

Ad hoc

Bergur Snorrason

17. janúar 2022

Lausnar aðferðir

- ▶ Þegar við leysum dæmi í keppnisforritun notumst við oftast við eina af eftirfarandi aðferðum:

Lausnar aðferðir

- ▶ Þegar við leysum dæmi í keppnisforritun notumst við oftast við eina af eftirfarandi aðferðum:
 - ▶ *Ad hoc*,

Lausnar aðferðir

- ▶ Þegar við leysum dæmi í keppnisforritun notumst við oftast við eina af eftirfarandi aðferðum:
 - ▶ *Ad hoc*,
 - ▶ *Tæmandi leit eða ofbeldis aðferðin* (e. *complete search, brute force*),

Lausnar aðferðir

- ▶ Þegar við leysum dæmi í keppnisforritun notumst við oftast við eina af eftirfarandi aðferðum:
 - ▶ *Ad hoc*,
 - ▶ *Tæmandi leit eða ofbeldis aðferðin* (e. *complete search, brute force*),
 - ▶ *Gráðug reiknirit* (e. *greedy algorithms*),

Lausnar aðferðir

- ▶ Þegar við leysum dæmi í keppnisforritun notumst við oftast við eina af eftirfarandi aðferðum:
 - ▶ *Ad hoc*,
 - ▶ *Tæmandi leit eða ofbeldis aðferðin* (e. *complete search, brute force*),
 - ▶ *Gráðug reiknirit* (e. *greedy algorithms*),
 - ▶ *Deila og drottna* (e. *divide and conquer*),

Lausnar aðferðir

- ▶ Þegar við leysum dæmi í keppnisforritun notumst við oftast við eina af eftirfarandi aðferðum:
 - ▶ *Ad hoc*,
 - ▶ *Tæmandi leit eða ofbeldis aðferðin* (e. *complete search, brute force*),
 - ▶ *Gráðug reiknirit* (e. *greedy algorithms*),
 - ▶ *Deila og drottna* (e. *divide and conquer*),
 - ▶ *Kvik bestun* (e. *dynamic programming*).

Lausnar aðferðir

- ▶ Þegar við leysum dæmi í keppnisforritun notumst við oftast við eina af eftirfarandi aðferðum:
 - ▶ *Ad hoc*,
 - ▶ *Tæmandi leit eða ofbeldis aðferðin* (e. *complete search, brute force*),
 - ▶ *Gráðug reiknirit* (e. *greedy algorithms*),
 - ▶ *Deila og drottna* (e. *divide and conquer*),
 - ▶ *Kvik bestun* (e. *dynamic programming*).
- ▶ Þessi skipting er ekki fullkomin, en hún hjálpar okkur að ræða dæmi.

Lausnar aðferðir

- ▶ Þegar við leysum dæmi í keppnisforritun notumst við oftast við eina af eftirfarandi aðferðum:
 - ▶ *Ad hoc*,
 - ▶ *Tæmandi leit* eða *ofbeldis aðferðin* (e. *complete search*, *brute force*),
 - ▶ *Gráðug reiknirit* (e. *greedy algorithms*),
 - ▶ *Deila og drottna* (e. *divide and conquer*),
 - ▶ *Kvik bestun* (e. *dynamic programming*).
- ▶ Þessi skipting er ekki fullkomin, en hún hjálpar okkur að ræða dæmi.
- ▶ Við munum byrja á því að fjalla almennt um þessar aðferðir og fara svo í sértækara efni.

Lausnar aðferðir

- ▶ Þegar við leysum dæmi í keppnisforritun notumst við oftast við eina af eftirfarandi aðferðum:
 - ▶ *Ad hoc*,
 - ▶ *Tæmandi leit* eða *ofbeldis aðferðin* (e. *complete search*, *brute force*),
 - ▶ *Gráðug reiknirit* (e. *greedy algorithms*),
 - ▶ *Deila og drottna* (e. *divide and conquer*),
 - ▶ *Kvik bestun* (e. *dynamic programming*).
- ▶ Þessi skipting er ekki fullkomin, en hún hjálpar okkur að ræða dæmi.
- ▶ Við munum byrja á því að fjalla almennt um þessar aðferðir og fara svo í sértækara efni.
- ▶ Þá er oft gott að hafa í huga hvernig flokka megi reikniritin.

