Lausnir á völdum dæmum úr viku tvö

Bergur Snorrason

27. janúar 2022

- Ég mun leysa eftirfarandi dæmi:

- Ég mun leysa eftirfarandi dæmi:
  - *Reiknirit*,

- Ég mun leysa eftirfarandi dæmi:
  - *Reiknirit*,
  - *Pipe Rotation*,

- Ég mun leysa eftirfarandi dæmi:
  - *Reiknirit*,
  - *Pipe Rotation*,
  - *Language Survey*.

Reiknirit

- Skoðum aftur skiladæmið *Reiknirit*.

Reiknirit

- Skoðum aftur skiladæmið *Reiknirit*.
- Fyrsta lína inntaksins inniheldur heiltölu n.

Reiknirit

- Skoðum aftur skiladæmið *Reiknirit*.
- Fyrsta lína inntaksins inniheldur heiltölu n.
- Síðan koma n heiltölur $a_1, a_2, \ldots, a_n$.

Reiknirit

- Skoðum aftur skiladæmið *Reiknirit*.
- Fyrsta lína inntaksins inniheldur heiltölu n.
- Síðan koma n heiltölur $a_1, a_2, \ldots, a_n$.
- Gerum ráð fyrir að við séum með forrit sem gerir eftirfarandi:

Reikniril

- Skoðum aftur skiladæmið *Reikniril*.
- Fyrsta lína inntaksins inniheldur heiltölu n.
- Síðan koma n heiltölur $a_1, a_2, \ldots, a_n$.
- Gerum ráð fyrir að við séum með forrit sem gerir eftirfarandi:
  - Prentar tölurnar.

Reiknirit

- Skoðum aftur skiladæmið *Reiknirit*.
- Fyrsta lína inntaksins inniheldur heiltölu n.
- Síðan koma n heiltölur $a_1, a_2, \ldots, a_n$.
- Gerum ráð fyrir að við séum með forrit sem gerir eftirfarandi:
  - Prentar tölurnar.
  - Fjarlægir öll eintök af algengustu tölunni í listan.

Reiknirit

- Skoðum aftur skiladæmið *Reiknirit*.
- Fyrsta lína inntaksins inniheldur heiltölu n.
- Síðan koma n heiltölur $a_1, a_2, \ldots, a_n$.
- Gerum ráð fyrir að við séum með forrit sem gerir eftirfarandi:
  - Prentar tölurnar.
  - Fjarlægir öll eintök af algengustu tölunni í listan.
  - Endurtekur skrefin að ofan þar til listinn er tómur.

Reiknirit

- Skoðum aftur skiladæmið *Reiknirit*.
- Fyrsta lína inntaksins inniheldur heiltölu n.
- Síðan koma n heiltölur $a_1, a_2, \ldots, a_n$.
- Gerum ráð fyrir að við séum með forrit sem gerir eftirfarandi:
  - Prentar tölurnar.
  - Fjarlægir öll eintök af algengustu tölunni í listan.
  - Endurtekur skrefin að ofan þar til listinn er tómur.
- Dæmið snýst um að finna hversu margar tölur eru prentaðar í heildina.

Reiknirit

- Síðast leystum við þetta dæmi með því að útfæra forritið sem er lýst í dæminu.

Reiknirit

- Síðast leystum við þetta dæmi með því að útfæra forritið sem er lýst í dæminu.
- Við komumst þó að því að sú lausn var $\mathcal{O}(n^2)$ sem reyndist of hæg.

Reiknirit

- Síðast leystum við þetta dæmi með því að útfæra forritið sem er lýst í dæminu.
- Við komumst þó að því að sú lausn var $\mathcal{O}(n^2)$ sem reyndist of hæg.
- Tökum eftir að við getum notað svipaða hugmynd og í hægu útfærslunni til að telja hversu oft hver tala kemur fyrir.

Reiknirit

- Síðast leystum við þetta dæmi með því að útfæra forritið sem er lýst í dæminu.
- Við komumst þó að því að sú lausn var $\mathcal{O}(n^2)$ sem reyndist of hæg.
- Tökum eftir að við getum notað svipaða hugmynd og í hægu útfærslunni til að telja hversu oft hver tala kemur fyrir.
- Tökum einnig eftir að hvert eintak af algengustu tölunni er prentað einu sinni, hvert eintak af næst algengustu tölunni er prentað tvisvar og svo framvegis.

Reiknirit

- Síðast leystum við þetta dæmi með því að útfæra forritið sem er lýst í dæminu.
- Við komumst þó að því að sú lausn var $\mathcal{O}(n^2)$ sem reyndist of hæg.
- Tökum eftir að við getum notað svipaða hugmynd og í hægu útfærslunni til að telja hversu oft hver tala kemur fyrir.
- Tökum einnig eftir að hvert eintak af algengustu tölunni er prentað einu sinni, hvert eintak af næst algengustu tölunni er prentað tvisvar og svo framvegis.
- Einnig skiptir talan sjálf ekki máli, heldur eingöngu hversu oft hún kemur fyrir.

Reiknirit

- Látum því $h_1 \geq h_2 \geq \cdots \geq h_k$ þannig að algengast talan kemur h_1 sinni fyrir, næst algengasta talan kemur h_2 sinnum fyrir og svo framvegis.

