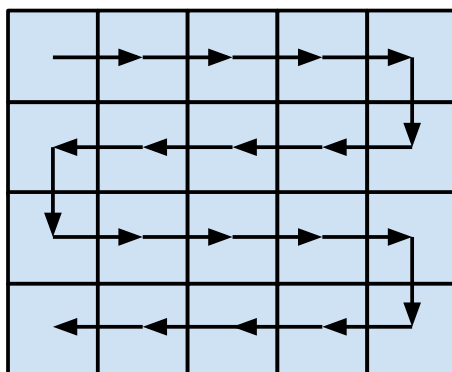


# Oppgave 1: Brodering

NIO 2013/2014 — 2. runde

En broderingsmaskin kan programmeres til å sy sting med forskjellige fargede tråder for å lage fine mønstre. Den lager rektangulære broderier bestående av  $M$  rader med  $N$  sting hver. Maskinen begynner med å brodere første rad fra venstre til høyre, deretter syr den neste rad fra høyre til venstre, deretter tredje rad fra venstre til høyre igjen, osv. (se bildet)



I denne oppgaven bruker vi store bokstaver fra A til Z for å representere de forskjellige fargene.

En oppskrift til broderingsmaskinen begynner med en linje med tre tall:

$K$  - antall kolonner i broderiet

$R$  - antall rader i broderiet

$N$  - antall instruksjoner i programmet

Deretter følger  $N$  linjer med én instruksjon til broderingsmaskinen hver. Hver linje er på formatet  $A F$ , der  $A$  er et heltall mellom 1 og 1000 og  $F$  er en bokstav i det engelske alfabetet (A til Z) som representerer en farge. Denne instruksjonen sier at de neste  $A$  stingene i broderiet skal ha farge  $F$ .

Summen av alle  $A$ 'ene i oppskriften vil alltid være nøyaktig lik  $K \times R$ . Du kan altså være sikker på at oppskriften vil inneholde akkurat så mange sting som trengs for å utføre broderiet.

Din oppgave er å lage et program som kan brukes til å vise hvordan et broderi vil bli sendes ut. Programmet ditt skal lese inn en oppskrift på et broderi og skrive ut et bilde av hvordan broderiet ville blitt.

## Input

En gyldig oppskrift til broderingsmaskinen, slik beskrevet ovenfor.

$$1 \leq K \leq 1000$$

$$1 \leq R \leq 1000$$

$$1 \leq N \leq 100000$$

## Output

Programmet ditt skal skrive ut  $R$  rader med  $K$  bokstaver hver. Disse skal lage et bilde av hvordan broderiet ville blitt sendes ut.

### Eksempel 1

#### Input

```
5 5 15
1 X
3 0
1 X
1 0
3 X
2 0
1 X
1 V
1 X
2 0
3 X
1 0
1 X
3 0
1 X
```

#### Output

```
X000X
OXXX0
OXVX0
OXXX0
X000X
```

### Eksempel 2

#### Input

```
14 9 40
1 I
3 R
1 H
1 B
1 H
14 R
1 H
1 B
1 H
3 R
```

2 I  
3 R  
1 H  
1 B  
1 H  
7 R  
8 H  
1 B  
4 H  
2 I  
13 B  
8 H  
1 B  
4 H  
2 I  
3 R  
1 H  
1 B  
1 H  
14 R  
1 H  
1 B  
1 H  
3 R  
2 I  
3 R  
1 H  
1 B  
1 H  
7 R

## Output

IRRRHBHRRRRRRR  
IRRRHBHRRRRRRR  
IRRRHBHRRRRRRR  
IHHHHBHHHHHHHH  
IBBBBBBBBBBBBB  
IHHHHBHHHHHHHH  
IRRRHBHRRRRRRR  
IRRRHBHRRRRRRR  
IRRRHBHRRRRRRR