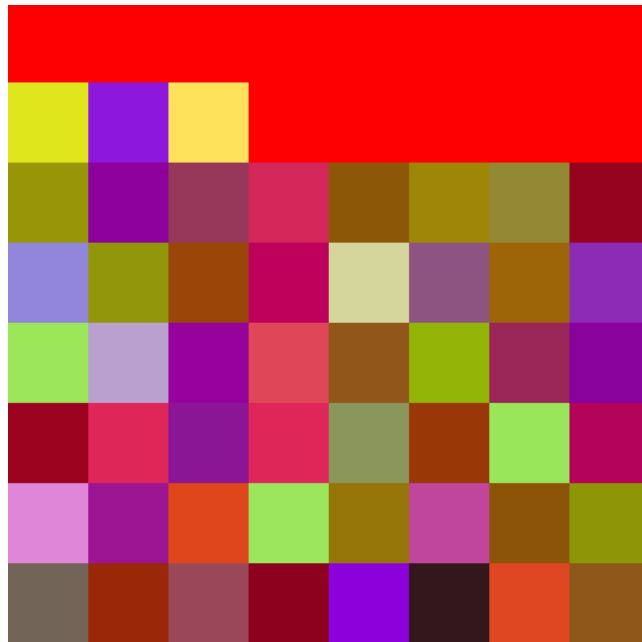


Im Rahmen des Projektes soll ein Programm erstellt werden, dass einen Text in eine Bitmap-Datei (BMP - Datei) umwandelt und filtert.

Text :

```
Wer reitet so spät durch Nacht und Wind? Es ist der
    Vater mit seinem Kind;
    Er hat den Knaben wohn in dem Arm,
    Er faßt ihn sicher, er hält ihn warm.
```

Bitmap:



BMP-Dateien bestehen aus drei Teilen:  
Dem Dateikopf, dem Informationsblock und den Bilddaten.

<b>Dateikopf</b> (BITMAPFILEHEADER)
<b>Informationsblock</b> (BITMAPINFO):
<b>Bitmap-Eigenschaften</b> (BITMAPINFOHEADER)
Eventuell: <b>Farbmasksen</b>
Eventuell: <b>Farbtabelle</b>
Eventuell: Ungenutzter Platz
<b>Bilddaten</b>
Eventuell: Ungenutzter Platz

BITMAPFILEHEADER (Größe: 14 Byte)						
Offset (Byte)		Datentyp		Größe	Name	Inhalt
Dez	Hex	Windows- Style	C-Style			
0	0	WORD	uint16_t	2 Byte	bfType	ASCII-Zeichenkette "BM" (Hex: 0x42 0x4D, Dezimal: 66 77).
2	2	DWORD	uint32_t	4 Byte	bfSize	Größe der BMP-Datei in Byte. (unzuverlässig)
6	6	DWORD	uint32_t	4 Byte	bfReserved	Reserviert, von der Software abhängig, standardmäßig 0
10	A	DWORD	uint32_t	4 Byte	bfOffBits	Offset der Bilddaten in Byte vom Beginn der Datei an. Dieser hat nicht immer den Wert 54 (er ergibt sich aus 14 Byte Header + 40 Byte Infoblock) und muss daher dynamisch ausgelesen werden, weil es sonst ggf. zu Fehldarstellungen im Bild kommt.