Reiknirit

- Látum því $h_1 \geq h_2 \geq \cdots \geq h_k$ þannig að algengast talan kemur h_1 sinni fyrir, næst algengasta talan kemur h_2 sinnum fyrir og svo framvegis.
- Svarið er því

$$\sum_{i=1}^{k} i \cdot h_i.$$

Reiknirit

- Látum því $h_1 \geq h_2 \geq \cdots \geq h_k$ þannig að algengast talan kemur h_1 sinni fyrir, næst algengasta talan kemur h_2 sinnum fyrir og svo framvegis.
- Svarið er því

$$\sum_{i=1}^{k} i \cdot h_i.$$

- Við þurfum þó að passa okkur aðeins.

Reiknirit

- Látum því $h_1 \geq h_2 \geq \cdots \geq h_k$ þannig að algengast talan kemur h_1 sinni fyrir, næst algengasta talan kemur h_2 sinnum fyrir og svo framvegis.
- Svarið er því
$$\sum_{i=1}^{k} i \cdot h_i.$$
- Við þurfum þó að passa okkur aðeins.
- Ef $h_1 = h_2 = \ldots h_k = 1$ (þá er einnig $k = n$) fæst að svarið er
$$\sum_{i=1}^{k} i \cdot h_i = \sum_{i=1}^{k} i = \frac{n \cdot (n+1)}{2}.$$

Reiknirit

- Látum því $h_1 \geq h_2 \geq \cdots \geq h_k$ þannig að algengast talan kemur h_1 sinni fyrir, næst algengasta talan kemur h_2 sinnum fyrir og svo framvegis.
- Svarið er því
$$\sum_{i=1}^{k} i \cdot h_i.$$
- Við þurfum þó að passa okkur aðeins.
- Ef $h_1 = h_2 = \ldots h_k = 1$ (þá er einnig $k = n$) fæst að svarið er
$$\sum_{i=1}^{k} i \cdot h_i = \sum_{i=1}^{k} i = \frac{n \cdot (n+1)}{2}.$$
- Svo, þar sem n getur verið allt að 10^6, getur svarið okkar orðið of stórt fyrir `int`.

Reiknirit

- Látum því $h_1 \geq h_2 \geq \cdots \geq h_k$ þannig að algengast talan kemur h_1 sinni fyrir, næst algengasta talan kemur h_2 sinnum fyrir og svo framvegis.
- Svarið er því
$$\sum_{i=1}^{k} i \cdot h_i.$$
- Við þurfum þó að passa okkur aðeins.
- Ef $h_1 = h_2 = \ldots h_k = 1$ (þá er einnig $k = n$) fæst að svarið er
$$\sum_{i=1}^{k} i \cdot h_i = \sum_{i=1}^{k} i = \frac{n \cdot (n+1)}{2}.$$
- Svo, þar sem n getur verið allt að 10^6, getur svarið okkar orðið of stórt fyrir `int`.
- Við þurfum því að nota `long long`.

Reiknirit

```c
#include <stdlib.h>
#include <stdio.h>
typedef long long ll;

#define CMP(E, F) (F <= E) - (E <= F)
int cmp(const void* p1, const void* p2)
{
    return CMP(*(ll*)p2, *(ll*)p1);
}

int main()
{
    ll i, j, k, r = 0, n;
    scanf("%lld", &n);
    ll a[n], b[n];
    for (i = 0; i < n; i++) scanf("%lld", &a[i]);
    qsort(a, n, sizeof *a, cmp);
    i = k = 0;
    while (i < n)
    { // teljum hvað hver tala kemur oft fyrir.
        j = i;
        while (j < n && a[i] == a[j]) j++;
        b[k++] = j - i;
        i = j;
    }
    qsort(b, k, sizeof *b, cmp);
    for (i = 0; i < k; i++) r = r + (i + 1)*b[i];
    printf("%lld\n", r);
    return 0;
}
```

Reiknirit

- Sjáum við byrjum á að raða, sem er $\mathcal{O}(\qquad)$.

Reiknirit

- Sjáum við byrjum á að raða, sem er $\mathcal{O}(n \log n)$.

Reiknirit

- Sjáum við byrjum á að raða, sem er $\mathcal{O}(n \log n)$.
- Við teljum síðan hvað hver tala kemur oft fyrir, sem er $\mathcal{O}(\quad)$.

Reiknirit

- Sjáum við byrjum á að raða, sem er $\mathcal{O}(n \log n)$.
- Við teljum síðan hvað hver tala kemur oft fyrir, sem er $\mathcal{O}(\ n\ )$.

Reiknirit

- Sjáum við byrjum á að raða, sem er $\mathcal{O}(n \log n)$.
- Við teljum síðan hvað hver tala kemur oft fyrir, sem er $\mathcal{O}(\ n\ )$.
- Síðan röðum við aftur.

Reiknirit

- Sjáum við byrjum á að raða, sem er $\mathcal{O}(n \log n)$.
- Við teljum síðan hvað hver tala kemur oft fyrir, sem er $\mathcal{O}(\ \ n\ \ )$.
- Síðan röðum við aftur.
- Að lokum reiknum við summuna í $\mathcal{O}(\qquad)$.

