', 'martina', 'michael', 'florence', 'eli']
print(players[1:4])
```

```
['martina', 'michael', 'florence']
```

- this time it starts with the second item (the first item is always 0) and ends with the fourth item which is *florence*

In [30]:

```
players = ['charles', 'martina', 'michael', 'florence', 'eli']
print(players[:4])
```

```
['charles', 'martina', 'michael', 'florence']
```

- without a starting index, it will start from the beginning of the list

In [31]:

```
players = ['charles', 'martina', 'michael', 'florence', 'eli']
print(players[2:])
```

```
['michael', 'florence', 'eli']
```

- the same syntax works also for the end of the index
- here we start from the third item *michael* and go to the end, since no end index is given

In [34]:

```
players = ['charles', 'martina', 'michael', 'florence', 'eli']
print(players[-3:])
```

```
['michael', 'florence', 'eli']
```

- distance from the end (-3) is the starting index
- and it goes to the end, because no end index is given

In [35]:

```
players = ['charles', 'martina', 'michael', 'florence', 'eli']

print("Here is my team:")
for player in players[:3]:
    print(player.title())
```

```
Here is my team:
Charles
Martina
Michael
```

- instead of **looping through the complete list**
- we have a index from where to where we want to loop

In [39]:

```
my_foods = ['pizza', 'falafel', 'carrot cake']
friends_foods = my_foods[:]

print("My favorite foods are:")
print(my_foods)

print("\nMy firends favorite foods are:")
print(friends_foods)
```

```
My favorite foods are:
['pizza', 'falafel', 'carrot cake']

My firends favorite foods are:
['pizza', 'falafel', 'carrot cake']
```

- **copying a list**
- we make a list of elements
- the second list will be the same so we take the whole index
- **we need the [:] because we have to have two different lists**
- if we just say friends_foods = my_foods then we have new items in both lists
- try it out in the next line!!!

In [43]:

```
my_foods = ['pizza', 'falafel', 'carrot cake']
friends_foods = my_foods[:]

my_foods.append('cannoli')
friends_foods.append('ice cream')

print("My favorite foods are:")
print(my_foods)

print("\nMy firends favorite foods are:")
print(friends_foods)
```

```
My favorite foods are:
['pizza', 'falafel', 'carrot cake', 'cannoli']

My firends favorite foods are:
['pizza', 'falafel', 'carrot cake', 'ice cream']
```

- see here: we append() new items and they are different for each list

Tasks

- 4-10. Slices: Using one of the programs you wrote in this chapter, add several lines to the end of the program that do the following:
  - Print the message, The first three items in the list are: . Then use a slice to print the first three items from that program's list.
  - Print the message, Three items from the middle of the list are: . Use a slice to print three items from the middle of the list.
  - Print the message, The last three items in the list are: . Use a slice to print the last three items in the list.
- 4-11. My Pizzas, Your Pizzas: Start with your program from Exercise 4-1 (page 60). Make a copy of the list of pizzas, and call it friend_pizzas. Then, do the following:
  - Add a new pizza to the original list.
  - Add a different pizza to the list friend_pizzas.
  - Prove that you have two separate lists. Print the message, My favorite pizzas are:, and then use a for loop to print the first list. Print the message, My friend's favorite pizzas are:, and then use a for loop to print the second list. Make sure each new pizza is stored in the appropriate list.
- 4-12. More Loops: All versions of foods.py in this section have avoided using for loops when printing to save space. Choose a version of foods.py, and write two for loops to print each list of foods.

In [52]:

```
#4.10
items = ['charles', 'martina', 'michael', 'florence', 'eli', 'alex', 'nico', 'ro
man']
print("The first three items of the list are: " + str(items[0:3]) + ".")
print(len(items))
print("The items from the middle of the list are: " + str(items[3:-3]) + ".")
print("The last three items of the list are: " + str(items[-3:]) + ".")
```

