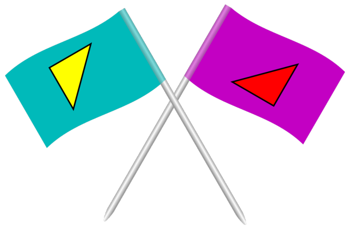
Norveksit

I flere århundre har Norvestige og Sørøstia dannet kongeriket Firkantia. Etter mye politisk uenighet i de senere årene har Novestige bestemt seg for å bryte ut og bli et selvstendig land igjen, i en prosess som kalles Norveksit. Din oppgave er å hjelpe politikerene med å finne ut av hvilke byer som skal tilhøre Norvestige og hvilke som skal tilhøre Sørøstia, slik at prosessen kan gjennomføres så smertefritt som mulig.



Firkantia er formet som et rutenett, med et hjørne i punktet $(0, 0)$ og som strekker seg uendelig langt mot øst og nord. Punktet 5 meter nord og 3 meter øst for dette hjørnet er altså $(3, 5)$. Byene i Firkantia er formet som rektangeler med heltallskoordinater på alle sidene. Grensen mellom Norvestige og Sørøstia skal være en rett linje som går diagonalt mot nordøst gjennom punktet $(0, 0)$ - altså gjennm alle punkter (x, y) hvor $x = y$. Se bildet i eksempelet dersom noe er uklart.

Hvor mange av byene kommer til å ligge i Norvestige? Dersom noen byer vil ligge delvis i Norvestige og Sørøstia så vil ikke Norveksit kunne gjennomføres, og du skal skrive ut ordet `UMULIG`

Input

Første linje inneholder et heltall N - antall byer i Firkantia. Deretter følger N linjer som hver beskriver en by. By nummer i er beskrevet med 4 heltall S_i, N_i, V_i og $\O_i$ som er henholdsvis hvor langt nord, sør, øst og vest byen strekker seg.

Output

Dersom ingen av byene krysser den foreslåtte grensen mellom Norvestige og Sørøstia, skriv ut antall byer som ligger i Norvestige. Dersom noen av byene krysser den foreslåtte grensen, skriv ut ordet `UMULIG`.

Begrensninger

$2 \leq N \leq 50\ 000$

$0 \leq S_i < N_i \leq 1\ 000\ 000$ for alle $0 \leq i < N$

$0 \leq V_i < \O_i \leq 1\ 000\ 000$ for alle $0 \leq i < N$

Ingen av byene vil overlappe hverandre, men de kan ha felles grenser.

Tidsbegrensning 1 s.

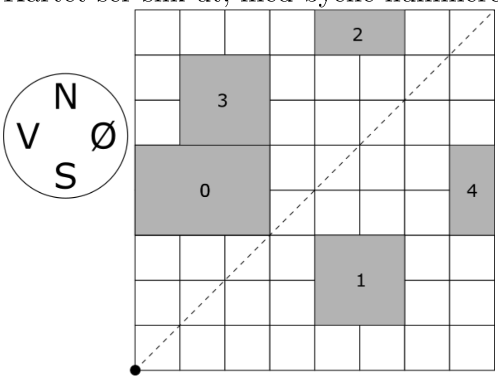
Testsettgruppe	**Poeng**	**Ytligere begrensninger**
Gruppe 1	20	$N_i - S_i \leq 10$ og $Ø_i - V_i \leq 10$ for alle i
Gruppe 2	35	$N_i - S_i \leq 1000$ og $Ø_i - V_i \leq 1000$ for alle i
Gruppe 3	45	Ingen andre begrensninger

Eksempler

Input	**Output**

`5`
`3 5 0 3`
`1 3 4 6`
`7 8 4 6`
`5 7 1 3`
`3 5 7 8`
	`3`

Kartet ser slik ut, med byene nummeret 0 til 4 etter rekkefølgen de har i input:



3 av byene er i området som vil tilhøre Norvestige.

Input	**Output**

`3`
`5 7 1 3`
`3 5 3 5`
`1 3 5 7`
	`UMULIG`

Rettferdig fordeling

Lise og Lotte har funnet onkel David sin samling med gamle samlekort. Mange av kortene er sjeldne og verdifulle, og Lise og Lotte har spurt David om de kan få noen av de. David gir de gjerne bort, men vet at det blir mye krangling hvis ikke Lise og Lotte får kort til nøyaktig lik samlet verdi. Dette betyr at det kan hende han ikke kan gi bort alle kortene.

Hva er den største samlede verdien han kan gi til hver av de, slik at de begge får kort til nøyaktig samme verdi?