Reiknirit

- Sjáum við byrjum á að raða, sem er $\mathcal{O}(n \log n)$.
- Við teljum síðan hvað hver tala kemur oft fyrir, sem er $\mathcal{O}(\ n\ )$.
- Síðan röðum við aftur.
- Að lokum reiknum við summuna í $\mathcal{O}(\ n\ )$.

Reiknirit

- Sjáum við byrjum á að raða, sem er $\mathcal{O}(n \log n)$.
- Við teljum síðan hvað hver tala kemur oft fyrir, sem er $\mathcal{O}(\ n\ )$.
- Síðan röðum við aftur.
- Að lokum reiknum við summuna í $\mathcal{O}(\ n\ )$.
- Tímaflækjan er því í heildina $\mathcal{O}(\qquad)$.

Reiknirit

- Sjáum við byrjum á að raða, sem er $\mathcal{O}(n \log n)$.
- Við teljum síðan hvað hver tala kemur oft fyrir, sem er $\mathcal{O}(\ n\ )$.
- Síðan röðum við aftur.
- Að lokum reiknum við summuna í $\mathcal{O}(\ n\ )$.
- Tímaflækjan er því í heildina $\mathcal{O}(n \log n)$.

Reiknirit

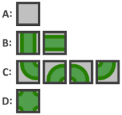
- Sjáum við byrjum á að raða, sem er $\mathcal{O}(n \log n)$.
- Við teljum síðan hvað hver tala kemur oft fyrir, sem er $\mathcal{O}(\ n\ )$.
- Síðan röðum við aftur.
- Að lokum reiknum við summuna í $\mathcal{O}(\ n\ )$.
- Tímaflækjan er því í heildina $\mathcal{O}(n \log n)$.
- Nú er $10^{-8} \cdot 10^6 \cdot \log 10^6 \sim 0,2$, svo þessi lausn er nógu hröð.

Pipe Rotation

- Okkur er gefið $n \times m$ $(1 \leq n, m \leq 100)$ borð af spilum af eftirfarandi gerðum:

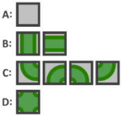
Pipe Rotation

- Okkur er gefið $n \times m$ $(1 \leq n, m \leq 100)$ borð af spilum af eftirfarandi gerðum:



Pipe Rotation

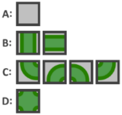
- Okkur er gefið $n \times m$ $(1 \leq n, m \leq 100)$ borð af spilum af eftirfarandi gerðum:



- Við eigum að ákvarða hvort hægt sé að snúa spilunum þannig að allar hliðar sem eru grænar eru gagnstæðar hliðum sem eru einnig grænar.

Pipe Rotation

- Okkur er gefið $n \times m$ $(1 \leq n, m \leq 100)$ borð af spilum af eftirfarandi gerðum:



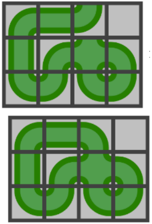
- Við eigum að ákvarða hvort hægt sé að snúa spilunum þannig að allar hliðar sem eru grænar eru gagnstæðar hliðum sem eru einnig grænar.
- Takið eftir að við megum ekki endurraða spilunum og að hliðar sem snúa út mega ekki vera grænar.

Pipe Rotation

- Sjáum dæmi.

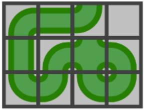
Pipe Rotation

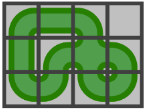
- Sjáum dæmi.



Pipe Rotation

- Sjáum dæmi.

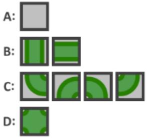




- Efra dæmið er ekki löglegt en neðra er löglegt.

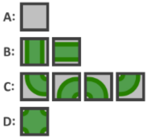
Pipe Rotation

- Sjáum fyrst að ef við vitum hvað er vinstra megin og fyrir ofan spil þá vitum við hvernig það þarf að snúa.



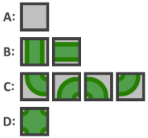
Pipe Rotation

- Sjáum fyrst að ef við vitum hvað er vinstra megin og fyrir ofan spil þá vitum við hvernig það þarf að snúa.
- Ef hliðin fyrir ofan er auð og hliðin til vinstri er auð þá þarf spilið að vera af gerð `A` eða af gerð `C` í öðrum snúning.



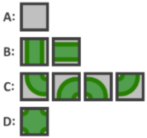
Pipe Rotation

- Sjáum fyrst að ef við vitum hvað er vinstra megin og fyrir ofan spil þá vitum við hvernig það þarf að snúa.
- Ef hliðin fyrir ofan er auð og hliðin til vinstri er auð þá þarf spilið að vera af gerð `A` eða af gerð `C` í öðrum snúning.
- Ef hliðin fyrir ofan er auð og hliðin til vinstri er græn þá þarf spilið að vera af gerð `B` í seinni snúning eða af gerð `C` í þriðja snúning.



