

Duplikált bináris részsorozatok, 1. rész

Egy nemüres T karaktersorozatot **duplikált bináris sorozatnak** nevezünk, ha csak a 0 és 1 karaktereket tartalmazza és felírható $T = \overline{S}S$ alakban, ahol S tetszőleges bináris karakterlánc és $\overline{ab}$ művelet az a és b karaktersorozatok konkatenációját (egymás utáni felírását egy sorozatként) jelenti. Például 0000 és 011011 duplikált bináris sorozatok, de 01, 0110 vagy 000 nem azok.

Adott egy N hosszú bináris karaktersorozat. Írj programot, ami megszámolja, hogy a sorozat hány **különböző** összefüggő duplikált bináris részsorozatot tartalmaz. Két részsorozatot akkor tekintünk különbözőnek, ha legalább egy karakterben eltérnek egymástól.

Bemenet

A standard bemenet első és egyetlen sorában egy bináris karaktersorozat található.

Kimenet

A standard kimenetre egyetlen egész számot kell kiírni, a bináris sorozat különböző összefüggő duplikált részsorozatainak a számát.

Példa

Bemenet	Kimenet
0101	1
Bemenet	Kimenet
0000	2

Az első példában a részsorozatok: 0 (két alkalommal), 1 (két alkalommal), 01 (két alkalommal), 10 (egy alkalommal), 010 (egy alkalommal), 101 (egy alkalommal), 0101 (egy alkalommal). Tehát egyetlen duplikált részsorozat van, a 0101.

A második példában a részsorozatok: 0 (négy alkalommal), 00 (három alkalommal), 000 (két alkalommal), 0000 (egy alkalommal). Két különböző duplikált részsorozat van, a 00 és a 0000.

Korlátok

$$4 \leq N \leq 100$$

Időlimit: 1.0 s

Memórialimit: 256 MB

Pontozás

A megoldásodat sok különböző tesztesetre lefuttatjuk. A tesztesetek részfeladatokba vannak csoportosítva. Egy-egy részfeladatot akkor tekintünk megoldottnak, ha volt legalább egy olyan beadásod, amely az adott részfeladat minden tesztesetére helyes megoldást adott. A feladat összpontszámát a megoldott részfeladatokra kapott pontszámok összege adja.

Részfeladat	Korlátok	Pontszám
0	a minta	0
1	$N = 4$	34
2	nincsenek további megkötések	66

Ebben a feladatban **részpontokat** lehet szerezni. Egy tesztesetre a részfeladat pontjainak 50%-a jár abban az esetben, ha a programod a helyes válasz helyett az összes (nem feltétlen különböző) duplikált részsorozatok darabszámát adja meg (például a második mintában 2 helyett 4-et ír ki, mert a 00 háromszor, a 0000 egyszer szerepel). Minden más esetben a tesztesetre 0 pont jár.

A részfeladatra kapott pontszám a hozzá tartozó tesztesetekre kapott pontszámok minimuma.

Számológép

Zorkának egy különleges számológépe van, mely egyetlen X változót kezel. A számológép bekapcsolásakor az X változó értéke 1.

A számológép processzora nagyon egyszerű, csak az alábbi három utasítást ismeri:

- ADD: az X értékéhez hozzáad 3-at.
- MUL: az X értékét megszorozza 2-vel.
- END: kiírja az X értékét a kijelzőre.

A feladatod meghatározni egy adott N pozitív egészhez azt a **legrövidebb** programot, aminek végrehajtása után a számológép kijelzőjén az N szám fog állni, vagy megállapítani, hogy nem létezik ilyen program.

Bemenet

A standard bemenet első és egyetlen sorában a kijelzőre írandó N pozitív egész érték szerepel.

Kimenet

A standard kimenet első sorába a legrövidebb program utasításainak K számát kell kiírni. A következő K sor mindegyike az ADD, MUL vagy END utasítások valamelyikét tartalmazza a végrehajtásuk sorrendjében. Több lehetséges megoldás esetén bármelyik megadható.