Input

Første linje inneholder et heltall N, antall kort i samlingen. Deretter følger N linjer hver med et heltall V_i - verdien på kort nummer i.

Output

Et tall S, høyeste mulige samlede verdi på kortene som David kan gi bort

Begrensninger

$1 \leq N \leq 200$

$1 \leq V_i \leq 2\,500$ for alle i

Tidsbegrensning 4 s.

Testsettgruppe	**Poeng**	**Ytligere begrensninger**
Gruppe 1	17	$N \leq 15$
Gruppe 2	19	$N \leq 26$
Gruppe 3	25	Alle V_i er toerpotenser (altså kan skrives på formen 2^k hvor k er et heltall)
Gruppe 4	39	Ingen andre begrensninger

Eksempler

Input	Output	Kommentarer
5 7 3 3 3 6	6	F.eks. kan lise få kortet med verdi 6 og Lotte kan få to av kortene med verdi 3

Input	Output
5 5 2 3 4 5	7

Input	Output
10 1 2 3 4 5 6 7 8 9 10	27

Snømenn

Du har rullet mange snøballer som du ønsker å sette sammen til så mange snømenn som mulig. Alle snømenn består av nøyaktig 3 snøballer - en stor nederst, en mindre snøball oppå den, og en enda mindre øverst som hode. For at snømannen ikke skal kollapse så kan ingen av de to øverste snøballene være mer enn halvparten så store som snøballen som den står på. F.eks. gir baller med størrelse $14, 7, 3$ en gyldig snømann, siden $14 \geq 2 \times 7$ og $7 \geq 2 \times 3$. $14, 8, 3$ er ikke en gyldig snømann fordi $14 < 2 \times 8$.

Gitt et sett med snøballer så lurer du på hvor mange snømenn det er mulig å lage samtidig.

Input

Første linje inneholder et heltall N - antall snøballer. Deretter følger N linjer, hver med ett heltall S_i - størrelsen på snøball nummer i.

Output

Et heltall - antall snømenn det er mulig å lage med de gitte snøballene.

Begrensninger

$1 \leq N \leq 100\ 000$

$1 \leq S_i \leq 1\ 000\ 000\ 000$ for alle i.

Tidsbegrensning 2 s.

Testsettgruppe	**Poeng**	**Ytligere begrensninger**
Gruppe 1	15	$1 \leq N \leq 21$
Gruppe 2	27	Minst en tredjedel av snøballene har størrelse 1
Gruppe 3	28	$1 \leq N \leq 4\ 000$
Gruppe 4	30	Ingen andre begrensninger

Eksempler

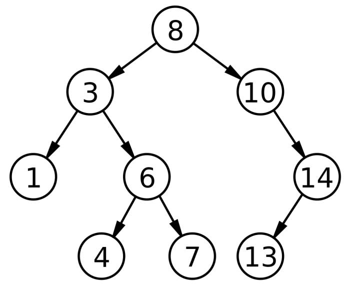
Input	**Output**	**Kommentarer**

`10`
`1`
`2`
`3`
`4`
`5`
`6`
`7`
`8`
`9`
`10`
	`2`	F.eks. kan du lage snømennene
		`1 5 10`
		`2 4 8`

Input	**Output**	**Kommentarer**

`9`
`1`
`3`
`10`
`4`
`1`
`1`
`12`
`5`
`11`
	`3`	Dette er et eksempel fra testsettgruppe 2.

Binært Søketre



Et binært søketre er en enkal datastruktur for å lagre tall i en sortert rekkefølge. Et tre kan beskrives rekursivt som å være enten

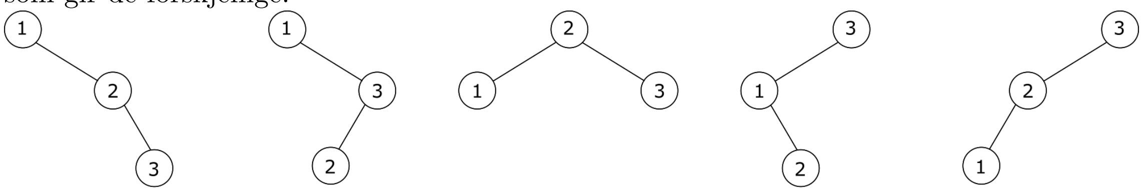
(A) Et tomt tre

(B) En node n som inneholder en tallverdi, og som har to trær som henholdsvis venstre og høyre barn. Disse barna kan være på enten form (A) eller (B). Dersom venstre barn ikke er et tomt tre så må verdien i noden dens være mindre enn verdien i node n. Tilsvarende hvis høyre barn ikke er et tomt tre så må verdien i noden dens ha en verdi som er større enn verdien i noden til n.