```
The first three items of the list are: ['charles', 'martina', 'micha
el'].
8
The items from the middle of the list are: ['florence', 'eli'].
The last three items of the list are: ['alex', 'nico', 'roman'].
```

In [74]:

```
#4.11
my_pizza = ['sucuk', 'extra extra cheese', 'tuna']
friend_pizza = my_pizza[:]

my_pizza.append('vegan')
friend_pizza.append('chicken')

print("My favorite pizzas are: " + str(my_pizza) + ".")
print("My friends favorite pizzas are: " + str(friend_pizza) + ".\n")

for pizza in my_pizza:
    print(pizza.title() + " , is my favorite Pizza!")

print("\n")

for pizza_friend in friend_pizza:
    print(pizza_friend.title() + " , is my friends favorite Pizza!")
```

```
My favorite pizzas are: ['sucuk', 'extra extra cheese', 'tuna', 'veg
an'].
My friends favorite pizzas are: ['sucuk', 'extra extra cheese', 'tun
a', 'chicken'].

Sucuk , is my favorite Pizza!
Extra Extra Cheese , is my favorite Pizza!
Tuna , is my favorite Pizza!
Vegan , is my favorite Pizza!


Sucuk , is my friends favorite Pizza!
Extra Extra Cheese , is my friends favorite Pizza!
Tuna , is my friends favorite Pizza!
Chicken , is my friends favorite Pizza!
```

Tuples

- same as lists, you just cannot change the items
- immutable means, that you cannot change it - *tuples*

In [46]:

```
dimensions = (200, 50, 1222)
print(dimensions[0])
print(dimensions[1])
print(dimensions[2])
```

```
200
50
1222
```

- we define a tuple
  - we have parantheses instead of square brackets
  - we can use indexing for each element

In [47]:

```
dimensions = (200, 50, 1222)
dimensions[0] = 250
```

```
---------------------------------------------------------------------
-------
TypeError                                 Traceback (most recent cal
l last)
<ipython-input-47-8e51121e7181> in <module>
      1 dimensions = (200, 50, 1222)
----> 2 dimensions[0] = 250

TypeError: 'tuple' object does not support item assignment
```

- we cannot change an item of the tuple
- a tuple is immutable

In [49]:

```
dimensions = (200, 50, 1222)
for dimension in dimensions:
    print(dimension)
```

```
200
50
1222
```

- you can **loop a tuple like a list**

In [53]:

```
dimensions = (200, 50, 1222)
print("Original dimensions:")
for dimension in dimensions:
    print(dimension)

dimensions = (400, 100)
print("\nModified dimensions:")
for dimension in dimensions:
    print(dimension)

print("Our new dimensions are: " + str(dimensions) + ".")
```

```
Original dimensions:
200
50
1222

Modified dimensions:
400
100
Our new dimensions are: (400, 100).
```

- we can ***redefine*** a tuple
- we are ***overwriting*** a variable!

Tasks

- 4-13. Buffet: A buffet-style restaurant offers only five basic foods. Think of five simple foods, and store them in a tuple.
  - Use a for loop to print each food the restaurant offers.
  - Try to modify one of the items, and make sure that Python rejects the change.
  - The restaurant changes its menu, replacing two of the items with different foods. Add a block of code that rewrites the tuple, and then use a for loop to print each of the items on the revised menu.

In [92]:

```
#4.13
basic_foods = ('hambuger', 'kebab', 'pommes', 'salat', 'pizza')
for foods in basic_foods:
    print(foods)

basic_foods = ('tacos', 'döner')
print("\nModified foods:")
for foods in basic_foods:
    print(foods)

print("\nOur new foods are: " + str(basic_foods) + ".")
```

```
hambuger
kebab
pommes
salat
pizza

Modified foods:
tacos
döner

Our new foods are: ('tacos', 'döner').
```

Styling cour Code

- make your code as easy to possible to read
- Python Enhancement Proposal (PEP)
- code is read more than written
- between code that is easier to write and easier to read choose always the last one!
- don't forget the importance of intentation in python!
- each line should be not more than 80 characters
- each comment less than 72 characters

5. If Statements

A simple example

In [54]:

