Aufgabe 1

true := λt f. t
false := λt f. f

zero := λz s. z
succ := λn z s. s (n z s)

one := λ z s. s (z)
two := λz s. s (s z)

1.1

add := λn m. n m (λn z s. s (n z s))

1.2

not := λb. b (λt f. f) (λt f. t)
even := λ n . n (λt f. t) (λb. b (λt f. f) (λt f. t))

Aufgabe 2

2.1

lsb := λ l z o. l
zero := λ b l z o. z (b l z o)
one := λ b l z o. o (b l z o)

one (zero (zero lsb))
= (λ b l z o. o (b l z o)) (zero (zero lsb))
→β (λ l z o. o (zero (zero lsb) l z o))
= (λ l z o. o ((λ b l z o. z (b l z o)) (zero lsb) l z o)
→β (λ l z o. o ((λ l z o. z ((zero lsb) l z o)) l z o)
→β* (λ l z o. o (z (zero lsb l z o)))
= (λ l z o. o (z ((λ b l z o. z (b l z o)) lsb l z o)))
→β (λ l z o. o (z (z (lsb l z o))))
→ (λ l z o. o (z (z (l))))

2.2

zero lsb

shift := λ b . b ((λ b l z o. z (b l z o)) (λ l z o. l))
(λ b l z o. z (b l z o)) } zero
(λ b l z o. o (b l z o)) } one

shift (one lsb)
= (λ b. b (zero lsb) zero one) (one lsb)
→β (one lsb) (zero lsb) zero one
= (λ b l z o. o (b l z o)) lsb (zero lsb) zero one
→β (λ l z o. o (lsb l z o)) (zero lsb) zero one
→β (λ z o. o (lsb (zero lsb) z o)) zero one
→β (λ o. o (lsb (zero lsb) zero o)) one
→β one (lsb (zero lsb) zero one)
= (λ b l z o. o (b l z o)) (lsb (zero lsb) zero one)
→β (λ l z o. o ((lsb (zero lsb) zero one) l z o)
= (λ l z o. o ((one (λ l z o. l) (zero lsb) zero one) l z o)
→β* (λ l z o. o (zero lsb l z o))
= (λ l z o. o ((λ b l z o. z (b l z o)) lsb l z o)
= (λ l z o. o (z (lsb l z o))
= λ l z o. o (z ((λ l z o. l) l z o
= λ l z o. o (z (l))

Aufgabe 3

3.1

binleer := λ e k. e
binknoten := λ a l r e k. k a (l e k) (r e k)

3.2

binKnoten one (binknoten one binleer binleer) binleer

= (λ a l r e k . k a (l e k) (r e k)) one (binknoten one binleer binleer) binleer
→β (λ l r e k. k one (l e k) (r e k)) (binknoten one binleer binleer) binleer
→β (λ r e k. k one ((binknoten one binleer binleer) e k) (r e k)) binleer
→β (λ e k. k one ((binknoten one binleer binleer) e k) (binleer e k))
→β λ e k. k one ((λ a l r e k. k a (l e k) (r e k)) one binleer binleer) e k) (binleer e k)
→β λ e k. k one (λ l r e k. k one (l e k) (r e k)) binleer binleer) e k) (binleer e k)
→β* λ e k. k one (λ e k. k one (binleer e k) (binleer e k)) e k) (binleer e k)
→β λ e k. k one (k one (binleer e k) (binleer e k)) (binleer e k)
→β* λ e k. k one (k one e e) e ⇒ ssp Baum

3.3

add3 := λ a l r. add a (add l r)

knotenSumme := λ b. b (λ l z. z)

(λ a l r. add a (add l r))