Ad hoc

- ▶ Ef lausn dæmisins byggir ekki á sérþekkingu flokkast dæmið sem *Ad hoc*.

Ad hoc

- ▶ Ef lausn dæmisins byggir ekki á sérþekkingu flokkast dæmið sem *Ad hoc*.
- ▶ Þessi dæmi eru stundum flokkuð undir *útfærsludæmi* (e. *implementation*).

Ad hoc

- ▶ Ef lausn dæmisins byggir ekki á sérþekkingu flokkast dæmið sem *Ad hoc*.
- ▶ Þessi dæmi eru stundum flokkuð undir *útfærsludæmi* (e. *implementation*).
- ▶ Þetta er gert því flest Ad hoc dæmi snúast um að fylgja beint leiðbeiningum.

Ad hoc

- ▶ Ef lausn dæmisins byggir ekki á sérþekkingu flokkast dæmið sem *Ad hoc*.
- ▶ Þessi dæmi eru stundum flokkuð undir *útfærsludæmi* (e. *implementation*).
- ▶ Þetta er gert því flest Ad hoc dæmi snúast um að fylgja beint leiðbeiningum.
- ▶ Það eru þó undantekningar.

Ad hoc

- ▶ Ef lausn dæmisins byggir ekki á sérþekkingu flokkast dæmið sem *Ad hoc*.
- ▶ Þessi dæmi eru stundum flokkuð undir *útfærsludæmi* (e. *implementation*).
- ▶ Þetta er gert því flest Ad hoc dæmi snúast um að fylgja beint leiðbeiningum.
- ▶ Það eru þó undantekningar.
- ▶ Í NCPC 2020 var Ad hoc dæmi sem mætti ekki flokkast sem útfærsludæmi.

Ad hoc

- ▶ Ef lausn dæmisins byggir ekki á sérþekkingu flokkast dæmið sem *Ad hoc*.
- ▶ Þessi dæmi eru stundum flokkuð undir *útfærsludæmi* (e. *implementation*).
- ▶ Þetta er gert því flest Ad hoc dæmi snúast um að fylgja beint leiðbeiningum.
- ▶ Það eru þó undantekningar.
- ▶ Í NCPC 2020 var Ad hoc dæmi sem mætti ekki flokkast sem útfærsludæmi.
- ▶ Léttari dæmin í keppnum eru oft Ad hoc dæmi.

Ad hoc

- ▶ Ef lausn dæmisins byggir ekki á sérþekkingu flokkast dæmið sem *Ad hoc*.
- ▶ Þessi dæmi eru stundum flokkuð undir *útfærsludæmi* (e. *implementation*).
- ▶ Þetta er gert því flest Ad hoc dæmi snúast um að fylgja beint leiðbeiningum.
- ▶ Það eru þó undantekningar.
- ▶ Í NCPC 2020 var Ad hoc dæmi sem mætti ekki flokkast sem útfærsludæmi.
- ▶ Léttari dæmin í keppnum eru oft Ad hoc dæmi.
- ▶ Áður nefnt NCPC dæmi er þó aftur undanteking, því engin keppandi náði að leysa það í keppninni.

Ad hoc

- ▶ Ef lausn dæmisins byggir ekki á sérþekkingu flokkast dæmið sem *Ad hoc*.
- ▶ Þessi dæmi eru stundum flokkuð undir *útfærsludæmi* (e. *implementation*).
- ▶ Þetta er gert því flest Ad hoc dæmi snúast um að fylgja beint leiðbeiningum.
- ▶ Það eru þó undantekningar.
- ▶ Í NCPC 2020 var Ad hoc dæmi sem mætti ekki flokkast sem útfærsludæmi.
- ▶ Léttari dæmin í keppnum eru oft Ad hoc dæmi.
- ▶ Áður nefnt NCPC dæmi er þó aftur undanteking, því engin keppandi náði að leysa það í keppninni.
- ▶ Samkvæmt skilgreiningu getum við ekki rætt Ad hoc dæmi ítarlega. Tökum því nokkur dæmi.

Blandað brot

- ▶ Þú átt að breyta almennu broti í blandað brot.