```
cars = ['audi', 'bmw', 'subaru', 'toyota']

for car in cars:
    if car == 'bmw':
        print(car.upper())
    else:
        print(car.title())
```

```
Audi
BMW
Subaru
Toyota
```

- we want all names of the cars printed
- however, we want all names printed in *title()* but BMW should be printed in *upper()*
- we have a if statement in our for loop which checks the condition
  - if the car in the list equal to 'bmw'
  - then print it in upper case
  - else print it in title case
- "==" means equal and is not the same as "=" which means assign something to a variable

Conditional Tests

In [55]:

```
car = 'bmw'
car == 'bmw'
```

Out[55]:

```
True
```

- this conditional test checks wether the value is the same or not
- if it's the same it gives back true

In [56]:

```
car = 'audi'
car == 'bmw'
```

Out[56]:

```
False
```

- this time it it false

In [57]:

```
car = 'audi'
car == 'Audi'
```

Out[57]:

```
False
```

- python is ***case sensitive***

In [59]:

```
car = 'Audi'
car.lower() == 'audi'
```

Out[59]:

```
True
```

- if you just want to check the value, you have to lower case it before and then check it

In [60]:

```
car = 'Audi'
car.lower() == 'audi'
car
```

Out[60]:

```
'Audi'
```

- the value of car is not being affected by the test, it is the same

In [61]:

```
requested_topping = 'mushrooms'

if requested_topping != 'anchovies':
    print("Hold the anchovies!")
```

```
Hold the anchovies!
```

- checking for inequality
- with the **!=**
- the if statement runs if it's true
- here it is true, the topping is **not !!!** mushrooms
- so it gives out the print

In [62]:

```
age = 18
age == 18
```

Out[62]:

```
True
```

- numerical comparisons

In [65]:

```
answer = 17

if answer != 42:
    print("That's not the correct answer!")
```

```
That's not the correct answer!
```

- if the conditional test passes it runs the intented code block!

In [68]:

```
answer = 17

if answer == 18:
    print("That's not the correct answer!")
```

- here the conditional test don't pass, so it does not run the intented code block!!!

In [77]:

```
age = 19
```

- first we assing a variable named age and give it a value 10

In [78]:

```
age < 21
```

Out[78]:

```
True
```

In [75]:

```
age <= 21
```

Out[75]:

```
True
```

In [71]:

```
age > 21
```

Out[71]:

```
False
```

In [72]:

```
age >= 21
```

Out[72]:

```
False
```

- we can use various mathematical comparisons for our conditional tests

In [81]:

```
age_0 = 22
age_1 = 18
age_2 = 18
age_0 >= 21 and age_1 >= 21 and age_2 >= 17
```

Out[81]:

```
False
```

- we want to check multiple conditions
- we assing three variables and chehck their condition
- note that age_2 is >= than 17 but in this case all conditions have to be true, which is not the case

In [83]:

```
age_0 = 22
age_1 = 18
age_2 = 18
age_0 >= 21 or age_1 >= 21 or age_2 >= 17
```

Out[83]:

```
True
```

- in this case it is true
- we check multiple conditions again, this time one of them (**or**) has to be true, so the whole statement is true
- age_2 is true so the whole statement is true

In [84]:

```
requested_toppings = ['mushrooms', 'onions', 'pineapple']
'mushrooms' in requested_toppings
```

Out[84]:

```
True
```

In [85]:

```
requested_toppings = ['mushrooms', 'onions', 'pineapple']
'pepperoni' in requested_toppings
```

Out[85]:

```
False
```

- you want to check whether an element is in a list
- use the **in** keyword to check that out

In [91]:

```
banned_users = ['andrew', 'carolina', 'david']
user = 'marie'

if user not in banned_users:
    print(user.title() + " is a not banned user!")
else:
    print(user.title() + " is a banned user!")
```

