

Opgave 8

I denne opgave betragtes arrays, hvis elementer er *cifre*, det vil sige arrays af formen

$$X = (1, 2, 5, 3, 7, 8, 2, 0, 5, 8, 9, 6, 8, 9, 8)$$

Et *monotont afsnit* i et sådant array er et delarray hvori værdierne er ikke aftagende; fx er $X[3..5] = (3, 7, 8)$ et monotont afsnit i X , mens $X[1..3] = (2, 5, 3)$ ikke er et monotont afsnit. Vi betegner med $llma(X)$ længden af det længste monotone afsnit i X , og med $lsma(X)$ længden af det sidste monotone afsnit i X . For ovenstående X er $llma(X) = 4$ og $lsma(X) = 1$.

- a) Argumenter for at følgende algoritme er korrekt (find blandt andet en passende invariant og bevis, at den er gyldig).

Algoritme længsteMonotoneAfsnit(X):

Input: Array X med længde n

Output: længden af længste monotone afsnit $r = llma(X)$.

```
 $r \leftarrow 1; h \leftarrow 1; i \leftarrow 1$ 
while  $i \neq |X|$  do  $\{I\}$ 
  if  $X[i - 1] \leq X[i]$  then
     $h \leftarrow h + 1$ 
  else
     $h \leftarrow 1$ 
  if  $h > r$  then
     $r \leftarrow h$ 
   $i \leftarrow i + 1$ 
```

I stedet for et monotont afsnit i et array kan man interessere sig for en *monoton følge* i array'et, hvor en følge adskiller sig fra et afsnit ved, at dens elementer ikke behøver "af hænge sammen". Følgen

$$(2, 3, 5, 6, 8, 9)$$

udgør således den med understregning angivne monotone følge i

$$X = (1, \underline{2}, 5, \underline{3}, 7, 8, 2, 0, \underline{5}, 8, 9, \underline{6}, \underline{8}, \underline{9}, 8)$$

- b) Konkretiser de udkommenterede ubestemte stumper, så følgende algoritme bliver gyldig og korrekt. Med $\text{llmf}(X)$ betegner vi længden af den længste monotone følge i X , og med $\text{llmf}_d(X)$ betegnes længden af den længste monotone følge i X , hvis sidste ciffer er D .

Algoritme længsteMonotoneFølge(X):

Input: Array X med længde n

Output: længden r af længste monotone følge $\text{llmf}(X)$.

```
// Initialiser  $i$  og  $h$ 
```

```
while  $i \neq |X|$  do  $\{(h[d] = \text{llmf}_d(X[0..i]) \text{ for } 0 \leq d \leq 9) \wedge (0 \leq i \leq |X|)\}$ 
```

```
    // Iterer
```

```
    // Beregn  $r$ 
```