Blandað brot

- ▶ Þú átt að breyta almennu broti í blandað brot.
- ▶ Munið að almenna brotið p/q , og blandaða brotið $a + b/c$ tákna sömu töluna ef $p/q = a + b/c$.

Blandað brot

- ▶ Þú átt að breyta almennu broti í blandað brot.
- ▶ Munið að almenna brotið p/q , og blandaða brotið $a\ b/c$ tákna sömu töluna ef $p/q = a + b/c$.
- ▶ Munið einnig að ef $a\ b/c$ er almennt brot þá gildir $b < c$.

Blandað brot

- ▶ Þú átt að breyta almennu broti í blandað brot.
- ▶ Munið að almenna brotið p/q , og blandaða brotið $a\ b/c$ tákna sömu töluna ef $p/q = a + b/c$.
- ▶ Munið einnig að ef $a\ b/c$ er almennt brot þá gildir $b < c$.
- ▶ Blandaða brotið ykkar á að hafa sama nefnara og upprunarlega brotið.

Blandað brot

- ▶ Þú átt að breyta almennu broti í blandað brot.
- ▶ Munið að almenna brotið p/q , og blandaða brotið $a\ b/c$ tákna sömu töluna ef $p/q = a + b/c$.
- ▶ Munið einnig að ef $a\ b/c$ er almennt brot þá gildir $b < c$.
- ▶ Blandaða brotið ykkar á að hafa sama nefnara og upprunarlega brotið.
- ▶ Inntakið inniheldur tvær heiltölur $1 \leq p, q \leq 10^9$.

Blandað brot

- ▶ Þú átt að breyta almennu broti í blandað brot.
- ▶ Munið að almenna brotið p/q , og blandaða brotið $a\ b/c$ tákna sömu töluna ef $p/q = a + b/c$.
- ▶ Munið einnig að ef $a\ b/c$ er almennt brot þá gildir $b < c$.
- ▶ Blandaða brotið ykkar á að hafa sama nefnara og upprunarlega brotið.
- ▶ Inntakið inniheldur tvær heiltölur $1 \leq p, q \leq 10^9$.
- ▶ Úttakið skal innihalda blandaða brotið sem svarar til p/q .

Blandað brot

- ▶ Þú átt að breyta almennu broti í blandað brot.
- ▶ Munið að almenna brotið p/q , og blandaða brotið $a\ b/c$ tákna sömu töluna ef $p/q = a + b/c$.
- ▶ Munið einnig að ef $a\ b/c$ er almennt brot þá gildir $b < c$.
- ▶ Blandaða brotið ykkar á að hafa sama nefnara og upprunarlega brotið.
- ▶ Inntakið inniheldur tvær heiltölur $1 \leq p, q \leq 10^9$.
- ▶ Úttakið skal innihalda blandaða brotið sem svarar til p/q .

	Inntak	Úttak
Sýnidæmi 1	27 12	2 3 / 12
Sýnidæmi 2	2460000 98400	25 0 / 98400
Sýnidæmi 3	3 4000	0 3 / 4000

Lausn á blandað brot

- ▶ Hér nægir okkur að reikna.

Lausn á blandað brot

- ▶ Hér nægir okkur að reikna.
- ▶ Við getum aðeins stytt okkur leið með því að nota heiltöludeilingu.

Lausn á blandað brot

- ▶ Hér nægir okkur að reikna.
- ▶ Við getum aðeins stytt okkur leið með því að nota heiltöludeilingu.
- ▶ Við fáum þá að a er heiltalan sem fæst með deilingunni p/q og b er afgangurinn.

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int p, q, a, b, c;
6     scanf("%d%d", &p, &q);
7     a = p/q;
8     b = p%q;
9     c = q;
10    printf("%d %d / %d\n", a, b, c);
11    return 0;
12 }
```

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int p, q, a, b, c;
6     scanf("%d%d", &p, &q);
7     a = p/q;
8     b = p%q;
9     c = q;
10    printf("%d %d / %d\n", a, b, c);
11    return 0;
12 }
```

```
1 #include <stdio.h>
2
3 int main()
4 {
5     int p, q;
6     scanf("%d%d", &p, &q);
7     printf("%d %d / %d\n", p/q, p%q, q);
8     return 0;
9 }
```

Barnahjal

- ▶ Pið eruð að reyna að kenna barni að telja.