```
Marie is a not banned user!
```

- you want to check whether an element is not in the list
- you use the **not** and **in** keywords and make an **if-statement**
- if that condition tests is true you get the message
- if the condition test is false you get the message after the else:

In [92]:

```
game_active = True
can_edit = False
```

- boolean expressions
- are either:
  - True
  - False
- with a capital!!!

Tasks

- 5-1. Conditional Tests: Write a series of conditional tests. Print a statement describing each test and your prediction for the results of each test. Your code should look something like this:

```
car = 'subaru'
print("Is car == 'subaru'? I predict True.")
print(car == 'subaru')
print("\nIs car == 'audi'? I predict False.")
print(car == 'audi')
```

  - Look closely at your results, and make sure you understand why each line evaluates to True or False.
  - Create at least 10 tests . Have at least 5 tests evaluate to True and another 5 tests evaluate to False.
- 5-2. More Conditional Tests: You don't have to limit the number of tests you create to 10. If you want to try more comparisons, write more tests and add them to conditional_tests.py . Have at least one True and one False result for each of the following:
  - Tests for equality and inequality with strings
  - Tests using the lower() function
  - Numerical tests involving equality and inequality, greater than and less than, greater than or equal to, and less than or equal to
  - Tests using the and keyword and the or keyword
  - Test whether an item is in a list
  - Test whether an item is not in a list

In [95]:

```
#5.2
car = 'citroen'
```

In [96]:

```
print("Is car == 'opel'? I predict True")
car == 'opel'
```

```
Is car == 'opel'? I predict True
```

Out[96]:

```
False
```

In [98]:

```
print("Is car == 'toyota'? I predict True")
car == 'toyota'
```

```
Is car == 'toyota'? I predict True
```

Out[98]:

```
False
```

In [99]:

```
print("Is car == 'citroen'? I predict True")
car == 'citroen'
```

```
Is car == 'citroen'? I predict True
```

Out[99]:

```
True
```

if Statements

In [94]:

```
age = 19
if age >= 18:
    print("You are old enough to vote!")
```

```
You are old enough to vote!
```

- this is a simple if statement
- it works like this:
  - if *conditional_test*:
    - do something
- you can put any conditonal test in the first line
- and just any action in the intented line
- if the test is true, the code will do the *action* in the intented line
- if the test is false, the code will do nothing and it will also ignore the if statement
- intentation plays the same role as for for-loops

In [95]:

```
age = 19
if age >= 18:
    print("You are old enough to vote!")
    print("Have you voted?")
```

```
You are old enough to vote!
Have you voted?
```

- you can add as many lines of code after the if statement!

In [96]:

```
age = 17
if age >= 18:
    print("You are old enough to vote!")
    print("Have you voted?")
else:
    print("Sorry, you are too young!")
    print("Please wait until you get 18!")
```

```
Sorry, you are too young!
Please wait until you get 18!
```

- the **else** statement allows you to define an action when the conditional test fails!
- in this case the test fails (age = 17) means the age is not >=18 so the test will give a **False**
- in cases for having a false we have an **else statement** which will do another action
- one of the two actions will always be executed
- **we just have two conditions - either the person is over 18 or not**

In [103]:

```
age = 12

if age < 4:
    print("Your admission costs 0€!")
elif age < 18:
    print("Your admission costs 5€!")
else:
    print("Your admission costs 10€!")
```

```
Your admission costs 5€!
```

- the elif is just another if statement which has to be true
- often we have not just two conditions (like before)
- here we have three conditions
  - the person is under 4 years
  - the person is under 18
  - the person is over 18 (rest of the people)
  - for each of theese three conditions we have different actions
- the elif statement is just another if statement which has to be tested
- the code tests first the first statement
- if this first statement is true it executes the corresponding action
- if the first statement is **False** it tests the second statement
- you can add as much elif statements as required!!
- if the if statement and all the elif statements are **False** then the code runs the else statements actions

In [105]:

```
age = 12

if age < 4:
    price = 0
elif age < 18:
    price = 5
else:
    price = 10

print("Your admission costs: " + str(price) + "€!")
```

```
Your admission costs: 5€!
```

- this is much nicer because you can modify the price for example!

In [107]:

```
age = 12

if age < 4:
    price = 0
elif age < 18:
    price = 5
elif age < 65:
    price = 10
elif age > 65:
    price = 5

print("Your admission costs: " + str(price) + "€!")
```

```
Your admission costs: 5€!
```

- you can add more elif statements
- you can ommit the else statement and just use another elif statement
- the last elif statement can be seen as an else statement (> goes till infinity)

In [109]:

```
requested_toppings = ['mushrooms', 'extra cheese']

if 'mushrooms' in requested_toppings:
    print("Adding mushrooms.")

if 'pepperoni' in requested_toppings:
    print("Adding pepperoni.")

if 'extra cheese' in requested_toppings:
    print("Adding extra cheese.")

print("\nFinished making your Pizza!")
```

```
Adding mushrooms.
Adding extra cheese.

Finished making your Pizza!
```

- **testing multiple conditions**
- if-elif-else is useful just when you have to pass one True from the conditions test
- if you want to pass more than one condition to pass True than you can do multiple if statements
- each of them checks whether the test is true and if true it does its action

In [110]:

```
requested_toppings = ['mushrooms', 'extra cheese']

if 'mushrooms' in requested_toppings:
    print("Adding mushrooms.")
elif 'pepperoni' in requested_toppings:
    print("Adding pepperoni.")
elif 'extra cheese' in requested_toppings:
    print("Adding extra cheese.")

print("\nFinished making your Pizza!")
```

```
Adding mushrooms.

Finished making your Pizza!
```

- see with the elif statement this code would just run until the first statement and than stop
- because after the first true the if statement is done!
- **if you want only one block to run use elif**
- **if more than one one block has to be run, use a series of independent is statements**

Tasks

- 5-3. Alien Colors #1: Imagine an alien was just shot down in a game. Create a variable called alien_color and assign it a value of 'green', 'yellow', or 'red'.
  - Write an if statement to test whether the alien's color is green. If it is, print a message that the player just earned 5 points.
  - Write one version of this program that passes the if test and another that fails. (The version that fails will have no output.)
- 5-4. Alien Colors #2: Choose a color for an alien as you did in Exercise 5-3, and write an if-else chain.
  - If the alien's color is green, print a statement that the player just earned 5 points for shooting the alien.
  - If the alien's color isn't green, print a statement that the player just earned 10 points.
  - Write one version of this program that runs the if block and another that runs the else block.
- 5-5. Alien Colors #3: Turn your if-else chain from Exercise 5-4 into an if-elif- else chain.
  - If the alien is green, print a message that the player earned 5 points.
  - If the alien is yellow, print a message that the player earned 10 points.
  - If the alien is red, print a message that the player earned 15 points.
  - Write three versions of this program, making sure each message is printed for the appropriate color alien.
- 5-6. Stages of Life: Write an if-elif-else chain that determines a person's stage of life. Set a value for the variable age, and then:
  - If the person is less than 2 years old, print a message that the person is a baby.
  - If the person is at least 2 years old but less than 4, print a message that the person is a toddler.
  - If the person is at least 4 years old but less than 13, print a message that the person is a kid.
  - If the person is at least 13 years old but less than 20, print a message that the person is a teenager.
  - If the person is at least 20 years old but less than 65, print a message that the person is an adult.
  - If the person is age 65 or older, print a message that the person is an elder.
- 5-7. Favorite Fruit: Make a list of your favorite fruits, and then write a series of independent if statements that check for certain fruits in your list.
  - Make a list of your three favorite fruits and call it favorite_fruits.
  - Write five if statements. Each should check whether a certain kind of fruit is in your list. If the fruit is in your list, the if block should print a statement, such as You really like bananas!

In [102]:

