DSA Blatt 05

Leonard Oertelt 1276156
Julian Opitz 1302082

Aufage 11:

a)

$$x^{(2^{-k})} = x^{\frac{1}{2^k}}$$

k = 4 :

$$x^{(1/16)} = \left(\left(\left(x^{(1/2)}\right)^{(1/2)}\right)^{(1/2)}\right)^{(1/2)}$$

```
Algorithm ln(x, anzahl) : double
        wert <= sqrt(x)
        produkt = 2 /(wert + 1/wert)
        for k <= 2 to anzahl do
                wert <= sqrt(wert)
                produkt <= produkt / (wert + 1/wert)
        return 0.5 * (x - 1/x) * produkt * 2^anzahl
```

12.a

```
Algorithm max(liste[]) : integer
        max <= liste[0]
        for i <= 1 to liste[].length do
                if liste[i] > max
                        max <= liste[i]
        return max
```

```
Algorithm min(liste[]) : integer
        min <= liste[0]
        for i <= 1 to liste[].length do
                if liste[i] < min
                        min <= liste[i]
        return min
```

12.b

insgesamt 2*(n-1) Schlüsselvergleiche

12.c

```
Algorithm divideConquer(liste[]) : array
        if liste[].length = 1 then
                min <= liste[0]
                max <= min
                return max,min
        if liste[].length = 2 then
                if liste[0] < liste[1]
                        min <= liste[0]
                        max <= liste[1]
                        return max,min
                min <= liste[1]
                max <= liste[0]
                return max,min
        middle <= length/2
        result1 <= divideConquer( new liste[0 to middle-1])
        result2 <= divideConquer( new liste[middle to length-1])
        if result1 < result2 then
                min <= result1
                max <= result2
                return max,min
        min <= result2
        max <= result1
        return max,min
```

12.d

```
Algorithm divideConquer(liste[], a, b) : array
    if a = b then
        min <= liste[a]
        max <= min
        return {min,max}
    if (b-a) = 1 then
        if liste[a] < liste[b]
            min <= liste[a]
            max <= liste[b]
            return max,min
        min <= liste[b]
        max <= liste[a]
        return {min,max}

    middle <= (a+b)/2
    result1[] <= divideConquer( liste[], a, middle-1)
    result2[] <= divideConquer( liste[], middle, b)

    if result1[0] <= result2[0] then
        min <= result1[0]
    else
        min <= result2[0]

    if result1[1] >= result2[1] then
        max <= result1[1]
    else
        max <= result2[1]
    return {min,max}
```