

PROBLEMA 1

Enunt

Se da un sir de N numere pozitive. Tuzgu si Ritza au la dispozitie acest sir si vor sa joace urmatorul joc: jucatorul la mutare are dreptul sa aleaga oricare dintre numerele ramase in sir si sa il elimine. Definim scorul unui jucator ca fiind suma numerelor eliminate de respectivul jucator pe parcursul jocului. Scopul fiecarui jucator este sa maximizeze diferenta dintre scorul sau si scorul adversarului.

Stiind ca Tuzgu si Ritza muta alternativ si ambii jucatori joaca optim, se cere determinarea diferentei maxime pe care o poate obtine Tuzgu in final.

Date de intrare

Pe prima linie a fisierului **p1.in** se afla un numar intreg N .

Pe urmatoarea linie se afla N numere intregi $v_1, v_2, v_3, \dots, v_N$ reprezentand cele N numere pe care le folosesc cei doi pe parcursul jocului.

Date de iesire

In fisierul **p1.out** se va scrie diferenta maxima dintre scorul lui Tuzgu si scorul Ritzei, daca cei doi jucatori joaca optim.

Restrictii si precizari

- $1 \leq N \leq 10^5$
- $0 \leq v_1, v_2, v_3, \dots, v_N \leq 10^9$
- Tuzgu incepe jocul
- Jocul se termina atunci cand au fost eliminate toate cele N elemente din sir
- Pentru teste in valoare de 15 puncte, $1 \leq N \leq 10^3$

Testare si punctare

- Punctajul maxim este de 30 puncte.
- Timpul de executie:
 - C/C++: 0.5 s
 - Java, Python: 1.0 s
- Sursa care contine functia **main** trebuie obligatoriu denumita: **p1.c**, **p1.cpp** sau **P1.java**.

Exemple

Exemplu 1

Exemplu 1		
p1.in	p1.out	Explicatie
3 1 5 3	3	<p>Tuzgu are la dispozitie sirul [1, 5, 3] si alege numarul 5, pe care il elimina.</p> <p>Ritza are la dispozitie sirul [1, 3] si alege numarul 3.</p> <p>Tuzgu are la dispozitie sirul [1] si alege numarul 1, iar jocul se termina.</p> <p>Tuzgu a obtinut scor 6, iar Ritza scor 3, diferenta maxima posibila pentru Tuzgu fiind egala cu 3.</p>

PROBLEMA 3

Enunt

Tuzgu si Ritza vor sa joace un al 3-lea joc. Acest joc seamana foarte mult cu jocul de la prima cerinta, singura diferenta fiind ca jucatorul la mutare are voie sa elimine ori elementul cel mai din stanga al sirului, ori elementul cel mai din dreapta.

Care este diferenta maxima pe care o poate obtine Tuzgu in acest caz?

Date de intrare

Pe prima linie a fisierului **p3.in** se afla un numar intreg **N**.

Pe urmatoarea linie se afla **N** numere intregi $v_1, v_2, v_3, \dots, v_N$ reprezentand cele **N** numere pe care le folosesc cei doi pe parcursul jocului.

Date de iesire

In fisierul **p3.out** se va scrie diferenta maxima dintre scorul lui Tuzgu si scorul Ritzei, daca cei doi jucatori joaca optim.

Restrictii si precizari

- $1 \leq N \leq 10^3$
- $0 \leq v_1, v_2, v_3, \dots, v_N \leq 10^9$
- Tuzgu incepe jocul
- Jocul se termina atunci cand au fost eliminate toate cele **N** elemente din sir

Testare si punctare

- Punctajul maxim este de **35** puncte.
- Timpul de executie:
 - C/C++: **0.5 s**
 - Java, Python: **1.0 s**
- Sursa care contine functia **main** trebuie obligatoriu denumita: **p3.c**, **p3.cpp** sau **P3.java**.