BITMAPINFOHEADER (Größe: 40 Byte)						
Offset (Byte)		Datentyp		Größe	Name	Inhalt
Dez	Hex	Windows- Style	C-Style			
14	0E	DWORD	uint32_t	4 Byte	biSize	Größe der BITMAPINFOHEADER-Struktur in Byte
18	12	LONG	int32_t	4 Byte	biWidth	Breite der Bitmap in Pixel. Dabei ist das erste Byte niederwertig und das letzte Byte höchstwertig.
22	16	LONG	int32_t	4 Byte	biHeight	Der Betrag gibt die Höhe der Bitmap in Pixel an. Dabei ist das erste Byte niederwertig und das letzte Byte höchstwertig. <ul style="list-style-type: none"> <li>Ist der Wert positiv, so ist die Bitmap eine sogenannte "bottom-up"-Bitmap (die Bilddaten beginnen mit der untersten und enden mit der obersten Bildzeile). Dies ist die gebräuchlichste Variante.</li> <li>Ist der Wert negativ, so ist die Bitmap eine "top-down"-Bitmap (die Bilddaten beginnen mit der obersten und enden mit der untersten Bildzeile).</li> </ul>
26	1A	WORD	uint16_t	2 Byte	biPlanes	1 (Stand in einigen älteren Formaten wie <b>PCX</b> für die Anzahl der Farbebenen, wird aber für BMP nicht verwendet)
28	1C	WORD	uint16_t	2 Byte	biBitCount	Gibt die Farbtiefe der Bitmap in <b>bpp</b> an; muss einer der folgenden Werte sein: 1, 4, 8, 16, 24 oder 32. Bei 1, 4 und 8 bpp sind die Farben <b>indiziert</b> .
30	1E	DWORD	uint32_t	4 Byte	biCompression	Einer der folgenden Werte: <ul style="list-style-type: none"> <li>0 (<b>BI_RGB</b>): Bilddaten sind unkomprimiert.</li> <li>1 (<b>BI_RLE8</b>): Bilddaten sind lauffängenkodiert für 8 bpp. Nur erlaubt wenn biBitCount=8 und biHeight positiv.</li> <li>2 (<b>BI_RLE4</b>): Bilddaten sind lauffängenkodiert für 4 bpp. Nur erlaubt wenn biBitCount=4 und biHeight positiv.</li> <li>3 (<b>BI_BITFIELDS</b>): Bilddaten sind unkomprimiert und benutzerdefiniert (mittels Farbmasksen) kodiert. Nur erlaubt wenn biBitCount=16 oder 32.</li> </ul>
34	22	DWORD	uint32_t	4 Byte	biSizeImage	<ul style="list-style-type: none"> <li>Wenn <b>biCompression=BI_RGB</b>: Entweder 0 oder die Größe der Bilddaten in Byte.</li> <li>Ansonsten: Größe der Bilddaten in Byte.</li> </ul>
38	26	LONG	int32_t	4 Byte	biXPelsPerMeter	Horizontale Auflösung des Zielausgabegerätes in Pixel pro Meter; wird aber für BMP-Dateien meistens auf 0 gesetzt.
42	2A	LONG	int32_t	4 Byte	biYPelsPerMeter	Vertikale Auflösung des Zielausgabegerätes in Pixel pro Meter; wird aber für BMP-Dateien meistens auf 0 gesetzt.
46	2E	DWORD	uint32_t	4 Byte	biClrUsed	<ul style="list-style-type: none"> <li>Wenn <b>biBitCount=1</b>: 0.</li> <li>Wenn <b>biBitCount=4</b> oder <b>8</b>: die Anzahl der Einträge der Farbtabelle; 0 bedeutet die maximale Anzahl (2, 16 oder 256).</li> <li>Ansonsten: Die Anzahl der Einträge der Farbtabelle (0=keine Farbtabelle). Auch wenn sie in diesem Fall nicht notwendig ist, kann dennoch eine für die <b>Farbquantisierung</b> empfohlene Farbtabelle angegeben werden.</li> </ul>
50	32	DWORD	uint32_t	4 Byte	biClrImportant	<ul style="list-style-type: none"> <li>Wenn <b>biBitCount=1</b>, <b>4</b> oder <b>8</b>: Die Anzahl sämtlicher im Bild verwendeten Farben; 0 bedeutet alle Farben der Farbtabelle.</li> <li>Ansonsten: <ul style="list-style-type: none"> <li>Wenn eine Farbtabelle vorhanden ist und diese sämtliche im Bild verwendeten Farben enthält: deren Anzahl.</li> <li>Ansonsten: 0.</li> </ul> </li> </ul>

Ein Pixel der Bitmap Datei besteht immer aus 3 Byte. Die Bytes stehen für RGB (Rot,Grün,Blau) Werte, also die Farbe des Pixel.

Beispiel Olivgrün: R=85,G=107,B=47

Um den Text in eine Bitmap Datei zu codieren, müssen also immer drei aufeinanderfolgende Bytes (ASCII Code) innerhalb des Textes in jeweils einen RGB Pixel umgewandelt werden.

Beispiel :

Text = Abc

ASCII Codes : A = 56 ; b = 98 ; c = 99

Pixel RGB = (65 , 98 , 99)

Der Text soll immer in eine quadratische Bitmap Datei umgewandelt werden. Dazu ist es unter Umständen nötig, fehlende Bytes durch Bytes mit dem Wert 0 aufzufüllen.

Folgende Annahmen für die Initialisierung der Werte im Informationsblock können getroffen werden:

biSize = muss berechnet werden

biWidth = biHeight = muss berechnet werden

biPlanes = 1

biBitCount = 24

alle weiteren Elemente können auf 0 gesetzt werden.

Um die Verschlüsselung des Textes zu verbessern, müssen die einzelnen Pixel noch mit einem Filter versehen werden. Der Filter soll die einzelnen RGB Werte so Transformieren das eine Rücktransformation möglich ist.

Die Filter sollen folgende Transformationen durchführen :

- Der rot Anteil eines Pixel soll binär invertiert werden.
- In dem grün Anteil eines Pixels sollen die oberen 4 Bit mit den unteren 4 Bit vertauscht werden.
- Im blau Anteil sollen die Pixel um 2 Bit nach rechts im Kreis rotiert werden, so dass die beiden Pixel die am Anfang raus fallen an das Ende gehängt werden:

Aufgabe:

1. Lesen Sie die gelieferte Textdatei ein. Der Pfad der Datei soll über die Kommandozeile übergeben werden.
2. Erstellen Sie ein Bitmap und initialisieren Sie den Dateikopf und den Informationsblock.
3. Wandeln Sie den Text in entsprechende RGB Pixel.
4. Filtern Sie die RGB Pixel nach den oben beschriebenen Vorgaben.
5. Speichern Sie die BMP Datei, wobei der Name der zu erstellenden Datei ebenfalls über die Kommandozeile übergeben werden soll.

Hinweise:

- Versuchen Sie so weit als möglich alle Themengebiete die in der Vorlesung behandelt wurden in dem Projekt abzudecken. Ausgenommen sind folgende Einzeltechniken:
  - mehrdimensional Arrays
  - union/enum
  - Fließkomma Zahlen
  - Printf
  - Rekursionen
- Unter Umständen ist es Hilfreich die Speicherausrichtung von Strukturen auf 1 Byte festzulegen