

Kalender



Opgave

We hebben het allemaal veel te druk tijdens de vakantie, en het wordt een hele klus om de dagen te vinden waarop zowel wijzelf als onze vrienden nog vrij zijn. Een programma dat een *eerste* gemeenschappelijke vrije periode zoekt in de kalenders van een aantal personen is daarom zeer nuttig. Beschouw in het volgende voorbeeld de eerste 14 dagen van de kalender van drie personen, waarbij de dagen van elke persoon op een andere regel getoond worden. Voor elke dag wordt aangegeven of de betreffende persoon al iets heeft gepland op die dag. Met een V wordt aangeduid dat de persoon *niets* gepland heeft en dus vrij is; een B betekent daarentegen dat er op die dag *wel* al iets is gepland en de persoon dus bezet is. In tekstformaat ziet dit er dan als volgt uit.

```
VVVBVVBBBBBVV
```

```
BVVBVVBBBBBVV
```

```
VVBBVVBBBBBVV
```

Het programma moet op zoek gaan naar de *eerste* periode die bestaat uit één dag of meerdere opeenvolgende dagen waarop iedereen vrij is (m.a.w. op elke kalender staat een V). Indien een dergelijke periode gevonden wordt, dan toont het programma het eerste dagnummer van die periode, waarbij we beginnen te tellen vanaf 1

Stel dat er voor het bovenstaande voorbeeld gevraagd wordt of er een periode van 2 dagen bestaat waarop alle drie personen vrij zijn. Het antwoord is dan ja, en het dagnummer is 5. Er is nog een tweede periode van minstens 2 opeenvolgende dagen met als eerste dagnummer 13, maar we plannen liefst zo vroeg mogelijk en willen dus het *eerste dagnummer* kennen. Indien er geen gemeenschappelijke vrije periode bestaat, schrijft het programma X uit. In het bovenstaande voorbeeld gebeurt dat als een periode van 3 opeenvolgende vrije dagen gezocht wordt.

Invoer

De eerste regel van de invoer bevat een geheel getal $1 \leq n \leq 1000$ dat het aantal testgevallen aangeeft. Per geval volgen dan een aantal regels. Alle letters op dezelfde regel worden aaneensluitend gegeven, m.a.w. er komen geen spaties voor. Alle regels worden beëindigd met een enkele newline \n.

Elke opgave bestaat uit op haar beurt uit $1 + 1 + 1 + p$ regels, waarbij het volgende formaat wordt gehanteerd.

- De eerste regel bevat het aantal personen $1 \leq p \leq 1000$.
- De tweede regel bevat het aantal dagen $1 \leq d \leq 1000$ in de kalender van elke persoon.
- De derde regel bevat de lengte $1 \leq l \leq d$ (dus het aantal dagen) van de op te zoeken vrije periode.
- Daarna volgen p regels die elke exact d karakters bevatten. Elk karakter stelt de planning voor 1 persoon op 1 bepaalde dag voor. Zoals hogerop vermeld staat er een V voor een vrije dag en een B voor een bezette dag.

Uitvoer

Voor elk testgeval dient je programma exact 1 regel uit te schrijven met daarop het dagnummer van de eerste dag van de vroegste vrije periode of X. De uitvoer van alle testgevallen wordt in de juiste volgorde onder elkaar geschreven naar het standaard uitvoerkanaal. Voor elke opgave bestaat de uitvoer uit 1 regel. Die regel vermeldt het dagnummer van de eerste dag van de *eerste* gevonden vrije periode. Wanneer er dus meerdere vrije periodes zijn dan moet je het dagnummer van de vroegste vrije periode tonen. Wanneer er geen vrije periode gevonden werd, dan wordt X getoond. In totaal schrijf je dus n regels uit.

Let op! Zorg ervoor dat je uitvoer geen overbodige tekens bevat, bijvoorbeeld een spatie op het einde van een regel of een lege regel op het einde van de uitvoer. Dat zorgt er immers voor dat je uitvoer als foutief wordt beschouwd.

Voorbeeld

Invoer

```
2
3
14
2
VVVBVVBBBBBVVV
BVVBVVBBBBBVVV
VVBBVVBBBBBVVV
7
15
```

3
VVVBVVVVVVVVVB
BVVBVVVBVVVVV
VVBBBVBBVVBBVV
VBVBVVVVVVBBVV
VVVVVVVVVVVVVV
VVBBBVBBBBBVVV
VBBBVVBVVBBVVV

Uitvoer

5
X