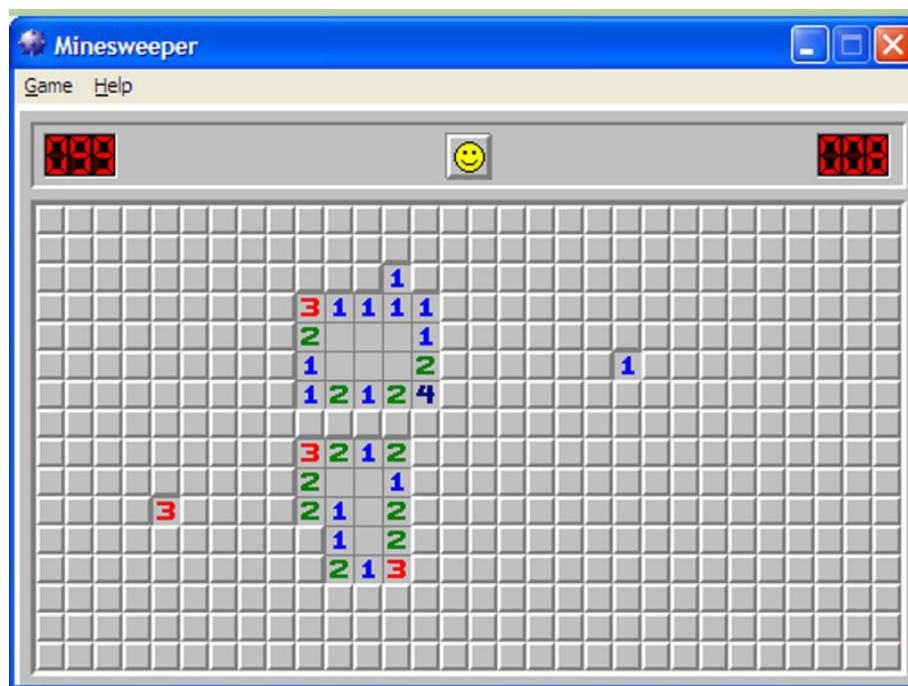


Hemuppgift: Minesweeper

Minesweeper var ett spel som brukade följa med i Windows-installationer tidigare än Windows 8. Av den anledningen är det lite av ett kultspel som mer eller mindre alla som ägt en dator med Windows XP som operativsystem har ägnat ett flertal timmar av sina liv åt.

Reglerna i Minesweeper är enkla. Spelet spelas på ett bräde bestående av rutor ordnade i ett antal rader och kolumner. Under ett antal av rutorna finns en mina gömd, medan resten av rutorna är tomma. Spelaren ska i tur och ordning klicka på rutorna, för att avslöja vad som finns under rutan. Om det finns en mina under rutan som spelaren klickat på är spelet förlorat. I annat fall visar rutan hur många minor som finns i angränsande rutor, och spelaren får fortsätta med att klicka på en ny ruta.

Om den ansökande mot förmodan inte har spelat Minesweeper förut kan spelet med fördel testas på följande länk: <http://minesweeperonline.com/>



1 Uppgiften

Den ansökande har nu i uppgift att skriva en funktion som räknar ut hur brädet förändras efter ett klick på en ruta. Funktionen tar ett bräde med ett antal gömda minor, ett antal öppnade och tomma rutor, samt ett antal öppnade rutor, där alltså antalet minor som angränsar till minan är angivet. Efter ett klick på en ruta ska följande hända:

1. Om det är en dold mina på rutan sprängs minan och spelet förloras
2. Om rutan angränsar till en eller flera minor visas det antal minor som angränsar till rutan, dvs. rutan visar en siffra 1-8.
3. Om rutan inte angränsar till en enda mina öppnas rutan som blank. Därefter öppnas alla angränsande rutor till den öppnade rutan, vilket fortsätter rekursivt tills det öppnade blanka området är helt inneslutet av öppnade rutor med siffror 1-8 på.

Med detta i åtanke är uppgiften för den ansökande att skriva en implementation av en funktion

```
static Board PerformMove(Board board, int iCol, int iRow);
```

där Board är en lämplig representation av ett bräde. Funktionen ska alltså givet ett bräde och koordinater för en ruta som det klickas på räkna ut ett uppdaterat bräde efter klicket och returnera detta.

Den ansökande har fått en uppsättning filer med C++-kod med några användbara hjälpklasser och hjälpfunktioner för att underlätta testning. I filen Sweeper.cpp är funktionen PerformMove() oimplementerad. Uppgiften är alltså att skriva en implementation av PerformMove() som följer beteendet som är beskrivet ovan.

Om den anökande av någon anledning inte kan eller vill genomföra uppgiften i C++ är det fritt fram att skriva funktionen PerformMove() i ett annat programspråk. I Java skulle till exempel funktionen kunna se ut som

```
static char[][] PerformMove(char[][] board, int iCol, int iRow){
```

om man låter brädet representeras av 2D-array. I Python skulle funktionens definition kunna vara

```
def PerformMove(board,iCol,iRow):
```

Om den ansökande väljer att skriva programmet i ett annat programmeringsspråk går det fortfarande att inspireras av den bifogade exempelkoden, men det är inget krav att använda exakt samma datatyper som i exempelkoden. Funktionen ska klara av bräden av de maximala dimensionerna 40x40.

2 Testning

Den ansökande kan testa sitt program på valfritt sätt. Nedan följer några exempel på inparametrar samt förväntad output från funktionen. Notationen som vi använder här är:

E = Tom oöppnad ruta

M = Öppnad ruta med mina

0 = Öppnad ruta utan minor som grannar

1-8 = Öppnad ruta med 1-8 minor som grannar

Exempel 1

Input:

```
[[ M E E
   E E E
   E M E ]]
```

Klick: 2 , 0

Output:

```
[[ M 1 0
   E 2 1
   E M E ]]
```

Exempel 2

Input:

```
[[ M E E
   E E E
   E M E ]]
```

Klick: 0 , 0

Output:

```
[[ X E E
   E E E
   E M E ]]
```

Exempel 3

Input:

```
[[ E E E E E
   E E M E E
   E E E E E
   E E E E E ]]
```

Klick: 0 , 3

Output:

```
[[ 0 1 E 1 0
   0 1 M 1 0
   0 1 1 1 0
   0 0 0 0 0 ]]
```