Barnahjal

- ▶ Þið eruð að reyna að kenna barni að telja.
- ▶ Það er þó ekki alltaf hægt að heyra hvað barnið segir.

Barnahjal

- ▶ Þið eruð að reyna að kenna barni að telja.
- ▶ Það er þó ekki alltaf hægt að heyra hvað barnið segir.
- ▶ Þið viljið ákvarða hvort það sem barnið er að segja gæti mögulega verið rétt.

Barnahjal

- ▶ Þið eruð að reyna að kenna barni að telja.
- ▶ Það er þó ekki alltaf hægt að heyra hvað barnið segir.
- ▶ Þið viljið ákvarða hvort það sem barnið er að segja gæti mögulega verið rétt.
- ▶ Fyrsta lína inntaksins inniheldur heiltölu $1 \leq n \leq 10^3$.

Barnahjal

- ▶ Þið eruð að reyna að kenna barni að telja.
- ▶ Það er þó ekki alltaf hægt að heyra hvað barnið segir.
- ▶ Þið viljið ákvarða hvort það sem barnið er að segja gæti mögulega verið rétt.
- ▶ Fyrsta lína inntaksins inniheldur heiltölu $1 \leq n \leq 10^3$.
- ▶ Síðan fylgir ein lína með n strengjum.

Barnahjal

- ▶ Þið eruð að reyna að kenna barni að telja.
- ▶ Það er þó ekki alltaf hægt að heyra hvað barnið segir.
- ▶ Þið viljið ákvarða hvort það sem barnið er að segja gæti mögulega verið rétt.
- ▶ Fyrsta lína inntaksins inniheldur heiltölu $1 \leq n \leq 10^3$.
- ▶ Síðan fylgir ein lína með n strengjum.
- ▶ Hver strengur er annaðhvort heiltala á bilinu $[0, 10^4]$ eða strengurinn “mumble”.

Barnahjal

- ▶ Þið eruð að reyna að kenna barni að telja.
- ▶ Það er þó ekki alltaf hægt að heyra hvað barnið segir.
- ▶ Þið viljið ákvarða hvort það sem barnið er að segja gæti mögulega verið rétt.
- ▶ Fyrsta lína inntaksins inniheldur heiltölu $1 \leq n \leq 10^3$.
- ▶ Síðan fylgir ein lína með n strengjum.
- ▶ Hver strengur er annaðhvort heiltala á bilinu $[0, 10^4]$ eða strengurinn “mumble”.
- ▶ Ef það er hægt að skipta út öllum “mumble” fyrir tölu þannig að talningin sé rétt skal prenta “jebb”.

Barnahjal

- ▶ Þið eruð að reyna að kenna barni að telja.
- ▶ Það er þó ekki alltaf hægt að heyra hvað barnið segir.
- ▶ Þið viljið ákvarða hvort það sem barnið er að segja gæti mögulega verið rétt.
- ▶ Fyrsta lína inntaksins inniheldur heiltölu $1 \leq n \leq 10^3$.
- ▶ Síðan fylgir ein lína með n strengjum.
- ▶ Hver strengur er annaðhvort heiltala á bilinu $[0, 10^4]$ eða strengurinn “mumble”.
- ▶ Ef það er hægt að skipta út öllum “mumble” fyrir tölu þannig að talningin sé rétt skal prenta “jebb”.
- ▶ Annars skal prenta “neibb”.

Lausn á Barnahjal

- ▶ Ef i -ti strengurinn inniheldur strenginn sem svarar til tölunnar i eða “mumble”, fyrir öll i , þá er barnið kannski að telja rétt.

Lausn á Barnahjal

- ▶ Ef i -ti strengurinn inniheldur strenginn sem svarar til tölunnar i eða “mumble”, fyrir öll i , þá er barnið kannski að telja rétt.
- ▶ Annars er barnið að telja rangt.

Útfærsla í Python

```
1 n = int(input())
2 l = input().split()
3 f = True
4 for i in range(n):
5     if l[i] != 'mumble' and l[i] != str(i + 1):
6         f = False
7         break
8 if f: print('jebb')
9 else: print('neibb')
```

