

1. Napíš program, ktorý zo znakov hviezdica (‘\*’) vytvorí takýto trojuholník: v 1. riadku je jedna hviezdica, v 2. dve, v 3. tri, ..., v  $n$ -tom riadku je  $n$  hviezdíček. Môžeš dostať takýto výstup:

```
2.  zadaj n: 6
3.  *
4.  **
5.  ***
6.  ****
7.  *****
8.  *****
```

2. Napíš program, ktorý bude podobný predchádzajúcej úlohe: program prečíta nejaké slovo a trojuholník bude skladat’ z písmen tohto slova: v 1. riadku je prvé písmeno, v 2. druhé písmeno ale dvakrát, v 3. tretia písmeno trikrát, ..., v poslednom je posledné písmeno veľa krát podľa počtu znakov slova. Môžeš dostať takýto výstup:

```
3.  zadaj slovo: PYTHON
4.  P
5.  YY
6.  TTT
7.  HHHH
8.  OOOOO
9.  NNNNNN
```

3. Aj nasledovný program bude podobný predchádzajúcemu: Program prečíta nejaké slovo a trojuholník sa bude skladat’ z písmen tohto slova: v 1. riadku je prvé písmeno, v 2. sú prvé dve, v 3. sú prvé tri, ... v poslednom riadku je kompletne slovo. Môžeš dostať takýto výstup:

```
4.  zadaj slovo: Python
5.  P
6.  Py
7.  Pyt
8.  Pyth
9.  Pytho
10. Python
```

4. Napíš program, ktorý  $n$ -krát vypíše zadané slovo takto: v 1. riadku bez odsadenia, v 2. s 1 odsadením (4 medzery), v 3. s 2 odsadeniami (8 medzier), v 4. s 3 odsadeniami (12 medzier), pre ďalšie riadky sa to opakuje od začiatku. V programe využi zvyšok po delení, napríklad  $i\%4*4$ . Môžeš dostať takýto výstup:

```
5.  zadaj slovo: Python
6.  zadaj n: 9
7.  Python
8.      Python
9.          Python
10.              Python
11. Python
12.     Python
13.         Python
14.             Python
15. Python
```

5. Napíš program, ktorý z hviezdíček vytvorí takúto pyramídu:
  - v prvom riadku je najprv  $n-1$  medzier a potom jedna hviezdica
  - v každom ďalšom riadku je o jednu medzeru menej a o dve hviezdičky viac

Môžeš dostať takýto výstup:

```
zadaj n: 7
*
***
****
*****
*****
*****
*****
*****
```

6. Aj v tejto úlohe treba napísať program, ktorý vytvorí pyramídu z hviezdíčiek, len z hviezdíčiek bude len obvod trojuholníka, vnútro trojuholníka bude zo znakov mínus ('-'). Môžeš dostať takýto výstup:

```
7. zadaj n: 7
8.      *
9.     *_*
10.    *---*
11.   *-----*
12.  *-----*
13. *-----*
14. *****
```

7. Celé číslo môžeme rozobrať na jednotlivé cifry tak, že ho najprv prevedieme na reťazec a potom vo for-cykle každú cifru (ako znak) zvlášť prevedieme na číslo. Napíš program, ktorý prečíta celé číslo, rozoberie ho na cifry, tieto vypíše a zistí ich súčet. Takýto súčet voláme **ciferný súčet**. Po spustení dostaneš, napríklad:

```
8. zadaj číslo: 784
9. cifra 7
10. cifra 8
11. cifra 4
12. ciferný súčet je 19
13.
14. zadaj číslo: 1003
15. cifra 1
16. cifra 0
17. cifra 0
18. cifra 3
19. ciferný súčet je 4
```

8. Napíš program, ktorý vyrobí jeden dlhý znakový reťazec, zložený z úsekov hviezdíčiek oddelených medzerami: na začiatku je 1 hviezdíčka (znak '\*'), za ňou medzera, potom 2 hviezdíčky a medzera, 3 hviezdíčky a medzera ... Počet hviezdíčkových úsekov je *n*. Program by mal mať takúto schému:

```
9. n = ...
10. retazec = ''
11. for ...:
12.     ...
13. print(retazec)
```

Po spustení môžeš dostať takýto výstup:

```
zadaj n: 10
* * * * * * * * * * * * * * * * * * * *
```

9. Budeš vytvárať dlhý znakový reťazec podobne ako v predchádzajúcej úlohe. Namiesto úsekov hviezdíčiek budeš do reťazca zapisovať čísla z nejakého intervalu (v tvare '<číslo>'). Schéma programu by mala byť podobná predchádzajúcej úlohe. Po spustení môžeš dostať takýto výstup:

```
10. zadaj od: 88
11. zadaj do: 111
12. <88> <89> <90> <91> <92> <93> <94> <95> <96> <97> <98> <99> <100>
13. <101> <102> <103> <104> <105> <106> <107> <108> <109> <110> <111>
```