Ha nem létezik olyan program, aminek végrehajtása után a kijelzőn N áll, akkor egyetlen sorba a 0 számot írd ki.

Példa

Bemenet	Kimenet
10	4 MUL ADD MUL END
Bemenet	Kimenet
12	0

Korlátok

$$1 \leq N \leq 10^{18}$$

Időlimit: 1.0 s

Memórialimit: 256 MB

Pontozás

A megoldásodat sok különböző tesztesetre lefuttatjuk. A tesztesetek részfeladatokba vannak csoportosítva. Egy-egy részfeladatot akkor tekintünk megoldottnak, ha volt legalább egy olyan beadásod, amely az adott részfeladat minden tesztesetére helyes megoldást adott. A feladat összpontszámát a megoldott részfeladatokra kapott pontszámok összege adja.

Részfeladat	Korlátok	Pontszám
0	a minta	0
1	$N \leq 10$	10
2	$N \leq 100$	16
3	$N \leq 1\,000\,000$	37
4	nincsenek további megkötések	37

Imposztorok

Zoli kedvenc számítógépes játéka a 'Köztünk járnak'. A játékot egyszerre N ember játsza és mindenkinek van egy véletlenszerűen kiosztott szerepköre: **ártatlan** vagy **imposztor**. Tudjuk, hogy pontosan 3 imposztor van a játékban. Minden játékos ismeri a saját szerepkörét, valamint az imposztorok azt is tudják, hogy ki a másik két imposztor.

Az ártatlanok célja az imposztorok megtalálása. Ehhez időnként szavazásokat tartanak az összes játékos részvételével, ahol bármely játékos bármely másik játékost megvádolhat azzal, hogy az illető imposztor, majd ezt követően szavazással döntenek el, hogy valamelyik játékosokat kiejtsék-e a további játékból.

Kibicként néztél egy ideig, ahogy Zoli játszik, és arra lettél figyelmes, hogy az ártatlanok jellemzően bárkit megvádolhatnak szinte véletlenszerű módon, viszont az imposztorok szinte soha sem vádolják meg egymást.

Egy most kezdődött új játékban az első szavazáson az N játékos mindegyike pontosan egy másik játékost vádolt meg. Szeretnéd meghatározni, hogy mely játékosok lehetnek az imposztorok, feltételezve, hogy a megfigyelésed helyes, azaz imposztor sosem vádol meg másik imposztort (az ártatlanok bárkit megvádolhatnak).

Írj programot, ami meghatározza azoknak a játékoshármaknak a számát, akik imposztorok lehetnek.

Bemenet

A standard bemenet első sorában a játékosok N száma található.

A második sor N darab pozitív egészet tartalmaz, az i -edik szám az i sorszámú játékos által megvádolt játékos A_i sorszáma. A játékosokat 1-től N -ig sorszámozzuk.

Kimenet

A standard kimenetre egyetlen egész számot kell kiírni, azoknak a játékoshármaknak a darabszámát, akik a vádlások alapján imposztorok lehetnek.

Példa

Bemenet	Kimenet
5	1
2 3 1 3 2	
Bemenet	Kimenet
6	4
2 5 4 1 4 5	

Az első példában a három imposztor közül legalább egy az 1, 2, 3 játékosok valamelyike. Mivel ez a három játékos egymást vádolta meg ($1 \rightarrow 2 \rightarrow 3 \rightarrow 1$), így hármuk közül pontosan egy lesz imposztor, tehát a 4 és 5 játékosok biztosan imposztorok. Mivel ők a 2 és 3 sorszámú játékosokat vádolták, és imposztor nem vádolhat imposztort, ezért csak az 1-es játékos lehet a harmadik imposztor. Így egyetlen hármas lehetséges, az (1, 4, 5) játékosok az imposztorok.

A második példában a lehetséges imposztorok: (1, 3, 5), (1, 3, 6), (2, 3, 6) vagy (2, 4, 6).

