

Zadanie 1. n -permutacja

Dla dodatniej liczby całkowitej n , **n -permutacją** nazywamy taki n -elementowy ciąg liczb całkowitych, który zawiera każdą z liczb $1, 2, \dots, n$ dokładnie jeden raz.

Przykład:

ciąg $(4, 2, 1, 3)$ jest **4-permutacją**,

ciąg $(6, 5, 4, 1, 2, 3)$ jest **6-permutacją**,

ciągi $(1, 3, 1, 2)$ i $(2, 3, 4, 5)$ nie są **4-permutacjami**.

W ciągu n liczb całkowitych, który nie jest **n -permutacją**, można podmienić niektóre elementy tak, aby otrzymać **n -permutację**.

Przykład:

w ciągu $(1, 3, 1)$ wystarczy podmienić jeden element – pierwszą lub ostatnią jedynkę (1) – na dwójkę (2), aby powstały ciąg był **3-permutacją**.

Zadanie 1.1. (0–2)

Uzupełnij poniższą tabelę – dla każdego z podanych ciągów podaj najmniejszą liczbę elementów, które trzeba podmienić, aby dany ciąg był **n -permutacją**. Jeśli ciąg jest już **n -permutacją**, wpisz 0.

n	ciąg	liczba elementów do podmiany
3	$(1, 3, 1)$	1
4	$(1, 4, 2, 5)$	
5	$(2, 2, 2, 2, 2)$	
4	$(4, 2, 3, 1)$	
6	$(5, 4, 1, 5, 6, 8)$	
6	$(8, 4, 9, 6, 5, 7)$	

Zadanie 1.2. (0–4)

Zapisz w pseudojęzyku lub wybranym języku programowania algorytm, który dla danego ciągu n dodatnich liczb całkowitych zapisanego w tablicy A obliczy najmniejszą liczbę elementów, które trzeba w nim podmienić, aby otrzymać **n -permutację**.

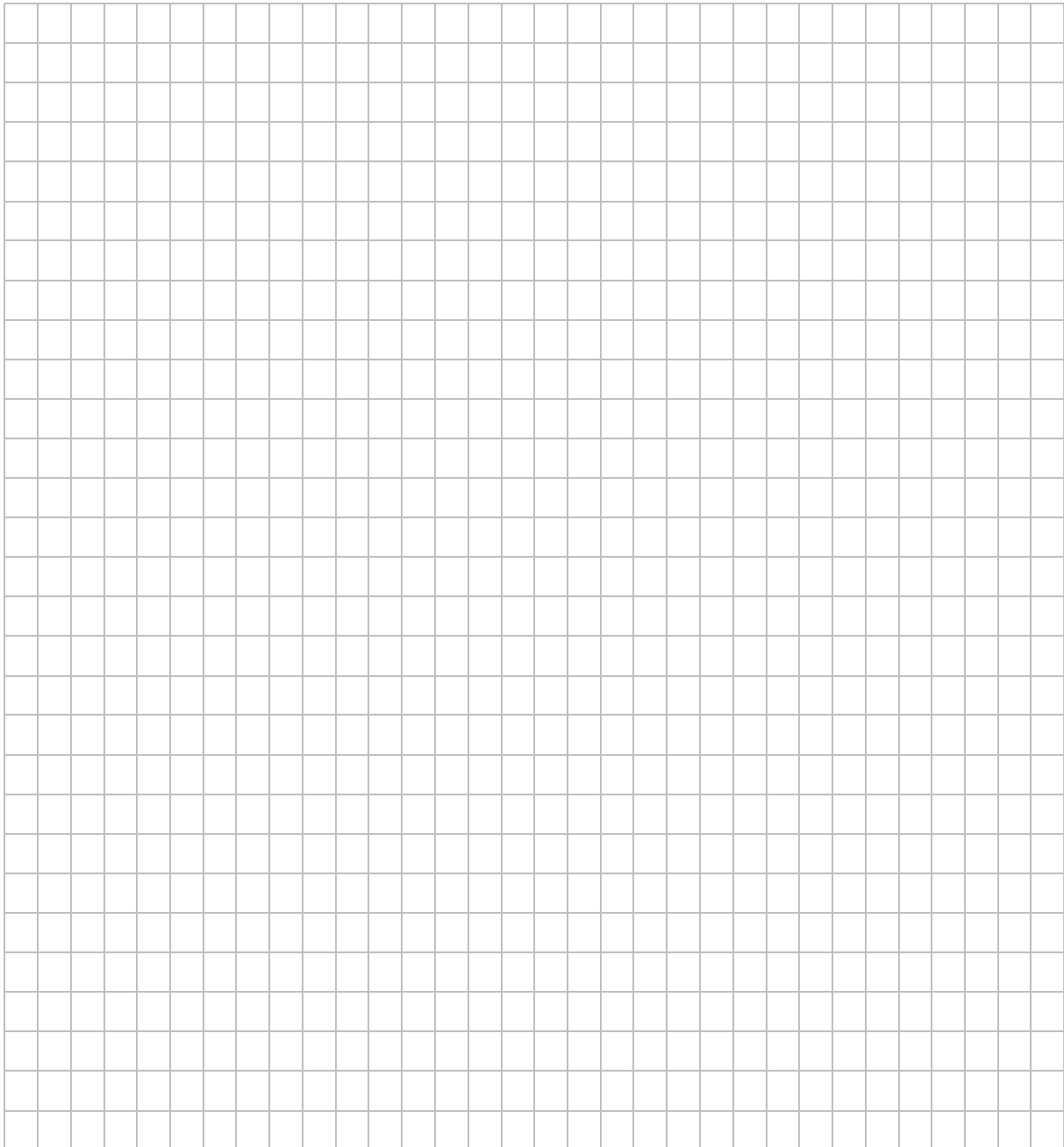
Uwaga: W zapisie algorytmu możesz korzystać tylko z instrukcji sterujących, operatorów arytmetycznych: dodawania, odejmowania, mnożenia, dzielenia, dzielenia całkowitego i reszty z dzielenia; operatorów logicznych, porównań, odwoływania się do pojedynczych elementów tablicy i instrukcji przypisania lub samodzielnie napisanych funkcji i procedur wykorzystujących powyższe operacje. **Zabronione** jest używanie funkcji wbudowanych oraz operatorów innych niż wymienione, dostępnych w językach programowania.

Specyfikacja:

Dane:

 n – dodatnia liczba całkowita $A[1..n]$ – tablica n dodatnich liczb całkowitych, gdzie $A[i]$ jest i -tym elementem ciągu

Wynik:

 k – minimalna liczba elementów, które trzeba podmienić w ciągu zapisanym w tablicy A , aby otrzymać n -permutację**Algorytm:**

Wypełnia egzaminator	Nr zadania	1.1.	1.2.
	Maks. liczba pkt.	2	4
	Uzyskana liczba pkt.		