10. Napiš program, ktorý vypíše **naformátovanú** tabuľku mocnín celých čísel z nejakého daného intervalu. Prvý stĺpec tabuľky obsahuje číslo, v druhom je druhá mocnina tohto čísla, v treťom tretia, vo štvrtom štvrtá. Môžeš dostať takýto výstup:

```
11. zadaj od: 95
12. zadaj do: 103
13. 95 9025 857375 81450625
14. 96 9216 884736 84934656
15. 97 9409 912673 88529281
16. 98 9604 941192 92236816
17. 99 9801 970299 96059601
18. 100 10000 1000000 100000000
19. 101 10201 1030301 104060401
20. 102 10404 1061208 108243216
21. 103 10609 1092727 112550881
```

11. Výpočet **pi** podľa Liebnizovho vzorca je takýto súčet radu:

```
12. 4 - 4/3 + 4/5 - 4/7 + 4/9 - 4/11 + 4/13 ...
```

Napiš program, ktorý vypočíta súčet tohto radu pre prvých  $n$  členov. Po spustení môžeš dostať:

```
zadaj počet: 10
pi = 3.0418396189294032

zadaj počet: 100000
pi = 3.1415826535897198
```

12. Asi poznáš pesničku 'Sedí mucha na stene, sedí a spí.'. Napiš program, ktorý si najprv vypýta zoznam nejakých samohlások a potom pre každú z nich vypíše túto vetu tak, že v nej všetky samohlásky nahradí touto konkrétnou. Zrejme využiješ for-cyklus a formátovací reťazec `f'S{i}d{i} m{i}ch{i} ...'`. Môžeš dostať takýto výstup:

```
13. zadaj samohlásky: eaöiuý
14. Sede meche ne stene, sede e spe.
15. Sada macha na stana, sada a spa.
16. Södö möchö nõ stönö, södö ö spö.
17. Sidi michi ni stini, sidi i spi.
18. Sudu muchu nu stunu, sudu u spu.
19. Sýdý mýchý ný stýný, sýdý ý spý.
```

13. Napiš program, ktorý vytvorí tabuľku násobenia, podobnú malej násobilke. Násobiť sa budú čísla z nejakého daného intervalu: v prvom riadku (aj stĺpci) sú násobky prvého čísla, v druhom druhého, atď. Môžeš dostať takýto výstup:

```
14. zadaj od: 8
15. zadaj do: 13
16. 64 72 80 88 96 104
17. 72 81 90 99 108 117
```

```
18. 80 90 100 110 120 130
19. 88 99 110 121 132 143
20. 96 108 120 132 144 156
21. 104 117 130 143 156 169
```

Do výpisu tabuľky pridaj prvý stĺpec aj riadok navyše s hodnotami z daného intervalu, napríklad v takomto tvare:

```
zadaj od: 8
zadaj do: 13
zadaj | 8 9 10 11 12 13
=====
8 | 64 72 80 88 96 104
9 | 72 81 90 99 108 117
10 | 80 90 100 110 120 130
11 | 88 99 110 121 132 143
12 | 96 108 120 132 144 156
13 | 104 117 130 143 156 169
```

14. Napíš program, ktorý vytvorí takúto tabuľku: pre všetky uhly (v stupňoch) z nejakého daného intervalu a kroku vypíše druhé mocniny príslušných sínusov a kosínusov a aj ich súčet. Druhé mocniny vypíše na šírku 6 a 4 desatinné miesta, súčet bez udania šírky a počtu desatinných miest. Môžeš dostať takýto výstup:

```
15. zadaj od: 0
16. zadaj do: 90
17. zadaj krok: 10
18. 0 sin**2=0.0000 cos**2=1.0000 súčet=1.0
19. 10 sin**2=0.0302 cos**2=0.9698 súčet=0.9999999999999999
20. 20 sin**2=0.1170 cos**2=0.8830 súčet=1.0
21. 30 sin**2=0.2500 cos**2=0.7500 súčet=1.0
22. 40 sin**2=0.4132 cos**2=0.5868 súčet=0.9999999999999999
23. 50 sin**2=0.5868 cos**2=0.4132 súčet=1.0
24. 60 sin**2=0.7500 cos**2=0.2500 súčet=1.0
25. 70 sin**2=0.8830 cos**2=0.1170 súčet=0.9999999999999999
26. 80 sin**2=0.9698 cos**2=0.0302 súčet=0.9999999999999999
27. 90 sin**2=1.0000 cos**2=0.0000 súčet=1.0
```

15. Napíš program, ktorý vygeneruje na číselnej osi dva náhodné body (v intervale <0, 100>) a vypočíta ich vzdialenosť. Toto urobí n-krát. Môžeš dostať takýto výstup:

```
16. zadaj n: 6
17. Prvý bod na priamke je 32, druhý bod 10. Ich vzdialenosť je 22
18. Prvý bod na priamke je 61, druhý bod 12. Ich vzdialenosť je 49
19. Prvý bod na priamke je 62, druhý bod 35. Ich vzdialenosť je 27
20. Prvý bod na priamke je 9, druhý bod 78. Ich vzdialenosť je 69
21. Prvý bod na priamke je 5, druhý bod 82. Ich vzdialenosť je 77
22. Prvý bod na priamke je 16, druhý bod 20. Ich vzdialenosť je 4
```