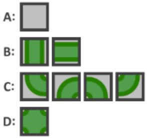
Pipe Rotation

- Sjáum fyrst að ef við vitum hvað er vinstra megin og fyrir ofan spil þá vitum við hvernig það þarf að snúa.
- Ef hliðin fyrir ofan er auð og hliðin til vinstri er auð þá þarf spilið að vera af gerð `A` eða af gerð `C` í öðrum snúning.
- Ef hliðin fyrir ofan er auð og hliðin til vinstri er græn þá þarf spilið að vera af gerð `B` í seinni snúning eða af gerð `C` í þriðja snúning.
- Ef hliðin fyrir ofan er græn og hliðin til vinstri er auð þá þarf spilið að vera af gerð `B` í fyrri snúning eða af gerð `C` í fyrsta snúning.



Pipe Rotation

- Sjáum fyrst að ef við vitum hvað er vinstra megin og fyrir ofan spil þá vitum við hvernig það þarf að snúa.
- Ef hliðin fyrir ofan er auð og hliðin til vinstri er auð þá þarf spilið að vera af gerð `A` eða af gerð `C` í öðrum snúning.
- Ef hliðin fyrir ofan er auð og hliðin til vinstri er græn þá þarf spilið að vera af gerð `B` í seinni snúning eða af gerð `C` í þriðja snúning.
- Ef hliðin fyrir ofan er græn og hliðin til vinstri er auð þá þarf spilið að vera af gerð `B` í fyrri snúning eða af gerð `C` í fyrsta snúning.
- Ef hliðin fyrir ofan er græn og hliðin til vinstri er græn þá þarf spilið að vera af gerð `C` í fjórða snúning eða af gerð `D`.



Pipe Rotation

- Nú getum við labbað í gegnum öll spilin og ákvarðað hvernig hvert spil snýr ef við byrjum á efstu línunni og vinnum okkur niður, og skoðum hverja línu frá vinstri til hægri.

Pipe Rotation

- Nú getum við labbað í gegnum öll spilin og ákvarðað hvernig hvert spil snýr ef við byrjum á efstu línunni og vinnum okkur niður, og skoðum hverja línu frá vinstri til hægri.
- Það er pínu vinna að útfæra þetta útaf öllum tilfellunum sem þarf að hafa í huga.

```c
void imp()
{
    printf("Impossible\n");
    exit(0);
}
```

```c
int main()
{
    int i, j, n, m;
    scanf("%d%d", &n, &m);
    char s[n][m + 1], a[n + 1][m + 1], b[n + 1][m + 1];
    for (i = 0; i < n + 1; i++) for (j = 0; j < m + 1; j++)
        a[i][j] = b[i][j] = 0;
    for (i = 0; i < n; i++) scanf("%s", s[i]);
    for (i = 0; i < n; i++) for (j = 0; j < m; j++)
    {
        if (a[i][j] == 0 && b[i][j] == 0)
        {
            if (s[i][j] == 'C') a[i + 1][j]++, b[i][j + 1]++;
            else if (s[i][j] != 'A') imp();
        }
        else if (a[i][j] == 1 && b[i][j] == 0)
        {
            if (s[i][j] == 'B') a[i + 1][j]++;
            else if (s[i][j] == 'C') b[i][j + 1]++;
            else imp();
        }
        else if (a[i][j] == 0 && b[i][j] == 1)
        {
            if (s[i][j] == 'B') b[i][j + 1]++;
            else if (s[i][j] == 'C') a[i + 1][j]++;
            else imp();
        }
        else
        {
            if (s[i][j] == 'D') a[i + 1][j]++, b[i][j + 1]++;
            else if (s[i][j] != 'C') imp();
        }
    }
    for (i = 0; i < n; i++) if (b[i][m]) imp();
    for (i = 0; i < m; i++) if (a[n][i]) imp();
    printf("Possible\n");
    return 0;
}
```

Pipe Rotation

- Þessi lausn er lítið annað en tvöföld `for`-lykkja sú ytri af lengd n og sú innri af lengd m.

Pipe Rotation

- Þessi lausn er lítið annað en tvöföld `for`-lykkja sú ytri af lengd n og sú innri af lengd m.
- Svo tímaflækjan er $\mathcal{O}($ $)$.

Pipe Rotation

- Þessi lausn er lítið annað en tvöföld `for`-lykkja sú ytri af lengd n og sú innri af lengd m.
- Svo tímaflækjan er $\mathcal{O}(n \cdot m)$.

Language Survey

- Þér er gefið land sem er skipt í grind eftir hvaða tungumál eru töluð hvar.

Language Survey

- Þér er gefið land sem er skipt í grind eftir hvaða tungumál eru töluð hvar.
- Þú veist að í heildina eru þrjú tungumál töluð.

Language Survey

- Þér er gefið land sem er skipt í grind eftir hvaða tungumál eru töluð hvar.
- Þú veist að í heildina eru þrjú tungumál töluð.
- Þér er einnig sagt, fyrir sérhvern reit, hvort eitt tungumál er talað þar eða fleiri.

Language Survey

- Þér er gefið land sem er skipt í grind eftir hvaða tungumál eru töluð hvar.
- Þú veist að í heildina eru þrjú tungumál töluð.
- Þér er einnig sagt, fyrir sérhvern reit, hvort eitt tungumál er talað þar eða fleiri.
- Einnig er gefið að hvert tungumál er talað á samanhangandi svæði reita.

