
Encoding II, Algorithms and Data Structures

Elias Gestrich

Exercise 1: IEE 754 Number format

a) first bit: 1 \rightarrow negative

following 8 bits: $10000001 = 0111\,1111_2 + 10_2 \rightarrow$ Exponent is 10_2

other bits: $101000000000000000000000 \rightarrow$ Mantissa is 1.101_2

$$\Rightarrow 11000000110100000000000000000000 = -1.101_2 \times 10_2^{10_2} = -1101.1_2 = -13.5_{10}$$

b) positive: first bit = 0

$20.5_{10} = 1\,0100.1_2 \rightarrow$ mantissa: 010010000000000000000000 , exponent: $0111\,1111_2 + 100_2 = 1000\,0011_2$

$$\Rightarrow 01000001101001000000000000000000$$

Exercise 2: ASCII

a) $S = 53_{16}$, $k = 6B_{16}$, $I = 49_{16}$: "SkI" = $536B49_{16} = 101\,0011\,0110\,1011\,0100\,1001_2$

Exercise 3: Binary Search Trees

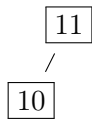
a)

insert(11)

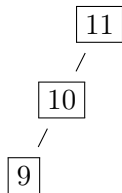
empty tree: 11 becomes root



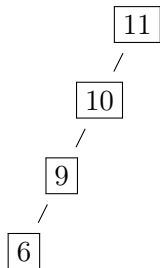
insert(10)

 $10 < 11$ 

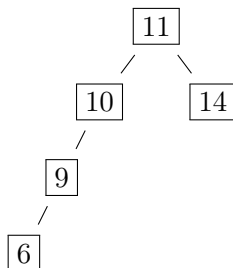
insert(9)

 $9 < 11 \rightarrow 9 < 10$ 

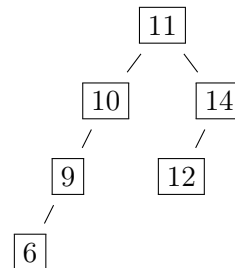
insert(6)

 $6 < 11 \rightarrow 6 < 10 \rightarrow 6 < 9$ 

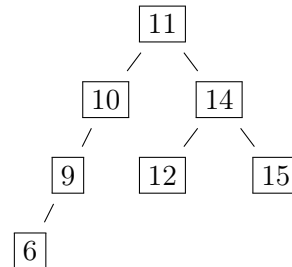
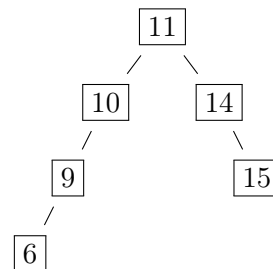
insert(14)

 $14 > 11$ 

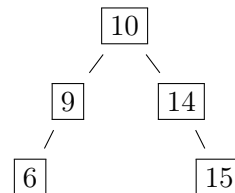
insert(12)

 $12 > 11 \rightarrow 12 < 14$ 

insert(15)

 $15 > 11 \rightarrow 15 > 14$ remove(12) \rightarrow 12 has no child:

remove(11) \rightarrow 11 has two childs: replace 11 with the most right element of the left child (the 10) and remove the child, because 10 has one child on the left side move this child to the left side of the root:



Exercise 4: Hashing

a) 13

b)

0	→
1	→ 40 → 27
2	→
3	→ 16
4	→
5	→ 5 → 31
6	→ 19
7	→ 20 → 7
8	→ 21 → 34
9	→
10	→ 23
11	→
12	→ 12

c)

newly inserted key	20	19	5	21	7	40	23	31	12	16	27	34
0												34
1						40	40	40	40	40	40	40
2											27	27
3										16	16	16
4												
5			5	5	5	5	5	5	5	5	5	5
6		19	19	19	19	19	19	19	19	19	19	19
7	20	20	20	20	20	20	20	20	20	20	20	20
8				21	21	21	21	21	21	21	21	21
9					7	7	7	7	7	7	7	7
10							23	23	23	23	23	23
11								31	31	31	31	31
12									12	12	12	12