Korlátok

$$4 \leq N \leq 100\,000$$

$1 \leq A_i \leq N$ és $A_i \neq i$ minden $i = 1 \dots N$ -re

Időlimit: 1.0 s

Memórialimit: 256 MB

Pontozás

A megoldásodat sok különböző tesztesetre lefuttatjuk. A tesztesetek részfeladatokba vannak csoportosítva. Egy-egy részfeladatot akkor tekintünk megoldottnak, ha volt legalább egy olyan beadásod, amely az adott részfeladat minden tesztesetére helyes megoldást adott. A feladat összpontszámát a megoldott részfeladatokra kapott pontszámok összege adja.

Részfeladat	Korlátok	Pontszám
0	a minta	0
1	$N \leq 100$	23
2	$N \leq 5000$	24
3	nincsenek további megkötések	53

Bindzsisztáni mindennapok

Lavrentij Bucservevics egy sikeres bindzsisztáni 'vállalkozó' remek kapcsolatokkal. A gazdasági válság azonban sajnos őt is utolérte és a vállalkozásai veszteségesek lettek. Lavrentijnek összesen N cége van amiket 1-től N -ig sorszámozunk. Az i -edik cég adóssága A_i millió forint.

Szerencsére az államigazgatás korrupt, így barátai, ismerősei és a jóisten segítségével el tudta érni, hogy az állam átvegyen tőle néhány céget, mentesítve őt az adósságok alól. Úgy döntöttek, hogy az L sorszámu cégtől az R sorszámu cégig megtarthatja Lavrentij a vállalkozásokat, az összes többi állami kézbe kerül.

Szerencsére az adóhatóság is korrupt, így Lavrentij még az átvétel előtt le tud fizetni néhány ügyintézőt, akik hajlandóak cégpárok azonosítóit kicserélni egymással. Ez a művelet nem túl egyszerű és kockázattal jár, így az i és j sorszámu cégek azonosítójának felcseréléséért Lavrentijnek $|i - j|$ millió forintot kell fizetnie.

Sajnos Lavrentijnek jelenleg csak K millió forint készpénze van. Segíts neki meghatározni, hogy legfeljebb ekkora összeg elköltésével hogyan tudja úgy megkenni az ügyintézőket, hogy a nevében maradó cégek adósságainak az összege minimális legyen.

Bemenet

A standard bemenet első sorában a vállalkozások N száma, a Lavrentij tulajdonában maradó cégeket megadó L és R értékek, valamint a rendelkezésre álló K pénzösszeg található.

A második sor N darab nemnegatív egészet tartalmaz, a vállalkozások A_i adósságait.

Kimenet

A standard kimenetre egyetlen egész számot kell kiírni, a Lavrentij nevében maradó vállalkozások összes adósságát ($A_L + A_{L+1} + \dots + A_R$ összeget) optimális sorszámcsere végrehajtása után.

Példa

Bemenet	Kimenet
3 2 2 1 1 2 3	1
Bemenet	Kimenet
5 2 3 3 21 54 12 2 0	12
Bemenet	Kimenet
6 4 6 100 1 2 3 4 5 6	6

Az első példában Lavrentij csak a 2-es számú vállalkozást tarthatja meg. Az állami átvétel előtt 1 millió forintért kicserélteti az 1-es és 2-es számú vállalkozások sorszámait, így csak 1 millió forint lesz a megtartott cég adóssága.

A második példában Lavrentij a 2-es és 3-as számú vállalkozásokat tarthatja meg. A 2-es és 5-ös sorszámokat 3 millióért ki tudja cseréltetni, így $0 + 12 = 12$ millió lesz a cégei adóssága. Belátható, hogy ennél kisebb adósság nem érhető el legfeljebb $K = 3$ millió forint elköltésével.