Language Survey

- Þér er gefið land sem er skipt í grind eftir hvaða tungumál eru töluð hvar.
- Þú veist að í heildina eru þrjú tungumál töluð.
- Þér er einnig sagt, fyrir sérhvern reit, hvort eitt tungumál er talað þar eða fleiri.
- Einnig er gefið að hvert tungumál er talað á samanhangandi svæði reita.
- Með öðrum orðum getur þú labbað milli allra reita sem tala sama tungumál án þess að þurfa að heimsækja reit sem talar ekki tungumálið.

Language Survey

- Fyrra sýniinntakið er

Language Survey

- Fyrra sýniinntakið er

```
3 4
2211
1112
1112
```

Language Survey

- Fyrra sýniinntakið er

```
3 4
2211
1112
1112
```

- og samsvarandi úttak er

Language Survey

- Fyrra sýniinntakið er

```
3 4
2211
1112
1112
```

- og samsvarandi úttak er

```
AAAA
...A
....

BB..
BBBB
...B

....
...C
CCCC
```

Language Survey

- Gerum ráð fyrir að við getum raðað tungumálunum þannig að eitt tungumál er talað í hverjum reit, en fyrir hvern reit er líka, að minnsta kosti, einn nágranni sem talar annað tungumál.

Language Survey

- Gerum ráð fyrir að við getum raðað tungumálunum þannig að eitt tungumál er talað í hverjum reit, en fyrir hvern reit er líka, að minnsta kosti, einn nágranni sem talar annað tungumál.
- Ef slík skipun er til þá er eftirleikurinn auðveldur.

Language Survey

- Gerum ráð fyrir að við getum raðað tungumálunum þannig að eitt tungumál er talað í hverjum reit, en fyrir hvern reit er líka, að minnsta kosti, einn nágranni sem talar annað tungumál.
- Ef slík skipun er til þá er eftirleikurinn auðveldur.
- Fyrir hvern reit þar sem fleiri en eitt tungumál er talað veljum við tungumálið sem er talað hjá nágrannanum sem talar annað tungumál og látum það tungumál líka talað í þeim reit.

Language Survey

- Gerum ráð fyrir að við getum raðað tungumálunum þannig að eitt tungumál er talað í hverjum reit, en fyrir hvern reit er líka, að minnsta kosti, einn nágranni sem talar annað tungumál.
- Ef slík skipun er til þá er eftirleikurinn auðveldur.
- Fyrir hvern reit þar sem fleiri en eitt tungumál er talað veljum við tungumálið sem er talað hjá nágrannanum sem talar annað tungumál og látum það tungumál líka talað í þeim reit.
- En er slík skipun til?

Language Survey

- Byrjum á að láta dálkinn lengst til vinstri fá tungumál `A`.

Language Survey

- Byrjum á að láta dálkinn lengst til vinstri fá tungumál `A`.
- Við látum svo `p` tákna reitinn efst til hægri og `q` tákna reitinn neðst til hægri og:

Language Survey

- Byrjum á að láta dálkinn lengst til vinstri fá tungumál A.
- Við látum svo p tákna reitinn efst til hægri og q tákna reitinn neðst til hægri og:
  - Færum p til vintri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál B.

Language Survey

- Byrjum á að láta dálkinn lengst til vinstri fá tungumál A.
- Við látum svo p tákna reitinn efst til hægri og q tákna reitinn neðst til hægri og:
    - Færum p til vintri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál B.
    - Færum p niður þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál B.

Language Survey

- Byrjum á að láta dálkinn lengst til vinstri fá tungumál $\mathtt{A}$.
- Við látum svo $\mathtt{p}$ tákna reitinn efst til hægri og $\mathtt{q}$ tákna reitinn neðst til hægri og:
  - Færum $\mathtt{p}$ til vintri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál $\mathtt{B}$.
  - Færum $\mathtt{p}$ niður þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál $\mathtt{B}$.
  - Færum $\mathtt{q}$ upp þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál $\mathtt{C}$.

Language Survey

- Byrjum á að láta dálkinn lengst til vinstri fá tungumál $\mathtt{A}$.
- Við látum svo p tákna reitinn efst til hægri og q tákna reitinn neðst til hægri og:
  - Færum p til vintri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál $\mathtt{B}$.
  - Færum p niður þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál $\mathtt{B}$.
  - Færum q upp þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál $\mathtt{C}$.
  - Færum q til vinstri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál $\mathtt{C}$.

Language Survey

- Byrjum á að láta dálkinn lengst til vinstri fá tungumál A.
- Við látum svo p tákna reitinn efst til hægri og q tákna reitinn neðst til hægri og:
  - Færum p til vintri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál B.
  - Færum p niður þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál B.
  - Færum q upp þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál C.
  - Færum q til vinstri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál C.
  - Færum p til hægri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál B.

Language Survey

- Byrjum á að láta dálkinn lengst til vinstri fá tungumál A.
- Við látum svo p tákna reitinn efst til hægri og q tákna reitinn neðst til hægri og:
  - Færum p til vintri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál B.
  - Færum p niður þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál B.
  - Færum q upp þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál C.
  - Færum q til vinstri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál C.
  - Færum p til hægri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál B.
  - Færum p upp þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál B.

