

# Statystyka

## 1 Zadanie

### 1.1 Średnia i wariancja z tablicy

Szablon programu należy uzupełnić o definicję funkcji `mean_variance()`, która na podstawie wpisanej jednowymiarowej tablicy liczb całkowitych zwraca wartość średniej arytmetycznej tych liczb oraz wariancji.

Definicje: <https://en.wikipedia.org/wiki/Variance>

- **Wejście**  
1 len array[len]
- **Wyjście**  
Wartości średniej arytmetycznej i wariancji
- **Przykład:**  
Wejście: 1 10 4 5 1 5 5 3 3 4 2 1  
Wyjście: 3.30 2.21

### 1.2 Wypełnianie tablicy wartościami prób Bernoulliego

Szablon programu należy uzupełnić o definicję funkcji `bernoulli_gen()`, która generuje losowo tablicę  $N$  prób Bernoulliego. Elementy tej tablicy to wartości 0 i 1. Jako wejścia wprowadzana jest liczba elementów tablicy oraz prawdopodobieństwo (w procentach) wystąpienia wartości 1.

Definicje: [https://en.wikipedia.org/wiki/Bernoulli\\_sampling](https://en.wikipedia.org/wiki/Bernoulli_sampling)

- **Wejście**  
2 seed n
- **Wyjście**  
Wylosowana tablica liczb binarnych.
- **Przykład:**  
Wejście: 2 2 20 30  
Wyjście: 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0

### 1.3 Dyskretny rozkład prawdopodobieństwa

Szablon programu należy uzupełnić o definicję funkcji `pmf()` (probability mass function), która tworzy dyskretny rozkład prawdopodobieństwa dla liczb losowych uzyskanych za pomocą dwóch kostek sześciennych w  $m$  próbach. Wynikiem jest prawdopodobieństwo wyrzucenia każdej z 11 liczb [2-12] wyrażone w procentach.

Definicje: <https://en.wikipedia.org/wiki/Probability-mass-function>

- **Wejście**  
3 seed trials

- **Wyjście**

Dyskretny rozkład prawdopodobieństwa wyrażający procentową reprezentację prawdopodobieństwa wyrzucenia każdej z liczb od 2 do 12.

- **Przykład:**

Wejście: 3 2 1000

Wyjście: 2.90 5.60 7.20 13.20 14.30 17.20 12.30 11.20 8.50 4.20 3.40

## 1.4 Dystrybuanta (Cumulative Distribution Function)

Szablon programu należy uzupełnić o definicję funkcji `calculate_cdf()`, która podaje wartości dystrybuanty dla całego przedziału zmienności zmiennej losowej w przypadku rzutu dwiema kostkami sześciennymi. Zmienna losowa przyjmuje 11 wartości całkowitych [2-12].

Definicje: [https://en.wikipedia.org/wiki/Cumulative\\_distribution\\_function](https://en.wikipedia.org/wiki/Cumulative_distribution_function)

- **Wejście**

3 seed len

- **Wyjście**

Tablica wartości dystrybuanty wyrażona w procentach dla możliwych sum wyrzuconych oczek na 2 kostkach.

- **Przykład:**

Wejście: 4 2 100

Wyjście: 1.00 6.00 12.00 26.00 44.00 61.00 70.00 83.00 94.00 97.00 100.00

## 1.5 Monty Hall problem, czyli jak wybierać “bramki” żeby wygrać

Paradoks Monty’ego Halla w przypadku trzech bramek do wyboru polega na tym, że intuicyjnie przypisujemy równe szanse dwóm sytuacjom – wskazanie wygranej w jednej z dwóch zakrytych ciągle bramek wydaje się równie prawdopodobne jak posiadanie bramki pustej, bo przecież “nic nie wiadomo”. Tymczasem układ jest warunkowany przez początkowy wybór zawodnika i obie sytuacje nie pojawiają się równie często.

Opis problemu: [https://en.wikipedia.org/wiki/Monty\\_Hall\\_problem](https://en.wikipedia.org/wiki/Monty_Hall_problem)

Szablon programu należy uzupełnić o definicję funkcji `monty_hall()`, która podaje liczbę wygranych podejść oraz prawdopodobieństwo wygranej jeżeli zawodnik po odsłonięciu jednej z trzech bramek zmieni swój wybór na przeciwny.

- **Wejście**

5 seed len

- **Wyjście**

Liczba wygranych przez zmianę wyboru po otwarciu 1 z 3 bramek oraz prawdopodobieństwo wygranej w procentach w takiej sytuacji oraz liczba wygranych gdy pozostajemy przy pierwotnym wyborze i prawdopodobieństwo wygranej w procentach w takiej sytuacji.

- **Przykład:**

Wejście: 5 2 1000

Wyjście: 673 67.30 327 32.70