Aktivitätsdiagramme

- prepeare_huffman_tree_element
- read_Code_Table [illegible] decodieren
- write_Code_Table (in Datei)

10000 1000...

100

101

001

100

is_code(10000 ...)
{
ob 10000
101
}
): bool

(code ;

while(!notfound())
{
CodeTable [illegible] get code( ) ;
}

1011

1001

Magic Number !

void* this;

Struct Auto

{

A_Räder = 4

gibAnzahlRäder (obj Auto)

}

Auto obj

int gibAnzahlRäder ( )

{

this.A_Räder

}

auto1.gibAnzahlRadr (auto1)

void* focus(void* obj)

{

this = obj;

return obj;

}

int gj

focus(auto1)

$auto1.gibAnzahlRäder()

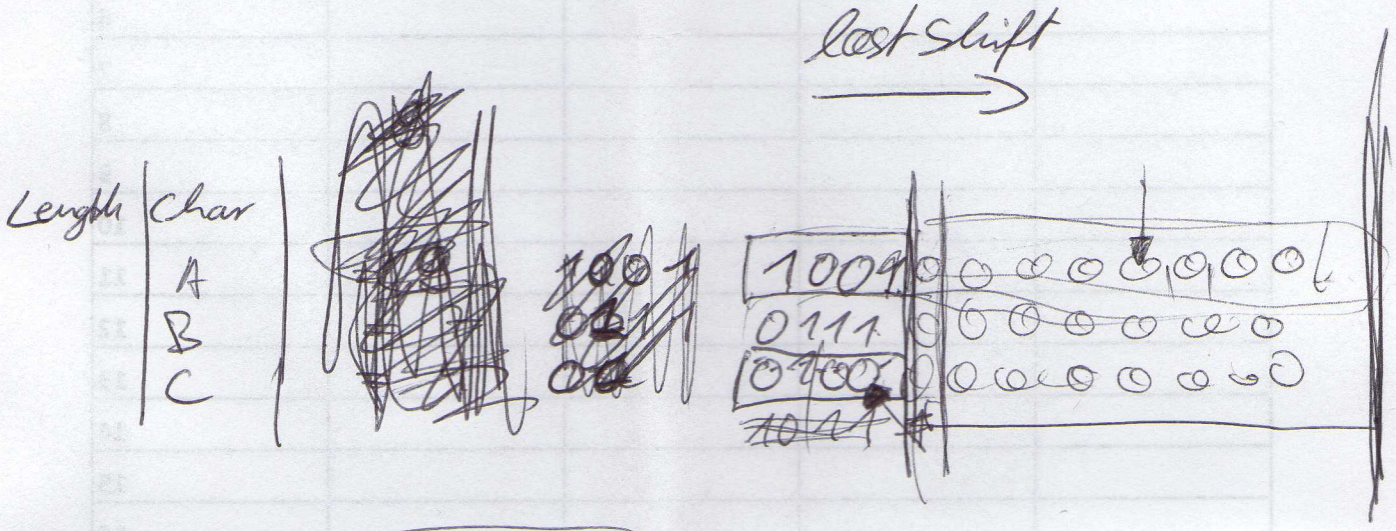
focus(...)

