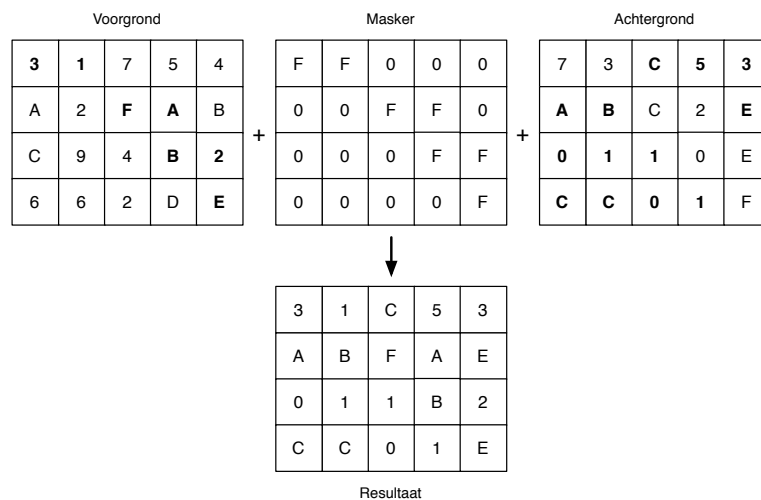


# Hexmaps



Een vaak voorkomende handeling in het domein van grafische gebruikersinterfaces is het combineren van tekeningen (bitmaps), bijvoorbeeld wanneer een besturingssysteem verschillende programmavensters op een bureaublad moet tonen. Concreet moet er dus een “voorggrond”-tekening worden gecombineerd met een “achtergrond”. Dit wordt meestal gedaan aan de hand van een binair “masker” dat aangeeft welke pixels van de achtergrond overschreven mogen worden.



De tekeningen in deze opgave bestaan uit hexadecimale getallen dewelke de kleurcode van de pixels aanduiden. Gevraagd wordt om gegeven een voorgrond, achtergrond en masker, een nieuwe tekening te genereren die voor- en achtergrond combineert volgens het masker.

## Invoer

De eerste regel van de invoer bevat een getal  $N$  waarbij  $1 \leq N \leq 200$  dewelke het aantal opgaven voorstelt. Daarna volgen  $N$  opgaven. Elke opgave bestaat uit 3 opeenvolgende tekeningen, in deze volgorde:

1. een “voorggrond”-tekening
2. een “masker”-tekening
3. een “achtergrond”-tekening

Alle 3 deze tekeningen hebben dezelfde breedte en hoogte. Een tekening begint met een regel met daarop breedte  $W$  en hoogte  $H$ , gegeven als gehele getallen met  $1 \leq W, H \leq 100$  en gescheiden door een spatie. Daarna volgen

H regels, elk bestaande uit W opeenvolgende karakters (niet gescheiden door komma's of spaties). Elke regel wordt beëindigd met een newline.

Elk karakter stelt een getal tussen 0 en 15 voor, via de karakters 0-9 en A-F. Een “masker” tekening bestaat uitsluitend uit de karakters 0 en F.

## **Uitvoer**

De uitvoer van je programma is een nieuwe tekening die bestaat uit alle pixels van de “voorggrond”-tekening waarvoor de overeenkomstige pixel in het “masker” op F staat, en alle pixels van de “achtergrond”-tekening waarvoor de overeenkomstige pixel in het “masker” op 0 staat. Je uitvoer-tekening dient in hetzelfde formaat uitgeprint te worden als de invoer-tekeningen.

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

## **Voorbeeld**

### **Invoer**

```
1
5 4
31754
A2FAB
C94B2
662DE
5 4
FF000
00FF0
000FF
0000F
5 4
73C53
ABC2E
0110E
CC01F
```

### **Uitvoer**

```
5 4
31C53
ABFAE
011B2
CC01E
```