

Algoritmer og Datastrukturer 2

Gerth Stølting Brodal

Grådige Algoritmer
[CLRS 16.1-16.3]



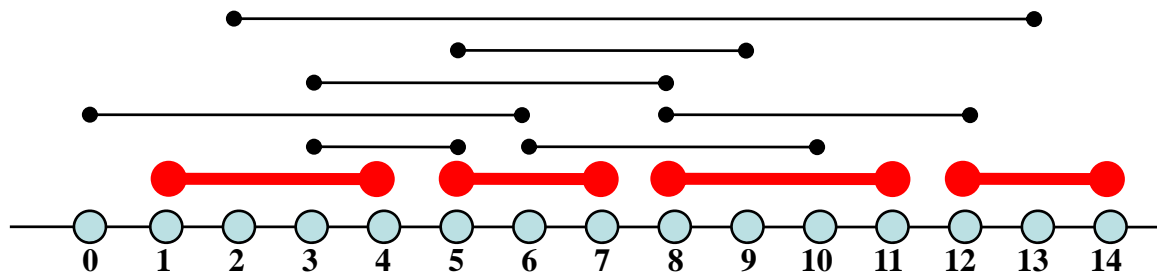
AARHUS UNIVERSITET

Grådige Algoritmer

- Problemer hvor en løsning kan konstrueres ud fra en løsning for kun **ét mindre delproblem**
- Delproblemet kan identificeres effektivt
(simplere end dynamisk programmering!)

Udvælgelse af Aktiviteter

i	1	2	3	4	5	6	7	8	9	10	11
s_i	1	3	0	5	3	5	6	8	8	2	12
f_i	4	5	6	7	8	9	10	11	12	13	14

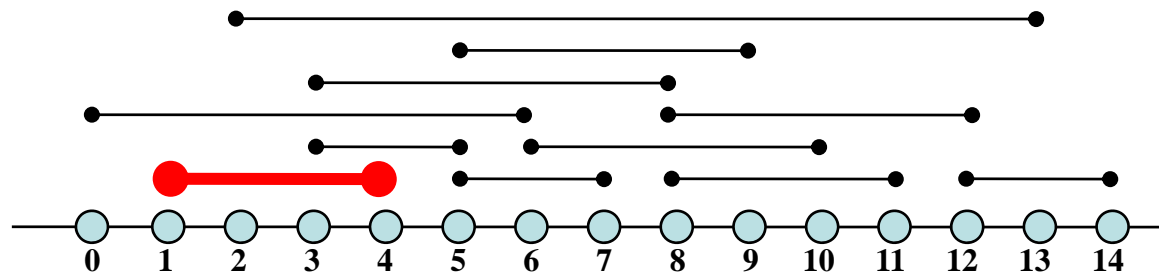


Input: n aktiviteter med starttid s_i og sluttid f_i

Output: Maximal mængde ikke-overlappende aktiviteter

Udvælgelse af Aktiviteter

i	1	2	3	4	5	6	7	8	9	10	11
s_i	1	3	0	5	3	5	6	8	8	2	12
f_i	4	5	6	7	8	9	10	11	12	13	14



Observation: Der findes altid en maximal løsning som indeholder aktiviteten med den *tidligste sluttid* (*grådig-valg egenskab*)

Preprocessing: Sorter aktiviteterene efter sluttidspunktet

Udvælgelse af Aktiviteter

i	1	2	3	4	5	6	7	8	9	10	11
s_i	1	3	0	5	3	5	6	8	8	2	12
f_i	4	5	6	7	8	9	10	11	12	13	14

GREEDY-ACTIVITY-SELECTOR(s, f)

```
1   $n \leftarrow \text{length}[s]$ 
2   $A \leftarrow \{a_1\}$ 
3   $i \leftarrow 1$ 
4  for  $m \leftarrow 2$  to  $n$ 
5      do if  $s_m \geq f_i$ 
6          then  $A \leftarrow A \cup \{a_m\}$ 
7               $i \leftarrow m$ 
8  return  $A$ 
```

Tid $O(n \cdot \log n)$

Huffman koder

Ascii Tabel

$$83_{10} = 53_{16} = 1010011_2$$

	0	1	2	3	4	5	6	7
0	NUL	DLE	space	0	@	P	`	p
1	SOH	DC1 XON	!	1	A	Q	a	q
2	STX	DC2	"	2	B	R	b	r
3	ETX	DC3 XOFF	#	3	C	S	c	s
4	EOT	DC4	\$	4	D	T	d	t
5	ENQ	NAK	%	5	E	U	e	u
6	ACK	SYN	&	6	F	V	f	v
7	BEL	ETB	'	7	G	W	g	w
8	BS	CAN	(8	H	X	h	x
9	HT	EM)	9	I	Y	i	y
A	LF	SUB	*	:	J	Z	j	z
B	VT	ESC	+	;	K	[k	{
C	FF	FS	,	<	L	\	l	
D	CR	GS	-	=	M]	m	}
E	SO	RS	.	>	N	^	n	~
F	SI	US	/	?	O	_	o	del