```
#5.3
alien_color = 'green'

if alien_color == 'green':
    print("You earned 5 points!")
```

```
You earned 5 points!
```

In [103]:

```
if alien_color == 'red':
    print("You earned 5 points!")
```

In [104]:

```
#5.4
alien_color = 'red'

if alien_color == 'green':
    print("You earned 5 points!")
else:
    print("You earned 10 points!")
```

```
You earned 10 points!
```

In [105]:

```
#5.5
alien_color = 'red'

if alien_color == 'green':
    print("You earned 5 points!")
elif alien_color == 'yellow':
    print("You earned 10 points!")
elif alien_color == 'red':
    print("You earned 15 points!")
```

```
You earned 15 points!
```

In [106]:

```
alien_color = 'green'

if alien_color == 'green':
    print("You earned 5 points!")
elif alien_color == 'yellow':
    print("You earned 10 points!")
elif alien_color == 'red':
    print("You earned 15 points!")
```

```
You earned 5 points!
```

In [107]:

```
alien_color = 'yellow'

if alien_color == 'green':
    print("You earned 5 points!")
elif alien_color == 'yellow':
    print("You earned 10 points!")
elif alien_color == 'red':
    print("You earned 15 points!")
```

```
You earned 10 points!
```

In [111]:

```
#5.6
age = 30

if age <= 2:
    print("You are a baby!")
elif age <= 4:
    print("You are a toddler!")
elif age <= 13:
    print("You are a kid!")
elif age <= 20:
    print("You are a teenager!")
elif age <= 65:
    print("You are an adult!")
else:
    print("You are an elder!")
```

```
You are an adult!
```

In [113]:

```
#5.7
fruits = ['banana', 'orange', 'cherry', 'kiwi']

if 'banana' in fruits:
    print("You like banana.")

if 'orange' in fruits:
    print("You like orange.")

if 'cherry' in fruits:
    print("You like cherry.")

if 'kiwi' in fruits:
    print("You like kiwi.")
```

```
You like banana.
You like orange.
You like cherry.
You like kiwi.
```

Using if Statements with Lists

In [114]:

```
requested_toppings = ['mushrooms', 'green peppers', 'extra cheese']

for requested_topping in requested_toppings:
    print("Adding " + requested_topping + ".")

print("\nFinished making your pizza!")
```

```
Adding mushrooms.
Adding green peppers.
Adding extra cheese.

Finished making your pizza!
```

- prevoius example

In [116]:

```
requested_toppings = ['mushrooms', 'green peppers', 'extra cheese']

for requested_topping in requested_toppings:
    if requested_topping == 'green peppers':
        print("Sorry, we are out of green peppers right now.")
    else:
        print("Adding " + requested_topping + ".")

print("\nFinished making your pizza!")
```

```
Adding mushrooms.
Sorry, we are out of green peppers right now.
Adding extra cheese.

Finished making your pizza!
```

- **checking for special items**
- we are adding an extra if statement into our for loop
- because we want to make a condition test before
- this statement will first test, if the item 'green onions' is in the list
- if it's in the list we will get a **True** and this will make our print action

In [120]:

```
requested_toppings = []

if requested_toppings:
    for requested_topping in requested_toppings:
        print("Adding " + requested_topping + ".")
    print("\nFinished making your pizza!")
else:
    print("Are you sure you want a plain pizza?")
```

```
Are you sure you want a plain pizza?
```

- **checking that a list is not empty**
- we have an empty list!
- instead of a for loop we check wheter
  - we use the name of the list in the if-statement
  - the code returns **True** if the list contains at least one item
  - an empty list will give a **False**
- if the if-statement (conditional test) runs **True** we run the **for loop** and this will add the requested_topping into print
- in this case it gives an **False** so we go to the **else-statement**
- which gives us the print-method

In [124]:

```
aviable_toppings = ['mushrooms', 'olives', 'green peppers', 'pepperoni', 'pineap
ple', 'extra cheese']

requested_toppings = ['mushrooms', 'french fries', 'extra cheese']

for requested_topping in requested_toppings:
    if requested_topping in aviable_toppings:
        print("Adding " + requested_topping + ".")
    else:
        print("Sorry, we don't have " + requested_topping + ".")

print("\nFinished making your Pizza!")
```

```
Adding mushrooms.
Sorry, we don't have french fries.
Adding extra cheese.