Følgende rekursive prosess kan brukes for å sette inn et tall X i et tre T:
Dersom T er tomt, ersatt det med en node med verdien X og tomme subtrær som venstre og høyre barn.
Dersom T ikke er er tomt og X er mindre enn tallet i noden til T, sett det inn i venstre barn av T.
Dersom T ikke er er tomt og X er større enn tallet i noden til T, sett det inn i høyre barn av T.

(I denne oppgaven så vil vi ikke se på tilfeller hvor to tall som skal settes inn i treet kan være like.)

Rekkefølgen på tallene man setter inn kan påvirke hvilket tre man ender opp med. Under vises alle mulige binære søketrær med tallene 1, 2, og 3, og hvilke innsettingsrekkefølger som gir de forskjellige.



1, 2, 3   1, 3, 2   2, 1, 3 eller 2, 3, 1   3, 1, 2   3, 2, 1

Gitt en sekvens S med tall skal du finne ut hvor mange andre sekvenser som vil gi nøyaktig det samme binære søketreet som S. Fordi dette tallet kan være veldig høyt skal du oppgi svaret modulo et stort primtall - nemmelig 200 009. (Dvs. hvis antall sekvenser er over 200 009 så skal du oppgi resten når dette tallet deles med 200 009. Feks. hvis antall sekvenser er 400 025 skal du skrive ut 7 fordi 400 025 $= 2 \times 200\,009 + 7$. Modulo kan regnes ut med %-operatoren i de fleste programeringsspråk.)

Input

Første linje inneholder et heltall N, antall tall i sekvensen. Neste linje inneholder sekvensen S som består av alle de N tallene fra 1 til og med N i en eller annen rekkefølge.

Output

Et heltall - antall andre sekvenser som gir likt tre som S, modulo 200 009.

Begrensninger

$1 \leq N \leq 3\ 000$

Tidsbegrensning 2 s.

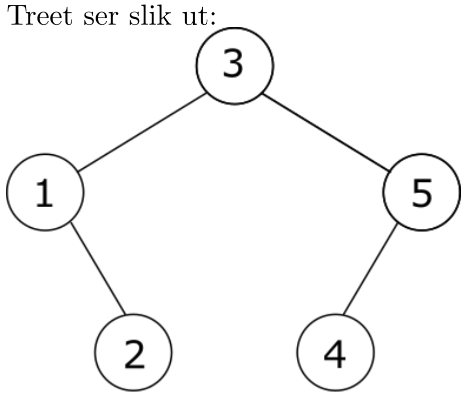
Testsettgruppe	Poeng	Ytligere begrensninger
Gruppe 1	17	$1 \leq N \leq 7$
Gruppe 2	27	$1 \leq N \leq 14$
Gruppe 3	22	$1 \leq N \leq 100$
Gruppe 4	34	Ingen andre begrensninger

Eksempler

Input	Output	Kommentarer
`3` `2 1 3`	`1`	`2 3 1` gir samme tre

Input	Output
`3` `3 1 2`	`0`

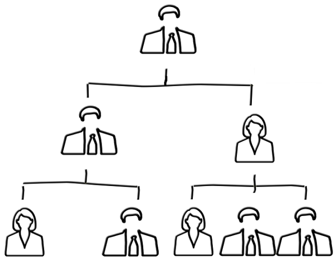
Input	**Output**
`5` `1 2 3 4 5`	`0`

Input	**Output**	**Kommentarer**
`5` `3 5 4 1 2`	`5`	Treet ser slik ut: 

Input	**Output**
`7` `4 2 6 1 3 5 7`	`79`

.

Stormannsgalskap



Fredrik er toppsjef i eget firma. Firmaet består av ham selv, nummerert 0, og N ansatte, nummerert fra 1 til og med N. Fredrik har planlagt en total omstrukturering av bedriften. Ingen ansatte vet noe som helst om hvor de ender opp i hierarkiet. Derfor starter alle ansatte med 0 i stormannsgalskap. Fredrik har laget en plan over hvordan bedriften skal se ut. Planen er et rotfestet tre med ham selv som rot. Alle ansatte skal få en sjef, som enten er Fredrik selv, eller en annen ansatt. Dersom b er sjefen til a sier vi at a er direkte underordnet b. Hvis b har en sjef c så er også a indirekte underordnet c og alle andre som evt. c er underordnet av. Treet er sammenhengende, slik at alle ansatte enten er direkte eller indirekte underordnet Fredrik.

