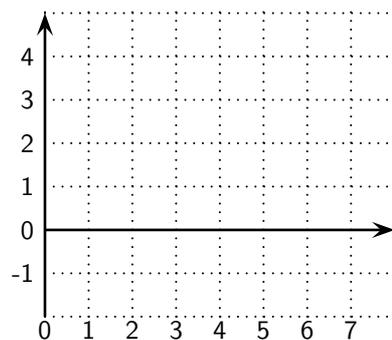
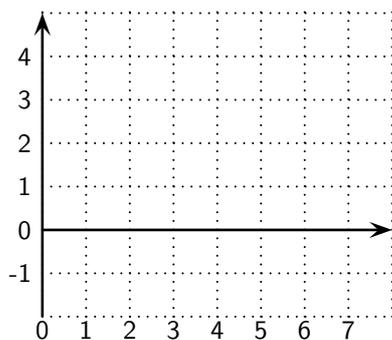
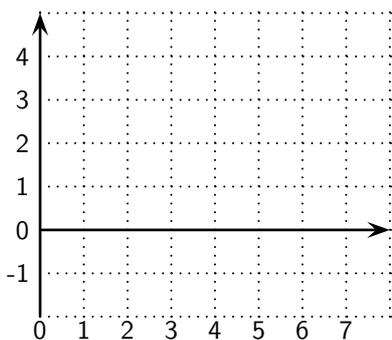
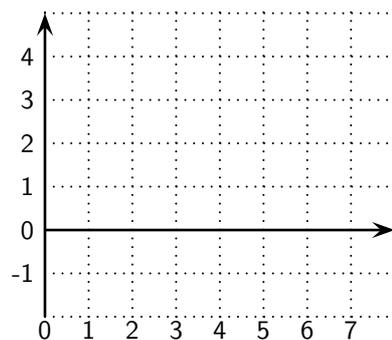
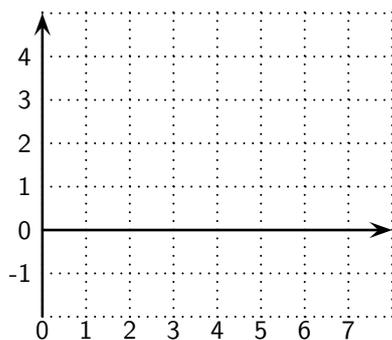
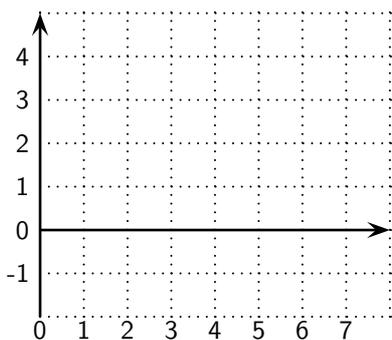
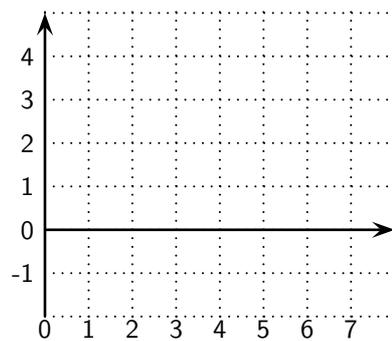
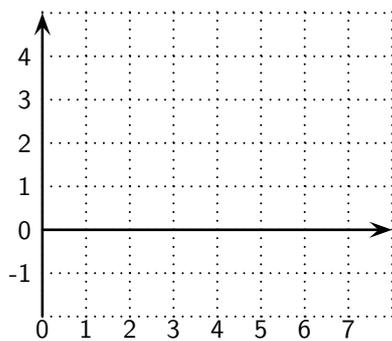
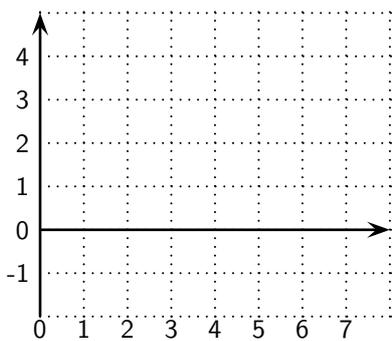
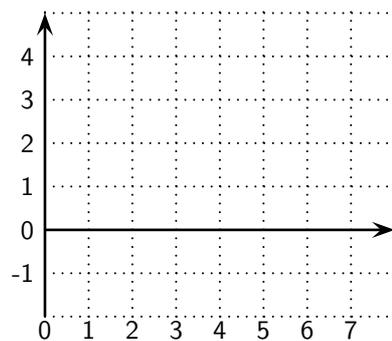
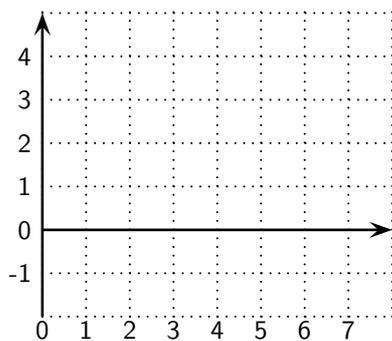
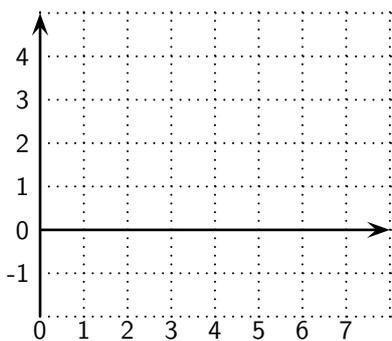




