Aufgabe 1

Ege Tekin, Paul Tröster

TI Ü2

$f(1)-1 = 0$

$f(2)-1 = 0$

$f(3)-1 = 1$

$f(4)-1 = 2$

$f(5)-1 = 4$

$f(6)-1 = 7$

$f(7)-1 = 12$

$s(0) = 0$

$s(1) = 1$

$s(2) = 2$

$s(3) = 4$

$s(4) = 7$

$s(5) = 12$

$s(6) = 20$

$f(3) = f(2) + f(1)$

$s(3) \hat{=}\ s(0)+1 \quad 1+1 \quad s(1)$

$s(n) = \sum_{0}^{n} f(n)$

$s(n) = \sum_{0}^{n} f(n-2) + f(n-1)$

$* \; n \geq 2$

$* \; n \geq 1$

$s(n) = s(n-1) + f(n)$

$*$ Rekursive Def. für $s(n)$

$s(1) = s(0) + f(1)$
$s(2) = s(1) + f(2)$
$s(3) = s(2) + f(3)$
$s(4) = s(3) + f(4)$
$s(5) = s(4) + f(5)$
$s(6) = s(5 + f(6)$

$\Longrightarrow$

$1 = 0 + 1$
$2 = 1 + 1$
$4 = 2 + 2$
$7 = 4 + 3$
$12 = 7 + 5$
$20 = 12 + 8$

Aufgabe 1

z: $f(n-1) + f(n-2) = 2 \cdot f(n-2) + f(n-3)$

*für $n \geq 3$

IA

$f(2) + f(1) = 2f(1) + f(0)$

$\Leftrightarrow 1 + 1 = 2 \cdot 1 + 0$

$\Leftrightarrow 2 \overset{\checkmark}{=} 2$

IV

Wir setzen voraus, $n = k$ $k \in \mathbb{N} : k \geq 3$

$f(k-1) + f(k-2) = 2f(k-2) + f(k-3)$

IS

Angenommen, $n = k+1$

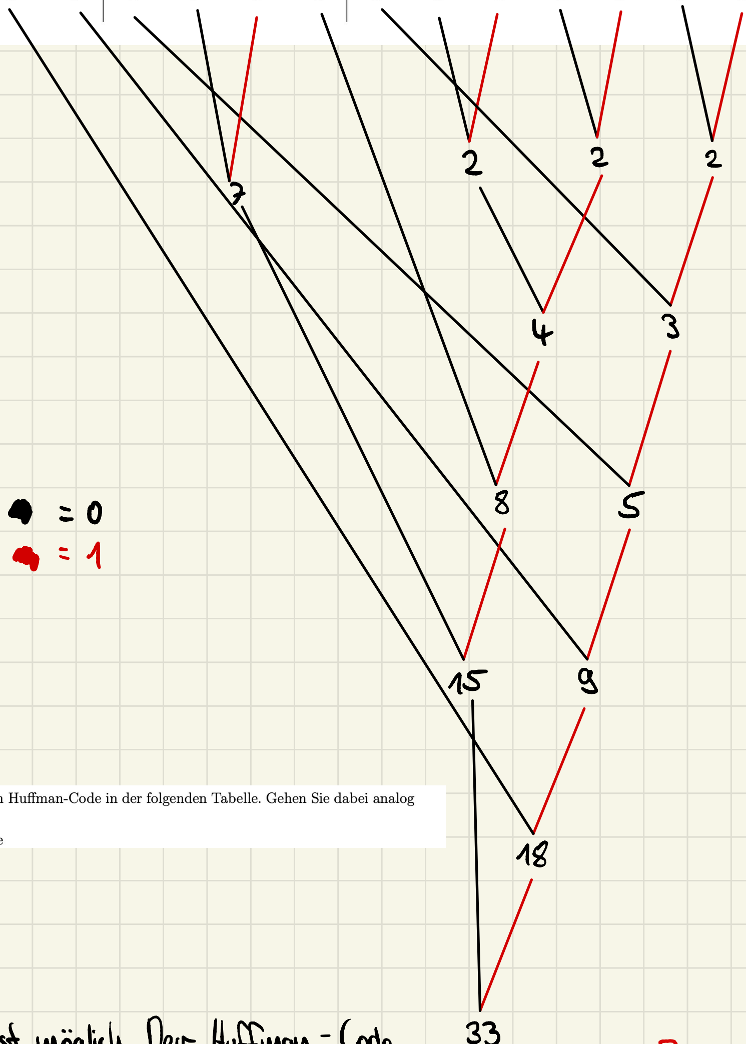
$f(k) + f(k-1) \overset{?}{=} 2f(k-1) + f(k-2)$ *subtrahiere

$= f(k) \overset{\checkmark}{=} f(k-1) + f(k-2)$

*Das gilt, weil es so angegeben ist, dass $f(n) = f(n-1) + f(n-2)$ stimmt. (laut dem Blatt)

Aufgabe 2

b)	E	U	I	S	H	T	B	D	C	A	L	R	K
	9	4	2	3	4	4	1	1	1	1	1	1	1



● = 0

● = 1

b) Geben Sie den resultierenden Huffman-Code in der folgenden Tabelle. Gehen Sie dabei analog zur Vorlesung vor.

Buchstabe	Huffman-Code
K	111111
B	11110
I	1110
E	10

c) Ja, das ist möglich Der Huffman-Code ist in der Minimierung der Codewortlänge optimal aber nicht hinsichtlich der Geschwindigkeit der Decodierung.

```
# Load n = 10 into S[30]
LOADI 10
STORE 30

# Initialize carry into S[31] (a)
LOADI 1
STORE 31

# Initialize carry for solution in S[33] (b)
LOADI 0
STORE 33

# ---------------------------

# Subtract one from n
LOAD 30
SUBI 1
STORE 30

# Jump to end if n < 0
JUMPC < 9

# Load b onto b-temp and initialize a-temp
LOAD 33
STORE 32

# Calculate b
LOAD 33
ADD 31
STORE 33

# Set a to previous b
LOAD 32
STORE 31
JUMP -11

# ---------------------------

# Print the result and loop forever
PRINT 33
JUMP 0
```

missing special cases for $n = 0, 1$

a)

Behauptung: Wenn ein Zeichen a_i in dem Alphabet $A = \{a_1, \ldots, a_m\}$ eine Häufigkeit $p(a_i) > 0.5$ hat wird der Huffman-Code ihm ein Codewort der Länge 1 zugeordnen.

Angenommen $p(a_i) > 0.5$, dann übersteigt die Häufigkeit von a_i die kombinierte Häufigkeit vom dem Rest im Alphabet, da die Summe der Häufigkeiten gleich 1 ist. Das bedeutet, dass kein anderer Knoten oder keine Kombination von Knoten eine höhere Häufigkeit als a_i erreichen kann.

b)

~~**Behauptung:** Falls einem Zeichen a_i in einem Huffman Code ein Codewort der Länge 1 zugewiesen wird, muss dessen Häufigkeit $p(a_i)$ mindestens $\frac{1}{3}$ betragen.~~

—

c)

~~**Behauptung:** Bei einer Gleichverteilung der Zeichenhäufigkeiten $p(a_i) = \frac{1}{m}$ in einem Huffman Code, wobei die Anzahl der Zeichen m eine Zweierpotenz ist, beträgt die mittlere Codelänge $\log_2(m)$.~~

—