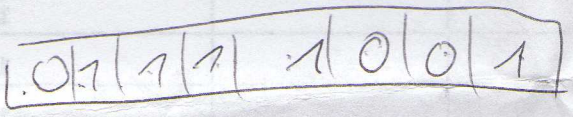
focus(auto1).gibAnzahlRäder() =

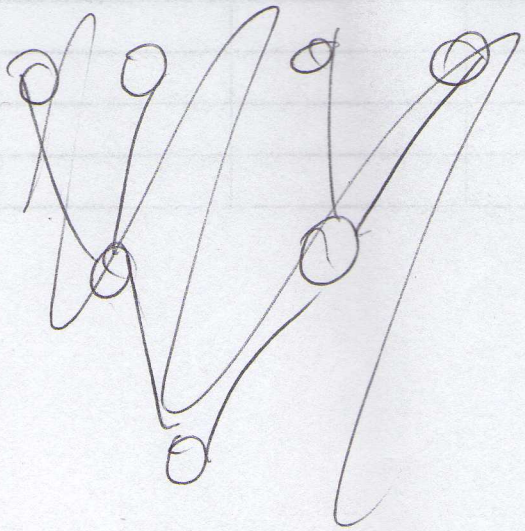
A B C

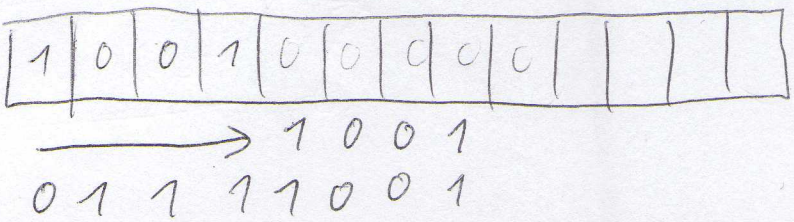
Codierer

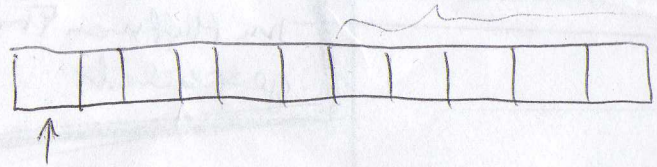
3, 4, 5, 5

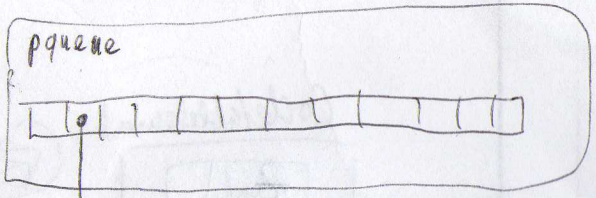


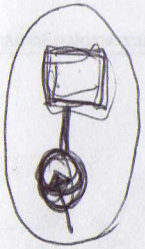


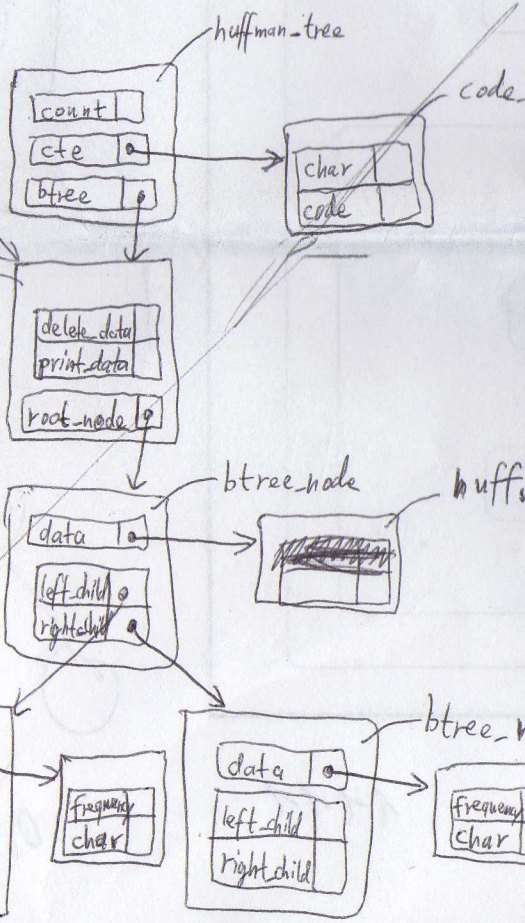


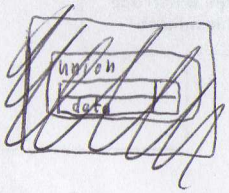


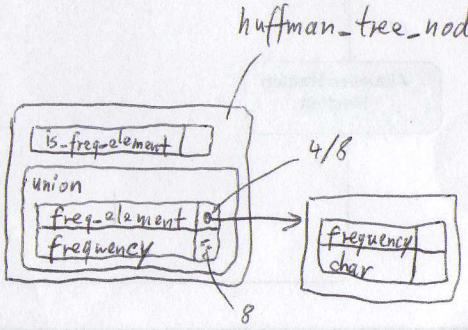


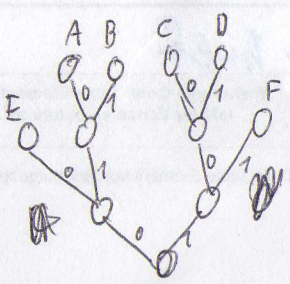


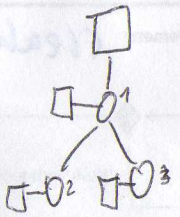


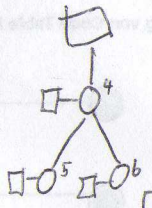


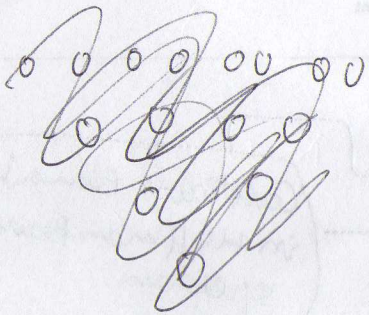


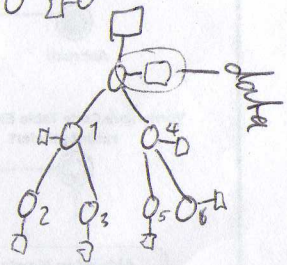