Exemple

Exemplu 1

Exemplu 1		
p3.in	p3.out	Explicatie
3 1 5 3	-1	<p>Tuzgu incepe jocul, avand la dispozitie sirul [1,5,3].</p> <p>La prima mutare, va elimina numarul 3.</p> <p>Ritza are la dispozitie sirul [1,5] si va elimina numarul 5.</p> <p>Tuzgu ramane doar cu o singura optiune, numarul 1, pe care il va si elimina.</p> <p>Scorul lui Tuzgu este 4, iar cel al Ritzei este 5, deci in cel mai bun caz, Tuzgu poate obtine o diferenta de -1.</p>

Exemplu 2

Exemplu 2		
p3.in	p3.out	Explicatie
4 1 5 3 1	2	<p>Ordinea in care se fac mutarile, stiind ca cei doi jucatori joaca optim, este urmatoarea:</p> <ol style="list-style-type: none"> 1. Tuzgu elimina numarul 1 din dreapta 2. Ritza elimina numarul 3 din dreapta 3. Tuzgu elimina numarul 5 din dreapta 4. Ritza elimina numarul 1 <p>In acest caz, scorul lui Tuzgu este 6, scorul Ritzei este 4 iar diferenta este 2.</p> <p>Daca Tuzgu ar fi eliminat numarul 1 din stanga, Ritza ar fi eliminat numarul 5, apoi Tuzgu ar fi eliminat 3, iar Ritza ar fi incheiat jocul, eliminand ultimul numar egal cu 1.</p> <p>In acest caz, scorul lui Tuzgu ar fi fost 4, iar cel al Ritzei ar fi fost 6, diferenta obtinuta de Tuzgu fiind -2, deci nu ar fi fost strategia optima.</p>

BONUS

Enunt

Dupa ce au jucat cele 3 jocuri prezentate anterior, a venit timpul ca Tuzgu si Ritza sa joace un ultim joc. Cu toate astea, acesta este identic cu cel de la prima cerinta, cu mentiunea ca, de data aceasta, Tuzgu va propune un sir de N numere pe care cei doi sa joace jocul. Bineinteles, stiind ca este liber sa aleaga ce sir doreste, Tuzgu isi propune sa gaseasca un sir pentru care, daca cei doi ar juca optim jocul de la prima cerinta, ar obtine diferenta dintre scorul sau si scorul Ritzei egala cu K .

Deoarece exista o infinitate de siruri de lungime N pentru care diferenta maxima dintre scorurile celor doi este egala cu K , impunem doua restrictii: sirul trebuie sa contina N numere intregi **distincte**, iar acestea pot lua valori din intervalul $[1, V]$.

Cerinta

Se cere sa se determine numarul de siruri de N numere intregi **distincte**, cu valori in $[1, V]$, pentru care diferenta maxima dintre scorul lui Tuzgu si scorul Ritzei este exact K . Deoarece acest numar poate fi foarte mare, se cere restul impartirii acestui numar la 1000000007 ($10^9 + 7$).

Date de intrare

Pe prima si unica linie a fisierului **p4.in** se vor gasi trei numere intregi N , K , respectiv V .

Date de iesire

In fisierul **p4.out** se va scrie numarul de siruri care respecta proprietatile de mai sus, modulo 1000000007 ($10^9 + 7$).

Restrictii si precizari

- Pentru 25 de puncte, $1 \leq N, K, V \leq 100$
- Pentru inca 10 puncte, $1 \leq N, K, V \leq 10^5$

Testare si punctare

- Punctajul maxim este de 35 puncte.
- Timpul de executie:

- C/C++: **0.5 s**
- Java, Python: **1.0 s**
- Sursa care conține funcția **main** trebuie obligatoriu denumită:
p4.c, **p4.cpp** sau **p4.java**.

Exemple

Exemplu 1

Exemplu 1		
p4.in	p4.out	Explicatie
3 2 4	12	Cele 12 siruri sunt: 1. [3,2,1] 2. [4,1,3] 3. [1,4,3] 4. [1,2,3] 5. [2,3,1] 6. [3,1,4] 7. [1,3,2] 8. [3,1,2] 9. [4,3,1] 10. [1,3,4] 11. [2,1,3] 12. [3,4,1]