16. Vo vlaku sa vezie 100 cestujúcich. V každej stanici, v ktorej zastane, niekoľko ľudí vystúpi a niekoľko nastúpi. Napíš program, ktorý odsimuluje n takýchto staníc s vystupovaním a nastupovaním cestujúcich. Predpokladáme, že v každej stanici vystúpi aj nastúpi náhodný počet cestujúcich z intervalu <0, 9>. Môžeš dostať takýto výstup:

```
17. zadaj n: 8
18. Vo vlaku bolo 100 ľudí, 0 nastúpilo, 7 vystúpilo. Zostalo 93.
19. Vo vlaku bolo 93 ľudí, 4 nastúpilo, 0 vystúpilo. Zostalo 97.
20. Vo vlaku bolo 97 ľudí, 9 nastúpilo, 5 vystúpilo. Zostalo 101.
21. Vo vlaku bolo 101 ľudí, 3 nastúpilo, 9 vystúpilo. Zostalo 95.
22. Vo vlaku bolo 95 ľudí, 6 nastúpilo, 8 vystúpilo. Zostalo 93.
23. Vo vlaku bolo 93 ľudí, 3 nastúpilo, 4 vystúpilo. Zostalo 92.
24. Vo vlaku bolo 92 ľudí, 8 nastúpilo, 6 vystúpilo. Zostalo 94.
25. Vo vlaku bolo 94 ľudí, 8 nastúpilo, 7 vystúpilo. Zostalo 95.
```

17. Dostali sme správu od mimozemšťanov, ktorá je zložená zo znakov '0' a '-'. Správa obsahuje istý počet riadkov a stĺpcov takýchto znakov. Napíš program, ktorým náhodne vygeneruješ podobnú správu. Môžeš dostať takýto výstup:

```
18. zadaj počet riadkov: 5
19. zadaj počet stĺpcov: 28
20. 0-000---00-000---0--0000-0
21. 000-0000----00----0-00000-0-
22. 0-00-00-000--0-000-0-0---000
23. ---00--00-0-0--00---0000--0
24. -0-----0--0000-00-000-00---0
```

18. Budeme simulovať hádzanie dvomi hracími kockami. Zakaždým vypíšeme aj ich súčet. Napíš program, ktorý to simuluje  $n$ -krát. Môžeš dostať takýto výstup:

```
19. zadaj n: 4
20. na 1. kocke padla 1
21. na 2. kocke padla 4
22. ich súčet je 5
23. =====
24. na 1. kocke padla 4
25. na 2. kocke padla 5
26. ich súčet je 9
27. =====
28. na 1. kocke padla 6
29. na 2. kocke padla 1
30. ich súčet je 7
31. =====
32. na 1. kocke padla 4
33. na 2. kocke padla 1
34. ich súčet je 5
35. =====
```

19. Podobný príklad predchádzajúcemu, lenže teraz budeme hádzať ľubovoľným počtom kociek: Napíš program, ktorý si najprv vypýta  $n$  (počet hádzaní) a počet kociek a potom  $n$ , krát vypíše čísla na kockách a ich súčet. Môžeš dostať takýto výstup:

```
20. zadaj n: 3
21. zadaj počet kociek: 4
22. na 1. kocke padla 3
23. na 2. kocke padla 2
24. na 3. kocke padla 2
25. na 4. kocke padla 2
26. ich súčet je 9
27. =====
28. na 1. kocke padla 4
29. na 2. kocke padla 6
30. na 3. kocke padla 1
31. na 4. kocke padla 5
32. ich súčet je 16
33. =====
34. na 1. kocke padla 1
35. na 2. kocke padla 4
36. na 3. kocke padla 6
37. na 4. kocke padla 3
38. ich súčet je 14
39. =====
```

20. Pomocou tohto programu vieme zaplniť štvorcovú tabuľku  $n \times n$ :

```
21. n = int(input('zadaj n: '))
22. for i in range(n):
23.     for j in range(n):
24.         print(f'{i*n + j + 1:2}', end=' ')
25.     print()
```

Oprav tento program tak, aby vypísal túto tabuľku trikrát vedľa seba, napríklad takto:

```
zadaj n: 5
1 2 3 4 5      1 2 3 4 5      1 2 3 4 5
6 7 8 9 10     6 7 8 9 10     6 7 8 9 10
11 12 13 14 15 11 12 13 14 15 11 12 13 14 15
16 17 18 19 20 16 17 18 19 20 16 17 18 19 20
21 22 23 24 25 21 22 23 24 25 21 22 23 24 25
```