Finished making your Pizza!
```

- **using multiple lists**
- we have 2 different lists
- the for loop checks for all items in the requested_toppings list
  - if the requested_topping is in the aviable_topping list
  - if true
    - it adds
  - else:
    - it says sorry

Tasks

- 5-8. Hello Admin: Make a list of five or more usernames, including the name 'admin' . Imagine you are writing code that will print a greeting to each user after they log in to a website. Loop through the list, and print a greeting to each user:
  - If the username is 'admin', print a special greeting, such as Hello admin, would you like to see a status report?
  - Otherwise, print a generic greeting, such as Hello Eric, thank you for log- ging in again.
- 5-9. No Users: Add an if test to hello_admin.py to make sure the list of users is not empty.
  - If the list is empty, print the message We need to find some users!
  - Remove all of the usernames from your list, and make sure the correct message is printed.
- 5-10. Checking Usernames: Do the following to create a program that simulates how websites ensure that everyone has a unique username.
  - Make a list of five or more usernames called current_users.
  - Make another list of five usernames called new_users. Make sure one or two of the new usernames are also in the current_users list.
  - Loop through the new_users list to see if each new username has already been used . If it has, print a message that the person will need to enter a new username. If a username has not been used, print a message saying that the username is available.
  - Make sure your comparison is case insensitive . If 'John' has been used, 'JOHN' should not be accepted.
- 5-11. Ordinal Numbers: Ordinal numbers indicate their position in a list, such as 1st or 2nd. Most ordinal numbers end in th, except 1, 2, and 3.
  - Store the numbers 1 through 9 in a list.
  - Loop through the list.
  - Use an if-elif-else chain inside the loop to print the proper ordinal end- ing for each number . Your output should read "1st 2nd 3rd 4th 5th 6th 7th 8th 9th", and each result should be on a separate line.

In [117]:

```
#5.8
usernames = ['roman', 'nico', 'alex', 'kevin', 'julia', 'admin']

for user in usernames:
    if user == 'admin':
        print("Welcome to your Site " + user + "!")
    else:
        print("Welcome to our Site " + user + "!")
```

```
Welcome to our Site roman!
Welcome to our Site nico!
Welcome to our Site alex!
Welcome to our Site kevin!
Welcome to our Site julia!
Welcome to your Site admin!
```

In [124]:

```
#5.9
usernames = []

if usernames:
    for user in usernames:
        print("Adding " + user + ".")
else:
        print("There is no user!")
```

```
There is no user!
```

In [127]:

```
#5.10
usernames = ['roman', 'nico', 'alex', 'kevin', 'julia', 'admin']
new_usernames = ['kevin', 'nico', 'jenny', 'simon', 'julia', 'admin']


for user in usernames:
    if user in new_usernames:
        print(user.title() + " is in the list!")
    else:
        print(user.title() + " is not in the list!")
```

```
Roman is not in the list!
Nico is in the list!
Alex is not in the list!
Kevin is in the list!
Julia is in the list!
Admin is in the list!
```

In [130]:

```
#5.11
numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9]

for number in numbers:
    if number == 1:
        print(str(number) + "st")
    elif number == 2:
        print(str(number) + "nd")
    elif number == 3:
        print(str(number) + "rd")
    else:
        print(str(number) + "th")
```

```
1st
2nd
3rd
4th
5th
6th
7th
8th
9th
```