Universität Siegen
Lehrstuhl Theoretische Informatik
Markus Lohrey

Winter Bonus Task

Task 1
Solve the following tasks.

(a) Compute $15 \cdot 13$ using the algorithm of Karatsuba.

Solution
Our base is $b = 10$. The length of the numbers is $n = 2$, so we will do one iteration of devide and conquer. We have $r = 15 = A \mid B$, hence $A = 1$, $B = 5$ and $s = 13 = C \mid D$, hence $C = 1$, $D = 3$.

$$\begin{aligned} rs &= AC \cdot 10^2 + (A - B)(D - C) \cdot 10^1 + (BD + AC) \cdot 10^1 + BD \\ &= 1 \cdot 1 \cdot 100 + (-4) \cdot 2 \cdot 10 + (5 \cdot 3 + 1 \cdot 1) \cdot 10 + 5 \cdot 3 \\ &= \cdots = 195 \end{aligned}$$

In total we have to compute 3 products.

(b) Compute the same product with Karatsuba, where 15 and 13 are given in base 2.
Hint: For example, 12 (base 10) is 1100 in base 2.

Solution
Our base is $b = 2$. The length of the numbers is $n = 4$, so we will do two iterations of devide and conquer. We have $r = 1111 = A \mid B$, hence $A = 11$, $B = 11$ and $s = 1101 = C \mid D$, hence $C = 11$, $D = 01$.

$$\begin{aligned} rs &= AC \cdot 2^4 + (A - B)(D - C) \cdot 2^2 + (BD + AC) \cdot 2^2 + BD \\ &= 11 \cdot 11 \cdot 10000 + 00 \cdot (-10) \cdot 100 + (11 \cdot 01 + 11 \cdot 11) \cdot 100 + 11 \cdot 01 \end{aligned}$$

The next step yields $11 \cdot 11 = 1001$ (3 products), $00 \cdot (-10) = 0$ (one product) and $11 \cdot 01 = 11$ (3 products). Your solutions in the exam should show each step in detail. This gives us $rs = 11000011$.

(c) How many more products do you need to compute for case (b) compared to case (a)?

Solution
The correct answer is $7 - 3 = 4$ products, but one can argue that $00 \cdot (-10) = 0$ can in fact be devided into 3 subproducts. Hence, $9 - 3 = 6$ would have been graded with full points as well, if the argumentation is correct.

Task 2
Sort the array

$$[912, 442, 345, 842, 159, 126, 875, 970]$$

using Radixsort and Mergesort.

Solution
Radixsort:

First we sort the array with respect to the least significant digit.

0	1	2	3	4	5	6	7	8	9
970		912, 442, 842			345, 875	126			159

Now the second digit.

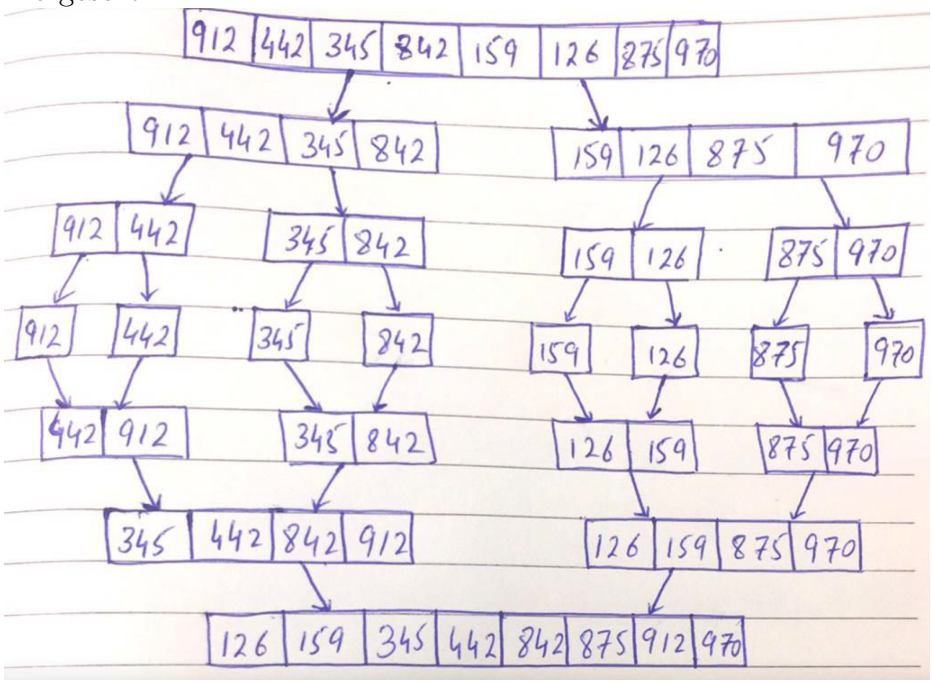
0	1	2	3	4	5	6	7	8	9
	912	126		442, 842, 345	159		970, 875		