Language Survey

- Byrjum á að láta dálkinn lengst til vinstri fá tungumál A.
- Við látum svo p tákna reitinn efst til hægri og q tákna reitinn neðst til hægri og:
  - Færum p til vintri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál B.
  - Færum p niður þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál B.
  - Færum q upp þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál C.
  - Færum q til vinstri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál C.
  - Færum p til hægri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál B.
  - Færum p upp þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál B.
  - Færum q niður þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál C.

Language Survey

- Byrjum á að láta dálkinn lengst til vinstri fá tungumál `A`.
- Við látum svo p tákna reitinn efst til hægri og q tákna reitinn neðst til hægri og:
  - Færum p til vintri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál `B`.
  - Færum p niður þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál `B`.
  - Færum q upp þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál `C`.
  - Færum q til vinstri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál `C`.
  - Færum p til hægri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál `B`.
  - Færum p upp þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál `B`.
  - Færum q niður þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál `C`.
  - Færum q til hægri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál `C`.

Language Survey

- Byrjum á að láta dálkinn lengst til vinstri fá tungumál A.
- Við látum svo p tákna reitinn efst til hægri og q tákna reitinn neðst til hægri og:
  - Færum p til vintri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál B.
  - Færum p niður þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál B.
  - Færum q upp þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál C.
  - Færum q til vinstri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál C.
  - Færum p til hægri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál B.
  - Færum p upp þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál B.
  - Færum q niður þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál C.
  - Færum q til hægri þangað til hann kemst ekki lengra og látum alla reiti sem hann lendir á fá tungumál C.
  - Endurtökum þar til allir reitir hafa tungumál.

```
x x x x x x x x x
x x x x x x x x x
x x x x x x x x x
x x x x x x x x x
x x x x x x x x x
x x x x x x x x x
x x x x x x x x x
x x x x x x x x x
x x x x x x x x x
```

A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x

```
A x x x x x x x B
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
```

```
A x x x x x x B B
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
```

```
A x x x x x B B B
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
```

```
A x x x x B B B B
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
```

A	x	x	x	B	B	B	B	B
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x

A	x	x	B	B	B	B	B	B
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x

```
A x B B B B B B B
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
```

A	B	B	B	B	B	B	B	B
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x

```
A B B B B B B B B
A B x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
```

```
A B B B B B B B B
A B x x x x x x x
A B x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
```

```
A B B B B B B B B
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
```

```
A B B B B B B B B
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
A x x x x x x x x
```

A	B	B	B	B	B	B	B	B
A	B	x	x	x	x	x	x	x
A	B	x	x	x	x	x	x	x
A	B	x	x	x	x	x	x	x
A	B	x	x	x	x	x	x	x
A	B	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x
A	x	x	x	x	x	x	x	x

```
A B B B B B B B B
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A x x x x x x x x
A x x x x x x x x
```

```
A B B B B B B B B
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A x x x x x x x x
```

```
A B B B B B B B B
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
```

```
A B B B B B B B B
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x C
```

```
A B B B B B B B B
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x C
A B x x x x x x C
```

```
A B B B B B B B B
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
```

```
A B B B B B B B B
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
```

```
A B B B B B B B B
A B x x x x x x x
A B x x x x x x x
A B x x x x x x x
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
```

```
A B B B B B B B B
A B x x x x x x x
A B x x x x x x x
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
```

```
A B B B B B B B B
A B x x x x x x x
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
```

```
A B B B B B B B B
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
```

```
A B B B B B B B B
A B x x x x x C C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
```

```
A B B B B B B B B
A B x x x x C C C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
```

```
A B B B B B B B B
A B x x x C C C C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
```

```
A B B B B B B B B
A B x x C C C C C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
```

```
A B B B B B B B B
A B x C C C C C C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
```

```
A B B B B B B B B
A B C C C C C C C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
```

A	B	B	B	B	B	B	B	B
A	B	C	C	C	C	C	C	C
A	B	x	x	x	x	x	x	C
A	B	x	x	x	x	x	x	C
A	B	x	x	x	x	x	x	C
A	B	x	x	x	x	x	x	C
A	B	x	x	x	x	x	x	C
A	B	x	x	x	x	x	x	C
A	B	B	x	x	x	x	x	C

```
A B B B B B B B B
A B C C C C C C C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B B B x x x x C
```

```
A B B B B B B B B
A B C C C C C C C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B B B B x x x C
```

```
A B B B B B B B B
A B C C C C C C C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B B B B B x x C
```

```
A B B B B B B B B
A B C C C C C C C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B B B B B B x C
```

```
A B B B B B B B B
A B C C C C C C C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x B C
A B x x x x x B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B x x x x x x C
A B x x x x x x C
A B x x x x x x C
A B x x x x x B C
A B x x x x x B C
A B x x x x x B C
A B B B B B B B C
```