Fredrik skal publisere planen sin, men publiserer den én kant av gangen. Først skal ansatt nummer 1 få vite sjefen sin, så skal ansatt 2 få vite sin, så 3, 4, 5 osv. til og med ansatt N. Hver gang en ny ansatt-sjef-relasjon offentliggjøres, øker stormannsgalskapen til alle som har fått flere direkte eller indirekte underordnede. I det øyeblikket en ansatt a får m flere underordnede, øker stormannsgalskapen til a med $m \times U$. Merk at hvis b allerede vet at han er direkte eller indirekte overordnet til a så vil dette også gjøre at b også får m nye underordenede, og dermed også $m \times U$ mer i stormannsgalskap.

For å ikke la galskapen komme ut av kontroll kan Fredrik velge å vente én eller flere dager fra han publiserer én ansatts sjef til den neste ansattes sjef publiseres. For hver dag som går vil stormannsgalskapen til hver ansatt med stormannsgalskap synke med D, med en nedre grense på 0.

Fredrik er redd for at om noen av de ansatte blir for stormannsgale så vil de prøve å ta over hele firmaet. Han ønsker dermed at det ikke på noe tidspunkt skal være noen andre enn ham som har mer enn K i stormannsgalskap.

Omstruktureringen begynner på dag 1. Hvilken dag kan omstruktureringen tidligst være ferdig, hvis det i det hele tatt er mulig?

Input

Første linje inneholder heltallene N, K, U og D slik defniert over.

Deretter følger N linjer, nummerert fra $i = 1$ til og med $i = N$. Den i-te av disse inneholder et heltall S_i - sjefen til ansatt nummer i.

Output

Én linje med ett heltall, hvilken dag prosessen tidligst kan være ferdig, eller `UMULIG` hvis det ikke er mulig.

Begrensninger

$1 \leq N \leq 100\ 000$

$1 \leq K \leq 1\ 000\ 000$

$0 \leq U \leq 1\ 000\ 000$

$0 \leq D \leq 1\ 000\ 000$

$0 \leq S_i \leq N$ for alle $1 \leq i \leq N$

Tidsbegrensning 1 s.

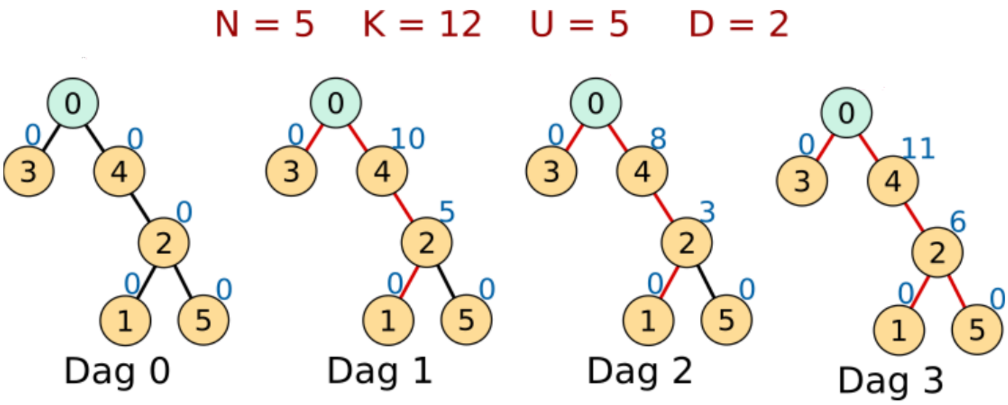
Testsettgruppe	**Poeng**	**Ytligere begrensninger**
Gruppe 1	10	$N \leq 100, K = 10 \times U, D = 0$
Gruppe 2	15	$N \leq 100, D = N \times U$
Gruppe 3	20	$N \leq 500, U = D$
Gruppe 4	20	Ingen ansatte har flere enn 8 direkte eller indirekte overordnede.
Gruppe 5	35	Ingen andre begrensninger

Eksempler

Input	**Output**

`5 12 5 2`
`2`
`4`
`0`
`0`
`2`
	`3`

Organisasjonskartet ser ut som vist under.



Fredrik kan publisere (markert i rødt) de fire første kantene på dag 1, men må vente helt til dag 3 før han trygt kan publisere den siste.

Input	Output

`4 5 3 6`
`0`
`3`
`1`
`0`
	`UMULIG`

Input	Output

`5 9 3 0`
`2`
`0`
`2`
`3`
`0`
	`1`