Now the most significant digit.

0	1	2	3	4	5	6	7	8	9
	126, 159		345	442				842, 875	912, 970

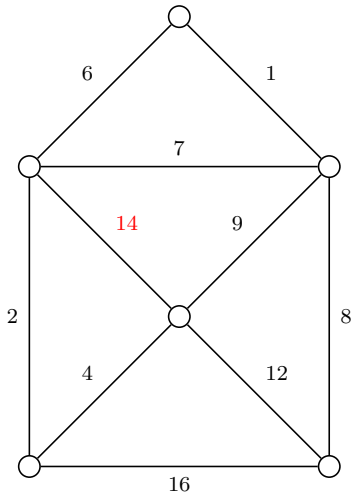
The array is sorted: $[126, 159, 345, 442, 842, 875, 912, 970]$

Mergesort:



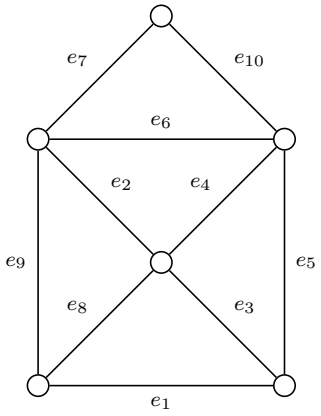
The upper part shows the mergesort (devide) step and the lower part shows the merge step. For the maximum number of points, one should give a more detailed example how 2 arrays (lengths 2 or more) are merged.
The version with purely functions and code like on Sheet 2 would also be accepted.

Task 3
Compute a maximal spanning tree using Kruskal's algorithm for the following graph:

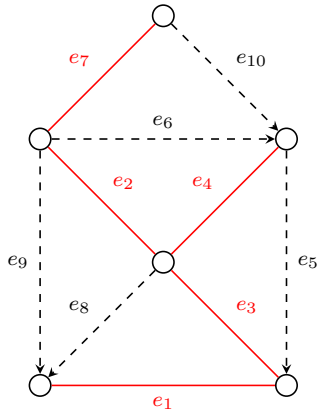


How does the spanning tree look like, if you change the red value to 1?

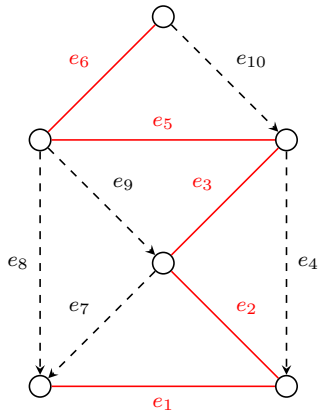
Solution
We first sort the edges by their weights in decreasing order. To illustrate better what it yields, we show the graph one more time:



Kruskal's algorithm now takes greedily any heavy edge into the set F, such that (V, F) has no cycles. Starting with $F = \emptyset$ this means that after sorting the edges, we will do $F = F \cup \{e_i\}$ for increasing i, but only if the new edge does not lead to a cycle.
In the end the spanning subtree has the following edges: $F = \{e_1, e_2, e_3, e_4, e_7\}$.



Changing the red value in the graph (14) to 1 means that we have to reorder the edges. This yields the following maximal spanning tree:



Calculating the total weight of the spanning tree is nice, but does not give any bonuspoints, if not requested.