Wstęp do programowania 2017 Pracownia 2

Dla każdego z poniższych zadań powinieneś napisać program, który umieścisz w osobnym pliku. Programy te nie będą pobierały żadnych danych, będą natomiast wypisywać na standardowym wyjściu pewne teksty. Na tych zajęciach **jeszcze nie ma wprawek**. Dla tej listy obowiązują dwie premie: 0.5p za zrobienie co najmniej 2.5 punktu, plus dodatkowo 0.5 p za zrobienie wszystkich czterech zadań w pełnej wersji. Do maksimum za tę listę wlicza się 4.5p.

Zadanie 1.(1pkt) Napisz procedurę szachowica(n,k), która rysuje kwadratową szachownicę, o boku 2*n, składającą się z kwadratów o boku k, jak na poniższym rysunku (na którym pokazany jest wynik wywołania szachownica(4,3)).

```
###
           ###
                  ###
   ###
           ###
                  ###
                         ###
   ###
           ###
                  ###
                         ###
###
              ###
                      ###
###
       ###
               ###
                  ###
                  ###
                  ###
###
       ###
              ###
###
               ###
                      ###
               ###
       ###
           ###
                  ###
           ###
                  ###
                         ###
           ###
                  ###
                         ###
       ###
              ###
               ###
              ###
   ###
           ###
                  ###
   ###
           ###
                  ###
                         ###
           ###
                  ###
                         ###
                      ###
       ###
              ###
###
       ###
              ###
                      ###
###
       ###
              ###
                      ###
```

Zadanie 2.(1pkt) Jesteś w jaskini, której wyjścia pilnuje smok. Smok wyjmuje sześcienną kostkę do gry i proponuje następującą zabawę: rzucasz sto razy tą kostką, jeżeli w tych stu rzutach znajdzie się ciąg, który zadowoli smoka, możesz wyjść. W przeciwnym przypadku musisz rzucać jeszcze raz. Oczywiście pytasz, jakie ciągi smokowi się podobają, na co ów odpowiada: pięcioelementowe rosnące oraz sześcioelementowe niemalejące – wybierz sam, ale przed grą.

Napisz program, który symuluje tę grę, czyli rzuca 100 razy "pseudokością" i zwraca w wyniku informację, czy wygraliśmy i możemy opuścić jaskinię (dla obu wariantów). Wykorzystaj ten program do oszacowania prawdopodobieństwa zadowolenia smoka w obu wariantach gry (i tym samym do wybrania, na co lepiej się zdecydować).

Z zadaniem związane są następujące premie (niewliczające się do maksimum):

- 1. Tak naprawdę obowiązkowy jest tylko jeden wariant, zrobienie obydwu to premia +0.2
- 2. Jeżeli Twoje rozwiązanie będzie efektywne to Prowadzący daje premię do ± 0.2
- 3. Jeżeli w sposób elegancki rozwiążesz kwestię dwóch wariantów gry (co oznacza łatwość dodania dowolnego innego wariantu, na przykład: "dobre ciągi to 10 elementowe, w których żadne dwa kolejne się nie powtarzają"., to dostaniesz premię +0.2.

Mogą ci się przydać dwie informacje dotyczące Pythona:

• Następująca pętla wykonuje potencjalnie nieskończone obliczenia, a w momencie, gdy w instrukcji if spełniony będzie warunek, kończy swoje działanie:

```
while True:
...
if WarunekKonca:
break
```

• Losowanie można wykonywać za pomocą funkcji randint z modułu random. Czyli program:

```
from random import randint
print (randint(0,9))
print (randint(0,9))
print (randint(0,9))
print (randint(0,9))
```

wypisze cztery cyfry, wśród których może znaleźć się zarówno 0 jak i 9

Zadanie 3.(1pkt) Napisz procedure kółko (n), która wypisuje na wyjściu kółko zrobione ze znaków #, mieszczące się w kwadracie $n \times n$. Wykorzystaj ją do stworzenia programu, który drukuje bałwanka, na przykład takiego:

```
###
     #####
    #######
    #######
    ######
     #####
      ###
     #####
    #######
   ########
   ########
   ########
   #########
   ########
    #######
     #####
     #####
  ########
  ###########
 ############
 ############
###############
###############
###############
###############
###############
#############
 ############
  ###########
   ########
     #####
```

Zastanów się, jak zmienić interfejs procedury kółko by umożliwiała wygodne tworzenie takich rysunków, jak powyższy bałwanek. Jeżeli uznasz to za wygodne, możesz założyć, że n jest nieparzyste. Zwróć uwagę, że kółka są okrągłe, czyli że ich obwody nie są ośmiokątami.

Zadanie 4.(1pkt) Jak wiadomo, w pythonie są duże liczby całkowite. W tym zadaniu należy napisać program, który wypisuje Duże Liczby Całkowite (tm), na przykład taką:

```
# ### #### # ####
```

```
## # # # # #
# ## ### #### ####
# # # # # #
### #### #### ####
```

W programie powinna być funkcja, która bierze jako argument liczbę całkowitą i wypisuje DLC na konsoli. Czcionki zdefiniowane są w module duze_cyfry.py, można z nich korzystać w następujący sposób:

```
from duze_cyfry import dajCyfre
dajCyfre(4)
for r in dajCyfre(6): print (r)
```

Funkcja dajCyfre zwraca zawsze pięcioelementową listę napisów o długości 5.

Uwaga: zadanie ma łatwiejszą wersję, wartą 0.5p, w której cyfry wypisywane są jedna pod drugą.