3.4

knotenSumme bspBaum

= knotenSumme $(\lambda e\, k.\ k\ one\ (k\ one\ e\ e)\ e)$

$= (\lambda b.\ b\ zero\ add\ 3)\ (\lambda e\, k.\ k\ one\ (k\ one\ e\ e)\ e)$
$\to_\beta (\lambda e\, k.\ k\ one\ (k\ one\ e\ e)\ e)\ zero\ add\ 3$
$\to_\beta^* add\ 3\ one\ (add\ 3\ one\ zero\ zero)\ zero$
$\to_\beta (\lambda a\, l\, r.\ add\ a\ (add\ l\ r))\ one\ (add\ 3\ one\ zero\ zero)\ zero$
$\to_\beta (\lambda l\, r.\ add\ one\ (add\ l\ r))\ (add\ 3\ one\ zero\ zero)\ zero$
$\to_\beta (\lambda r.\ add\ one\ (add\ (add\ 3\ one\ zero\ zero)\ r))\ zero$
$\to_\beta add\ one\ (add\ (add\ 3\ one\ zero\ zero)\ zero)$
$= (\lambda n\, m.\ n\ m\ succ)\ one\ (add\ (add\ 3\ one\ zero\ zero)\ zero)$
$\to_\beta^* one\ (add\ (add\ 3\ one\ zero\ zero)\ zero)\ succ$
$= one\ ((\lambda n\, m.\ n\ m\ succ)\ (add\ 3\ one\ zero\ zero)\ zero)\ succ$
$\to_\beta one\ ((\lambda a\, l\, r.\ add\ a\ (add\ l\ r))\ one\ zero\ zero)\ zero\ succ)\ succ$
$\to_\beta^* one\ ((add\ one\ (add\ zero\ zero))\ zero\ succ)\ succ$

$= one\ (((\lambda n\, m.\ n\ m\ succ)\ one\ (add\ zero\ zero))\ zero\ succ)\ succ$

$\to_\beta^* one\ ((one\ (add\ zero\ zero)\ succ)\ zero\ succ)\ succ$

$= one\ ((one\ ((\lambda n\, m.\ n\ m\ succ)\ zero\ zero)\ succ)\ zero\ succ)\ succ$
$\to_\beta^* one\ ((one\ (zero\ zero\ succ)\ succ)\ zero\ succ)\ succ$
$\to_\beta^* one\ ((one\ zero\ succ)\ zero\ succ)\ succ$
$= one\ (((\lambda z\, s.\ s\ (z))\ zero\ succ)\ zero\ succ)\ succ$
$\to_\beta^* one\ (((succ\ zero)\ zero\ succ)\ zero\ succ)\ succ$
$= one\ ((one\ zero\ succ)\ zero\ succ)\ succ$
$= one\ ((((\lambda z\, s.\ s\ (z))\ zero\ succ)\ zero\ succ)\ succ$
$\to_\beta^* one\ ((succ\ zero)\ zero\ succ)\ succ$
$\to_\beta one\ ((one\ zero\ succ)\ succ$
$= one\ ((\lambda z\, s.\ s\ z)\ zero\ succ)\ succ$
$\to_\beta one\ (\lambda s.\ s\ zero)\ succ)\ succ$
$\to_\beta one\ (succ\ zero)\ succ$
$= one\ ((\lambda n\, z\, s.\ s\ (n\ z\ s))\ zero)\ succ$
$\to_\beta one\ (\lambda z\, s.\ s\ (zero\ z\ s))\ succ$
$\to_\beta one\ (\lambda z\, s.\ s\ (z))\ succ$
$= one\ one\ succ$
$= (\lambda z\, s.\ s\ z)\ one\ succ$
$\to_\beta (\lambda s.\ s\ one)\ succ$
$\to_\beta succ\ one$
$= (\lambda n\, z\, s.\ s\ (n\ z\ s))\ one$
$\to_\beta (\lambda z\, s.\ s\ (one\ z\ s))$
$= \lambda z\, s.\ s\ ((\lambda z\, s.\ s\ z)\ z\ s)$
$\to_\beta \lambda z\, s.\ s\ ((\lambda s.\ s\ z)\ s)$
$\to_\beta \lambda z\, s.\ s\ (s\ z)$ ✓ Ja das Ergebnis entspricht $S(S\,z)$