Korlátok

$$1 \leq N \leq 100$$

$$1 \leq L \leq R \leq N$$

$$0 \leq K \leq 10\,000$$

$$0 \leq A_i \leq 1\,000\,000 \text{ minden } i = 1 \dots N\text{-re}$$

Időlimit: 2.0 s**Memórialimit:** 256 MB**Pontozás**

A megoldásodat sok különböző tesztesetre lefuttatjuk. A tesztesetek részfeladatokba vannak csoportosítva. Egy-egy részfeladatot akkor tekintünk megoldottnak, ha volt legalább egy olyan beadásod, amely az adott részfeladat minden tesztesetére helyes megoldást adott. A feladat összpontszámát a megoldott részfeladatokra kapott pontszámok összege adja.

Részfeladat	Korlátok	Pontszám
0	a minta	0
1	$N \leq 13$ és $R = N$	20
2	$N \leq 50$ és $R = N$	30
3	$N \leq 50$	30
4	nincsenek további megkötések	20

Duplikált bináris részsorozatok, 2. rész

Ez egy 'output only' feladat, ami azt jelenti, hogy a megoldást előállító programot nem, csak a program kimenetét kell beadnod az értékelő rendszernek.

Egy nemüres T karaktersorozatot **duplikált bináris sorozatnak** nevezünk, ha csak a 0 és 1 karaktereket tartalmazza és felírható $T = \overline{SS}$ alakban, ahol S tetszőleges bináris karakterlánc és $\overline{ab}$ művelet az a és b karaktersorozatok konkatenációját (egymás utáni felírását egy sorozatként) jelenti. Például 0000 és 011011 duplikált bináris sorozatok, de 01, 0110 vagy 000 nem azok.

Egy bináris karaktersorozat **erőssége** a sorozat **különböző** összefüggő duplikált bináris részsorozatainak a száma. Két részsorozatot akkor tekintünk különbözőnek, ha legalább egy karakterben eltérnek egymástól.

A feladatod, hogy megkeresd az $N = 100$ hosszú bináris sorozatok közül a leggyengébbet és a legerősebbet (azaz azokat, melyek erőssége minimális, illetve maximális). A kapott pontszámod attól függ, hogy mennyire erős sorozatokat találtál (ld. Pontozás rész).

Kimenet

Két fájlt kell leadnod, melyek közül az első fájl a legkisebb, a második fájl a legnagyobb erősségű 100 hosszú bináris karaktersorozatot tartalmazza egyetlen sorban.

Egy beadás során beadható csak az egyik fájl a kettőből, ekkor az értékelés során

- ha a másik fájl korábban már be lett adva, akkor a legutoljára beadott tartalma kerül értékelésre;
- ha a másik fájl korábban még nem lett leadva, akkor az arra szereshető pontokból 0 pontot kapsz.

Fájlankénti feltöltés esetén a feltöltött fájlok neve tetszőleges.

Lehetőség van a két fájlt egyetlen tömörített zip fájlként is beadni, ekkor ügyelni kell arra, hogy az archívumban a két fájl elnevezése `output_weakest.txt` és `output_strongest.txt` legyen.

Pontozás

Ha a beadott fájl nem felel meg a formai követelményeknek, azaz nem csak egyetlen 100 hosszú karaktersorozatot tartalmaz, vagy a sorozat tartalmaz 0-tól és 1-től eltérő karaktert is, akkor 0 pontot kap.

A leggyengébb bitsorozat megadásával legfeljebb 30 pont szereshető. Legyen X a beadott bitsorozat erőssége. Ekkor a szerzett pontszám:

- $25 \leq X$ esetén 0 pont;
- $5 \leq X < 25$ esetén $25 - X$ pont;
- $X = 4$ esetén 25 pont;
- $X = 3$ esetén 30 pont.

A legerősebb bitsorozat megadásával legfeljebb 70 pont szereshető. Legyen X a beadott bitsorozat erőssége. Ekkor a szerzett pontszám:

- $X \leq 50$ esetén 0 pont;
- $50 < X \leq 80$ esetén $X - 50$ pont;

- $X = 81$ esetén 35 pont;
- $X = 82$ esetén 40 pont;
- $X = 83$ esetén 45 pont;
- $X = 84$ esetén 70 pont.