

Dieser Zettel ist bewusst sehr umfangreich und wird dich vermutlich mehr als einen Tag beschäftigen.

Aufgabe 1. Installiere die `joelixblas` Bibliothek und mache dich mit dem Manual vertraut.

Implementiere das Beispiel “Vektor erstellen” aus Kapitel 6.1; kompiliere und teste es.

Aufgabe 2. Schreibe mit Hilfe der `joelixblas` Bibliothek ein Programm, welches die Einheitsmatrix erstellt und auf der Konsole ausgibt.

Nutze hierzu den Datentyp `Joelix_sMatrix` und die Funktionen `joelix_smatrix_init` und `joelix_smatrix_fuelleZeile`, wie sie im Manual beschrieben sind. Siehe hierzu auch das Beispiel in Kapitel 6.2.

Aufgabe 3. Implementiere das CG-Verfahren um die Lösung x für das lineare Gleichungssystem $Ax = b$ zu lösen. Dabei ist A eine gegebene symmetrische, positiv definite, $n \times n$ Matrix und b ein gegebener n -dimensionaler Vektor. Benutze hierzu die von der `joelixblas` Bibliothek bereitgestellten Datenstrukturen und Funktionen.

Tip: RTFM!

Algorithm 2.1: CG (Matrix A , Vektor x , Vektor b , double ϵ)

```
1  $x_0 \leftarrow 0$ ;  
2  $r_0 \leftarrow b$ ;  
3  $d_0 \leftarrow r_0$ ;  
4  $res \leftarrow 1$ ;  
5  $maxit \leftarrow 1000$ ;  
6  $k \leftarrow 0$ ;  
7 while  $res > \epsilon$  and  $k < maxit$  do  
8    $z \leftarrow Ad_k$ ;  
9    $\alpha_k \leftarrow \frac{r_k \cdot r_k}{d_k \cdot z}$ ;  
10   $x_{k+1} = x_k + \alpha_k d_k$ ;  
11   $r_{k+1} = r_k - \alpha_k z$ ;  
12   $\beta_k = \frac{r_{k+1} \cdot r_{k+1}}{r_k \cdot r_k}$ ;  
13   $d_{k+1} \leftarrow r_{k+1} + \beta_k d_k$ ;  
14   $res \leftarrow \|r_{k+1}\|$ ;  
15   $k++$ ;  
16  $x \leftarrow x_k$ ;
```

Aufgabe 4. Implementiere einen Löser für das Poissonproblem

$$-u'' = g \text{ in } (0, 1), \quad (1)$$

$$u(0) = T_0, \quad (2)$$

$$u(1) = T_0. \quad (3)$$

Stelle dazu abhängig von einer gegebenen Funktion

```
1 double g (double x);
```

das Gleichungssystem

$$A\hat{u} = b \quad (4)$$

aus der Vorlesung auf und löse es mit Hilfe des CG-Verfahrens.

Gebe den Lösungsvektor \hat{u} dann in eine Datei Namens `poissonout.txt` aus (Tip: vielleicht gibt es dafür ja auch eine Funktion in der `joelixblas` Bibliothek).

Benutze dann das von uns zur Verfügung gestellte Shell-skript, um deine Lösung als Graph sichtbar zu machen.