### Aufgabe 1 (8 Punkte)

Ermittle mit dem in der Vorlesung angegebenen Linearzeit-Algorithmus für AMSS alle maximal bewerteten Teilfolgen von  $a$  und gib dabei alle Zwischenschritte sowie die jeweils angewendeten Fälle an. Gib auch an, welche maximal bewerteten Teilfolgen vom Algorithmus ausgegeben werden.

$$a = (+2, -3, +2, -1, +4, -3, +2)$$



Vorname: \_\_\_\_\_ Name: \_\_\_\_\_ Matrikelnummer: \_\_\_\_\_

## Aufgabe 2 (8 Punkte)

Gib für  $t\$ = t_1 \cdots t_{10}\$ = aabababaab\$$  das zugehörige Suffix-Array  $A$  inklusive der zugehörigen LCP-Tabelle  $L$  an.

Zeichne dort die LCP-Intervalle mit den Suffix-Links (ohne die Suffix-Links von einelementigen LCP-Intervallen) ein.

Zeichne weiterhin den zugehörigen LCP-Intervall-Baum.

*Bemerkung:* Die Ordnung auf dem Alphabet  $\Sigma$  sei  $\$ < a < b$ .

$i$	$A[i]$	$L[i]$	$t^{A[i]}$ (und LCP-Intervalle / Suffix-Links)
0			
1			
2			
3			
4			
5			
6			
7			
8			
9			
10			

Vorname: \_\_\_\_\_ Name: \_\_\_\_\_ Matrikelnummer: \_\_\_\_\_

### **Aufgabe 3 (8 Punkte)**

Erstelle für das Wort  $t\$ = t_0 \cdots t_{10}\$ = aabaaabaaba\$$  ein Suffix-Array nach dem Algorithmus von Kärkkäinen und Sanders. Gib dabei alle Zwischenschritte an, wobei der rekursive Aufruf von Hand sortiert werden darf.

*Hinweis:* Beim Mischen von  $A_0$  mit  $A_{12}$  soll für jede Ergebnisposition in  $A$  angegeben werden, ob 1 oder 2 Zeichenvergleiche erforderlich sind oder ob auf die Ordnung von  $A_{12}$  zurückgegriffen wird.

Vorname: \_\_\_\_\_ Name: \_\_\_\_\_ Matrikelnummer: \_\_\_\_\_

#### Aufgabe 4 (8 Punkte)

Betrachte das Wort  $t = t_1 \cdots t_{12} \in \{A, I, N, S\}^*$  und die zugehörige Burrows-Wheeler-Transformierte  $\hat{t} = \hat{t}_0 \cdots \hat{t}_{12} = \text{SNSN\$NANAAAAI} \in \{\$, A, I, N, S\}$ .

- a) Gib die Tabellen  $C(\cdot)$  und  $Occ(\cdot, \cdot)$  für  $\hat{t}$  an.
- b) Rekonstruiere  $t$  aus  $\hat{t}$ .
- c) Konstruiere den Wavelet-Tree zu  $\hat{t}$ .
- d) Bestimme die Werte von  $Occ(A, 11)$  und  $Occ(N, 7)$  nach der in der Vorlesung vorgestellten Methode aus dem Wavelet-Tree aus c).

*Hinweis:* Das zu betrachtende Alphabet ist  $\Sigma \cup \{\$\} = \{\$, A, I, N, S\}$ , wobei die Ordnung auf dem Alphabet durch die hier angegebene Reihenfolge gegeben ist.

Vorname: \_\_\_\_\_ Name: \_\_\_\_\_ Matrikelnummer: \_\_\_\_\_

### **Aufgabe 5 (8 Punkte)**

Sei  $\Sigma$  ein Alphabet,  $t = t_1 \cdots t_n \in \Sigma^n$  und  $1 < k \in \mathbb{N}$ . Konstruiere einen Algorithmus mit optimaler Laufzeit, der alle kürzesten Wörter  $w \in \Sigma^*$  findet, die in  $t$  mindestens  $k$ -mal auftreten, aber das zugehörige gespiegelte Wort  $w^R$  nicht in  $t$  auftritt.

Zeige die Korrektheit des Algorithmus und analysiere dessen Laufzeit.

*Hinweis:* Für  $w = w_1 \cdots w_\ell \in \Sigma^\ell$  ist das zugehörige gespiegelte Wort  $w^R = w_\ell \cdots w_1$ .