A	B	B	B	B	B	B	B	B
A	B	C	C	C	C	C	C	C
A	B	x	x	x	x	x	x	C
A	B	x	x	x	x	x	x	C
A	B	x	x	x	x	x	B	C
A	B	x	x	x	x	x	B	C
A	B	x	x	x	x	x	B	C
A	B	x	x	x	x	x	B	C
A	B	B	B	B	B	B	B	C

```
A B B B B B B B B
A B C C C C C C C
A B x x x x x x C
A B x x x x x B C
A B x x x x x B C
A B x x x x x B C
A B x x x x x B C
A B x x x x x B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B x x x x x B C
A B x x x x x B C
A B x x x x x B C
A B x x x x x B C
A B x x x x x B C
A B x x x x x B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B C x x x x B C
A B x x x x x B C
A B x x x x x B C
A B x x x x x B C
A B x x x x x B C
A B x x x x x B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B C x x x x B C
A B C x x x x B C
A B x x x x x B C
A B x x x x x B C
A B x x x x x B C
A B x x x x x B C
A B B B B B B B C
```

A	B	B	B	B	B	B	B	B
A	B	C	C	C	C	C	C	C
A	B	C	x	x	x	x	B	C
A	B	C	x	x	x	x	B	C
A	B	C	x	x	x	x	B	C
A	B	x	x	x	x	x	B	C
A	B	x	x	x	x	x	B	C
A	B	x	x	x	x	x	B	C
A	B	B	B	B	B	B	B	C

```
A B B B B B B B B
A B C C C C C C C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B x x x x x B C
A B x x x x x B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B x x x x x B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C C x x x B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C C C x x B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C C C C x B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C C C C C B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B C x x x B B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C C C C C B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B C x x B B B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C C C C C B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B C x B B B B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C C C C C B C
A B B B B B B B C
```

A	B	B	B	B	B	B	B	B
A	B	C	C	C	C	C	C	C
A	B	C	B	B	B	B	B	C
A	B	C	x	x	x	x	B	C
A	B	C	x	x	x	x	B	C
A	B	C	x	x	x	x	B	C
A	B	C	x	x	x	x	B	C
A	B	C	C	C	C	C	B	C
A	B	B	B	B	B	B	B	C

```
A B B B B B B B B
A B C C C C C C C
A B C B B B B B C
A B C B x x x B C
A B C x x x x B C
A B C x x x x B C
A B C x x x x B C
A B C C C C C B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B C B B B B B C
A B C B x x x B C
A B C B x x x B C
A B C x x x x B C
A B C x x x x B C
A B C C C C C B C
A B B B B B B B C
```

A	B	B	B	B	B	B	B	B
A	B	C	C	C	C	C	C	C
A	B	C	B	B	B	B	B	C
A	B	C	B	x	x	x	B	C
A	B	C	B	x	x	x	B	C
A	B	C	B	x	x	x	B	C
A	B	C	x	x	x	x	B	C
A	B	C	C	C	C	C	B	C
A	B	B	B	B	B	B	B	C

```
A B B B B B B B B
A B C C C C C C C
A B C B B B B B C
A B C B x x x B C
A B C B x x x B C
A B C B x x x B C
A B C B x x x B C
A B C C C C C B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B C B B B B B C
A B C B x x x B C
A B C B x x x B C
A B C B x x x B C
A B C B x x C B C
A B C C C C C B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B C B B B B B C
A B C B x x x B C
A B C B x x x B C
A B C B x x C B C
A B C B x x C B C
A B C C C C C B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B C B B B B B C
A B C B x x x B C
A B C B x x C B C
A B C B x x C B C
A B C B x x C B C
A B C C C C C B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B C B B B B B C
A B C B x x C B C
A B C B x x C B C
A B C B x x C B C
A B C B x x C B C
A B C C C C C B C
A B B B B B B B C
```

A	B	B	B	B	B	B	B	B
A	B	C	C	C	C	C	C	C
A	B	C	B	B	B	B	B	C
A	B	C	B	x	C	C	B	C
A	B	C	B	x	x	C	B	C
A	B	C	B	x	x	C	B	C
A	B	C	B	x	x	C	B	C
A	B	C	C	C	C	C	B	C
A	B	B	B	B	B	B	B	C

A	B	B	B	B	B	B	B	B
A	B	C	C	C	C	C	C	C
A	B	C	B	B	B	B	B	C
A	B	C	B	C	C	C	B	C
A	B	C	B	x	x	C	B	C
A	B	C	B	x	x	C	B	C
A	B	C	B	x	x	C	B	C
A	B	C	C	C	C	C	B	C
A	B	B	B	B	B	B	B	C

```
A B B B B B B B B
A B C C C C C C C
A B C B B B B B C
A B C B C C C B C
A B C B x x C B C
A B C B x x C B C
A B C B B x C B C
A B C C C C C B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B C B B B B B C
A B C B C C C B C
A B C B x x C B C
A B C B x x C B C
A B C B B B C B C
A B C C C C C B C
A B B B B B B B C
```

```
A B B B B B B B B
A B C C C C C C C
A B C B B B B B C
A B C B C C C B C
A B C B x x C B C
A B C B x B C B C
A B C B B B C B C
A B C C C C C B C
A B B B B B B B C
```

A	B	B	B	B	B	B	B	B
A	B	C	C	C	C	C	C	C
A	B	C	B	B	B	B	B	C
A	B	C	B	C	C	C	B	C
A	B	C	B	x	B	C	B	C
A	B	C	B	x	B	C	B	C
A	B	C	B	B	B	C	B	C
A	B	C	C	C	C	C	B	C
A	B	B	B	B	B	B	B	C

A	B	B	B	B	B	B	B	B
A	B	C	C	C	C	C	C	C
A	B	C	B	B	B	B	B	C
A	B	C	B	C	C	C	B	C
A	B	C	B	C	B	C	B	C
A	B	C	B	x	B	C	B	C
A	B	C	B	B	B	C	B	C
A	B	C	C	C	C	C	B	C
A	B	B	B	B	B	B	B	C

A	B	B	B	B	B	B	B	B
A	B	C	C	C	C	C	C	C
A	B	C	B	B	B	B	B	C
A	B	C	B	C	C	C	B	C
A	B	C	B	C	B	C	B	C
A	B	C	B	C	B	C	B	C
A	B	C	B	B	B	C	B	C
A	B	C	C	C	C	C	B	C
A	B	B	B	B	B	B	B	C

Language Survey

- Þetta virkar alltaf ef önnur víddin er stærri en tveir.