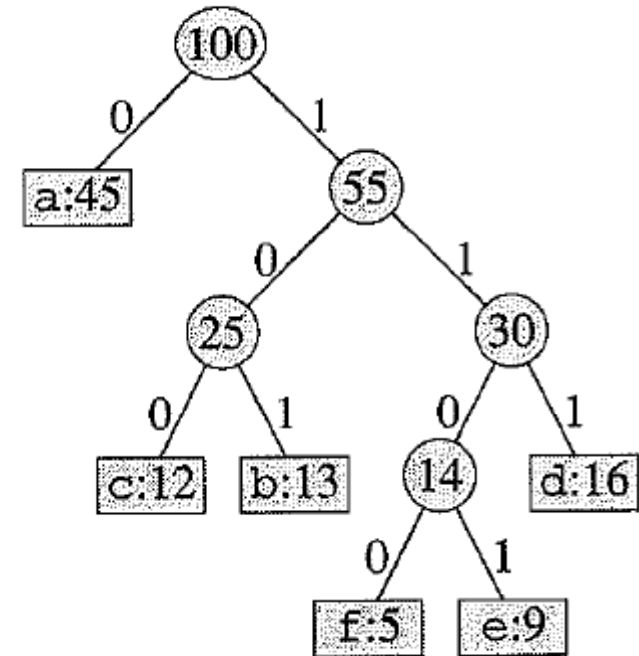
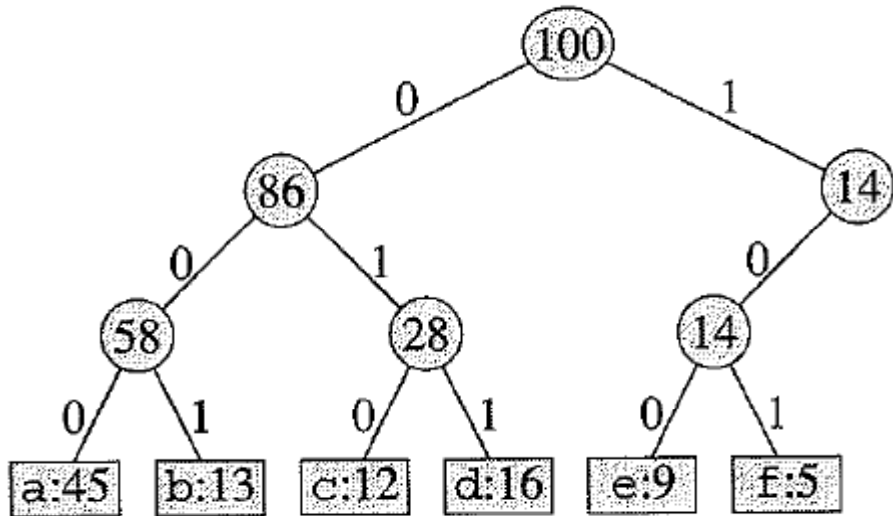
Komprimering

Givet frekvensen af symbolerne i input, erstat input symbolerne med **kortere bitstreng** (fixed-længde eller variabel-længde)

	a	b	c	d	e	f
Frequency (in thousands)	45	13	12	16	9	5
Fixed-length codeword	000	001	010	011	100	101
Variable-length codeword	0	101	100	111	1101	1100

NB: Variabel-længde skal være **prefix-fri**

Fixed-længde vs variabel-længde



	a	b	c	d	e	f
Frequency (in thousands)	45	13	12	16	9	5
Fixed-length codeword	000	001	010	011	100	101
Variable-length codeword	0	101	100	111	1101	1100

Huffman Koder

HUFFMAN(C)

1 $n \leftarrow |C|$

2 $Q \leftarrow C$

3 **for** $i \leftarrow 1$ **to** $n - 1$

4 **do** allocate a new node z

5 $left[z] \leftarrow x \leftarrow \text{EXTRACT-MIN}(Q)$

6 $right[z] \leftarrow y \leftarrow \text{EXTRACT-MIN}(Q)$

7 $f[z] \leftarrow f[x] + f[y]$

8 INSERT(Q, z)

9 **return** EXTRACT-MIN(Q)

▷ Return the root of the tree.

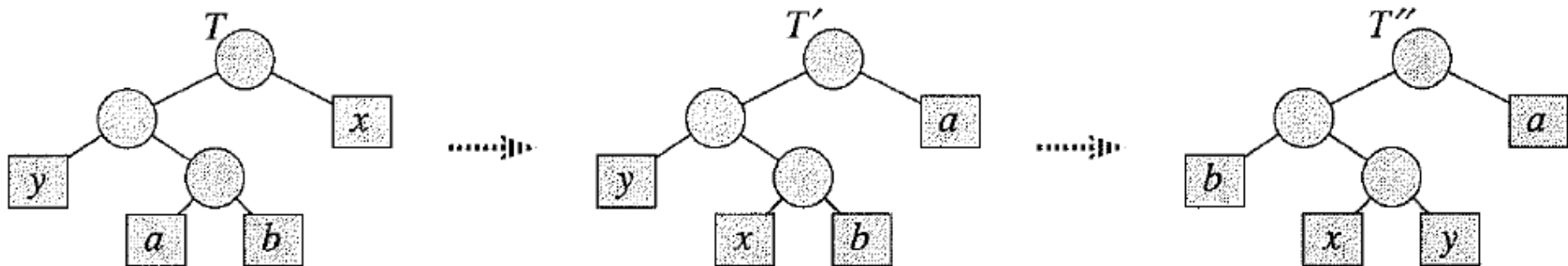
Tid $O(n \cdot \log n)$

Huffman Koder

(a) f:5 e:9 c:12 b:13 d:16 a:45

Korrektheden af Huffman Koder

Sætning Der findes altid en optimal prefix kode hvor de to mindst hyppige symboler (x og y) har samme kode og kun afskiller sig i sidste bit



Generelle Algoritmiske Design Teknikker

- **Del-og-Kombiner**
 - Disjunkte delproblemer
- **Dynamisk Programmering**
 - Overlappende delproblemer
 - Systematisk beregning af løsninger til alle mulige delproblemer
- **Grådige Algoritmer**
 - Kun et delproblem