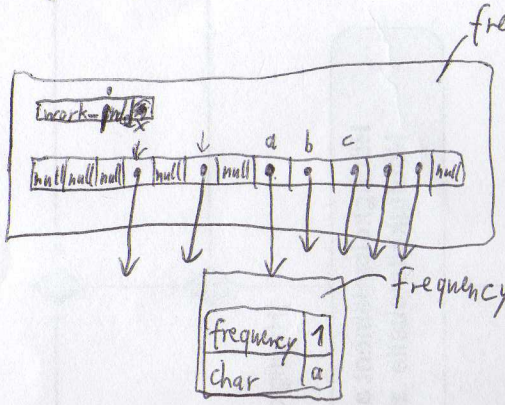






0 1 1 | 0 1 0 | 0 1 0

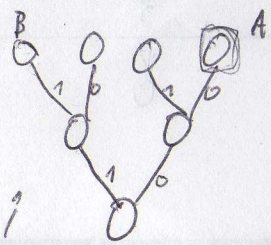
B A A



char* string = "Hallo"

(=)

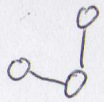
char *string = "Hallo"

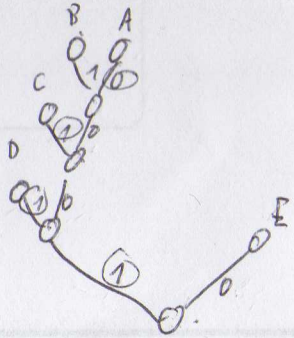


string = "Hallo";

char: A

code: 0





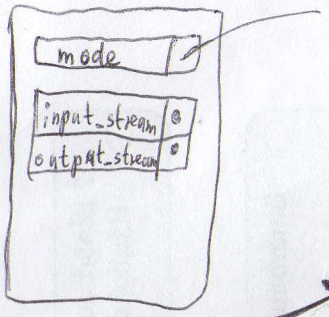
*string = "Halle";

0 0 0 0 0 0 0 0 0 0 1 1 | 0 0 0 0 0 1 0 1

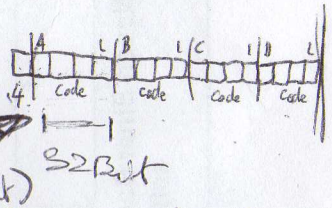
A B

0001

properties

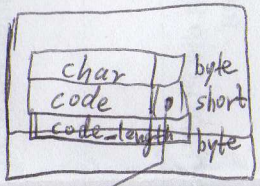


enum: MODE
compress
decompress
manpage

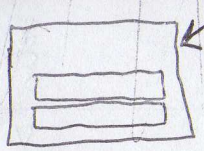


Code Table (CodeTableElement)

code_table_element



code



0 0 1 1 | 0 0 0 0

0 0 1 1 | 0 0 | 0 0

6

010