Language Survey

- Þetta virkar alltaf ef önnur víddin er stærri en tveir.
- Ef önnur víddin er tveir má nota

Language Survey

- Þetta virkar alltaf ef önnur víddin er stærri en tveir.
- Ef önnur víddin er tveir má nota

```
A B
A B
A B
A B
A C
```

Language Survey

- Þetta virkar alltaf ef önnur víddin er stærri en tveir.
- Ef önnur víddin er tveir má nota

```
A B
A B
A B
A B
A C
```

- Ef önnur víddin er einn þarf að leysa dæmið gráðugt.

Language Survey

- Þetta virkar alltaf ef önnur víddin er stærri en tveir.
- Ef önnur víddin er tveir má nota

```
A B
A B
A B
A B
A C
```

- Ef önnur víddin er einn þarf að leysa dæmið gráðugt.
- Þetta eru einu tilfellin þar sem maður getur fengið *impossible*.

Sértilfellið

```c
int foo(int* a, int n)
{
    int i, j, b[3][n], c[3] = {0, 0, 0};
    rep(i, n) rep(j, 3) b[j][i] = 0;
    i = 0;
    while (i < n && a[i] == 1) b[0][i++] = 1, c[0]++;
    while (i < n && a[i] == 2) b[0][i] = 1, b[1][i] = 1, i++, c[0]++, c[1]++;
    while (i < n && a[i] == 1) b[1][i++] = 1, c[1]++;
    while (i < n && a[i] == 2) b[1][i] = 1, b[2][i] = 1, i++, c[1]++, c[2]++;
    while (i < n && a[i] == 1) b[2][i++] = 1, c[2]++;
    if (c[2] == 0 && c[1] == 0)
    {
        if (n == 2) return 0;
        b[1][0] = b[2][1] = 1;
        b[0][0] = b[0][1] = 0;
    }
    else if (c[2] == 0)
    {
        rep(i, n) if (a[i] == 2) break;
        b[2][i] = 1;
    }
    rep(i, n) if (b[0][i] == 0 && b[1][i] == 0 && b[2][i] == 0) return 0;
    rep(j, 3) rep(i, n) r[j][i] = b[j][i] ? 'A' + j : '.';
    return 1;
}
```

Köllum á sértilfellið

```
    if (n == 1 && m == 1)
    {
        if (a[0][0] == 1) { printf("impossible\n"); return 0; }
        printf("A\n\nB\n\nC\n");
        return 0;
    }
    if (n == 1)
    {
        int g[m];
        rep(i, m) g[i] = a[0][i];
        if (!foo(g, m)) { printf("impossible\n"); return 0; }
        rep(x, 3) { rep(i, m) printf("%c", r[x][i]); printf("\n\n"); }
        return 0;
    }
    if (m == 1)
    {
        int g[n];
        rep(i, n) g[i] = a[i][0];
        if (!foo(g, n)) { printf("impossible\n"); return 0; }
        rep(x, 3) { rep(i, n) printf("%c\n", r[x][i]); printf("\n"); }
        return 0;
    }
```

Grunnskiptingin

```
    z = n - 1, w = m - 3, x = n, y = m - 1;
    while (1)
    {
        if (z <= 0) break;
        rep(i, z) b[--x][y] = 0;
        z -= 2;
        if (w <= 0) break;
        rep(i, w) b[x][--y] = 0;
        w -= 2;
        if (z <= 0) break;
        rep(i, z) b[++x][y] = 0;
        z -= 2;
        if (w <= 0) break;
        rep(i, w) b[x][++y] = 0;
        w -= 2;
    }
```

Finna nágrannann með annað tungamál

```
    rep(i, n) rep(j, m) c[i][j] = -1;
    rep(i, n) rep(j, m)
    {
        if (i > 0 && b[i - 1][j] != b[i][j]) c[i][j] = b[i - 1][j];
        if (j > 0 && b[i][j - 1] != b[i][j]) c[i][j] = b[i][j - 1];
        if (i < n - 1 && b[i + 1][j] != b[i][j]) c[i][j] = b[i + 1][j];
        if (j < m - 1 && b[i][j + 1] != b[i][j]) c[i][j] = b[i][j + 1];
    }
```

Prenta lausn

```
rep(x, 3)
{
    rep(i, n)
    {
        rep(j, m) printf("%c",
                (b[i][j] == x || (a[i][j] == 2 && c[i][j] == x))
                ? 'A' + x : '.');
        printf("\n");
    }
    printf("\n");
}
```