Aufgabe 1. Mit den bisherigen Aufgaben fertig werden.

Aufgabe 2. a) Implementiere eine c-Datei zu folgender Header-Datei:

```c
/* gibt die Länge eines Strings zurück */
int str_len(char *s);

/* gibt 0 zurück, wenn zwei strings gleich
   sind und 1 sonst */
int str_cmp(char *s1, char *s2);

/* kopiert s nach d und gibt d zurück */
char *str_cpy(char *d, char *s);

/* hänge s2 and s1 an und gib s1 zurück */
char *str_cat(char* s1, char* s2)

/* allokiere neuen Speicher für eine Kopie von
   s, kopiere s dorthin und liefere einen Pointer
   auf den neuen String. */
char *str_dup(char *s);
```

b) Implementiere nun noch folgende String-Funktionen, die man in der Praxis *niemals* benutzen würde:

```c
/* schreibt s rückwärts in s und gib es zurück */
char *str_reverse(char *s);

/* gibt 1 zurück, wenn ein String ein Palindrom
   ist und 0 sonst */
int str_ispalin(char *s);

/* hänge s2 an s1 an und stelle sicher, dass
   dazu genug Speicher in s1 zur verfügung steht */
char *str_smartcat(char *s1, char *s2)

/* verkleinert den Speicher auf den s zeigt
   auf die Länge von s */
char *str_compress(char *s);
```

Aufgabe 3. Implementiere folgende Funktion (ins gleiche Modul), die zu einem gegebenen String einen längsten Teilstring findet, der ein Palindrom ist. Speichere diesen Teilstring wieder in s und gib s zurück.

```
1 /* Beschreibung, selber machen */
2 char *str_glsp(char *s);
```

Das Problem ist in polynomieller Laufzeit lösbar und auf die Idee kann man auch kommen.

Aufgabe 4. Implementiere den Bucket-Sort Algorithmus: Unter der Annahme, dass die zu sortierenden Daten aus einem endlichen Wertebereich stammen, kann man solche Arrays sehr schnell sortieren.

Wir betrachten hier ein `unsigned short`-Array A. Bucketsort besteht nun aus zwei Durchläufen:

a) Sei n das Maximum aus A, erstelle dann ein `unsigned`-Array B mit $n+1$ Elementen und sorge dafür, dass an der i-ten Stelle steht, wie oft i in A vorkommt.

b) Wir überschreiben im zweiten Durchlauf A, indem wir für wachsendes k die Zahl k genau $B[k]$ mal nach A schreiben.