

W ramach wstępu proszę o przypomnienie niektórym Waszym kolegom/koleżankom o oddawanie/przesyłanie prac. Wysyłacie je ZAWSZE na tego samego maila- dawidkoch.szkoła@gmail.com W razie problemów czy innych spraw, które uniemożliwiają Wam oddanie pracy, proszę o kontakt przez tego maila. Gdy prace nie będą oddawane w terminie (i tak staram się zadawać stosunkowo mało, by Was nie przeładować obowiązkami) o sytuacji będą informowani Wasi wychowawcy.

04.06.2020r.

Definiowanie funkcji obliczeniowych

Z tego tematu nie ma pracy domowej

Linki do tematu:

<https://epodreczniki.pl/pdf/PHJusLbHU>

<https://pliki.migra.pl/f.php?d=s&id=1634>

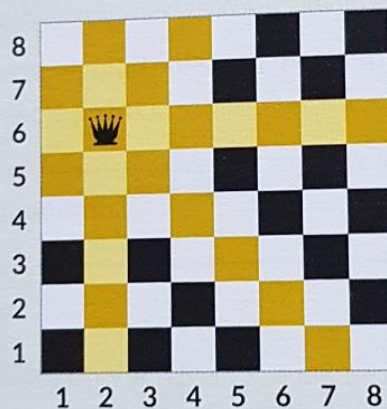
15. Definiowanie funkcji obliczeniowych

- Wykorzystanie instrukcji warunkowej w obliczeniach
- Instrukcja iteracji
- Analizowanie i testowanie rozwiązań

ZADANIE NA START

Ile pól atakuje hetman

Hetman (królowa) porusza się w tym samym rzędzie lub w tej samej kolumnie albo po przekątnych, w dowolnym kierunku, o dowolną liczbę niezajętych pól. Pola, do których figura może dotrzeć w jednym ruchu, są przez nią atakowane. Hetman stojący na szachownicy 8×8 na polu (2, 6), gdzie 2 oznacza numer kolumny, a 6 – numer wiersza, atakuje $14 + 9 = 23$ pól. Określ liczbę atakowanych pól na szachownicy 32×32 , gdy hetman znajduje się na polu (25, 30).



SZACHY, SZTUCZNA INTELIGENCJA I PROGRAMOWANIE

Najstarsza maszyna, która potrafiła w ograniczonym stopniu grać w szachy, powstała już pod koniec XIX w. Obecnie programiści tworzą coraz skuteczniejsze algorytmy, pozwalające maszynie zmieścić się w regulaminowym czasie rozgrywania partii szachowych, rozpatrujące pozycje, które wpłyną lub nie na zmianę sytuacji strategiczno-taktycznej, wybierające kilka najbardziej obiecujących ruchów i znajdujące prawdopodobnie najlepszy ruch. W zależności od złożoności problemu szachowe mogą wymagać zastosowania prostej funkcji z parametrem i operatorów arytmetycznych albo bardziej zaawansowanych narzędzi.

POSZUKAJ W INTERNECIE

Poszukaj w internecie informacji na temat pojedynku Garriego Kasparowa z systemem Deep Blue oraz innych pojedynków komputer vs. człowiek.

W tabeli przedstawiono podstawowe konstrukcje stosowane w języku Python.

Konstrukcja	Składnia	Przykład
Instrukcja przypisania	<code>nazwa_zmiennej = wartość</code>	<code>x = 3</code>
Instrukcja warunkowa prosta	<code>if warunek: instrukcje</code>	<code>if x % 3 == 0: print("liczba podzielna przez 3")</code>
Instrukcja warunkowa złożona	<code>if warunek: instrukcje elif warunek: instrukcje else: instrukcje</code>	<code>if x > 0: print("liczba dodatnia") elif x == 0: print("zero") else: print("liczba ujemna")</code>
Pętla for	<code>for nazwa_zmiennej in sekwencja: instrukcje</code>	<code>for i in range(10): print(i)</code>
Pętla while	<code>while warunek: instrukcje</code>	<code>while x > 10: print(x) x = x - 1</code>
Funkcja	<code>def nazwa(parametry): instrukcje</code>	<code>def pole_prostokata(bok1, bok2): return bok1 * bok2</code>

Ćwiczenie 1. Ile pól ma szachownica

Szachownica złożona jest z równych pól, naprzemiennie białych i czarnych. Przeanalizuj dane w tabeli i zdefiniuj funkcję `ile_pol(n)`, której parametrem jest rozmiar szachownicy `n`, a wynikiem liczba pól na szachownicy. Parametr `n` może przyjmować wartości od 2 do 32.

Wywołanie funkcji	Wynik
<code>ile_pol(8)</code>	64
<code>ile_pol(13)</code>	169

- Plansza złożona jest z $n \times n$ pól, a więc $8 \cdot 8 = 64$, $13 \cdot 13 = 169$.
- Zdefiniuj funkcję `ile_pol(n)`.
 1. `def ile_pol(n):`
 2. `return n * n`
- Sprawdź działanie skryptu z różnymi danymi.

Pamiętaj, aby następne ćwiczenia wykonywać w tym samym pliku.

- Instrukcja warunkowa na rys. 2 składa się z trzech części. Można ją odczytać następująco:
- przypisz zmiennej **x** wartość -12 ;
 - jeśli zmienna **x** (liczba -12) jest większa od 0 , przypisz zmiennej **zn** wartość 1 ;
 - jeśli poprzedni warunek jest fałszywy i **x** (liczba -12) jest mniejsze od 0 , przypisz zmiennej **zn** wartość -1 ;
 - w przeciwnym wypadku przypisz zmiennej **zn** wartość 0 .

```
1. x = -12
2. if x > 0:
3.     zn = 1
4. elif x < 0:
5.     zn = -1
6. else:
7.     zn = 0
8. print(zn)
```

Rys. 2. Przykładowa instrukcja warunkowa złożona z trzech elementów