

**Programmierung.**

- 19) Programmiere die “transponierte” Routine zu `prolongate`:

```
void restrict (const Mesh& fine_mesh, const Mesh& coarse_mesh,
              const Vector& fine_res, Vector& coarse_res);
```

Sobald die Routinen `prolongate` und `restrict` laufen, ändere die Routinen (und evtl. die Schnittstellen), sodass die Dirichlet-Randbedingung korrekt behandelt werden.

- 20) Stelle die Grobgittermatrix  $K_{\text{coarse}}$  auf, welche durch

$$(\underline{w}, K_{\text{coarse}} \underline{v})_{\ell^2} = (R_0^\top \underline{w}, K_{\text{fine}} R_0^\top \underline{v})_{\ell^2}$$

definiert ist.

- 21) Erweitere den überlappenden Schwarz-Vorkonditionierer aus Bsp. 16 um den Grobgitterlöser.

Untersuche die Anzahl der Iterationen sowie die geschätzte Konditionszahl im Vergleich zu Bsp. 17.

1024 Elemente, 64 ca. gleich große Blöcke

Overlap	Blockgröße	Grobgitterknoten	PCG-Iterationen	Konditionszahl
2	ca. 17	4		
2	ca. 17	8		
2	ca. 17	16		
2	ca. 17	32		
2	ca. 17	64		
4	ca. 19	16		
4	ca. 19	32		
4	ca. 19	64		
8	ca. 23	16		
8	ca. 23	32		
8	ca. 31	64		