Wie bist du vorgegangen bei CorC?

GruppeA

Vor/Nachbedingung -> Statements

Code schreiben (Boxen für Statements) -> Conditions

Beim UseCase angefangen und dann immer verfeinert

Schrittweise Konstruktion des Algorithmus -> Anpassen einzelner Statements

Statement runterbrechen -> unklare Baumtiefe führt zu keiner Lösung

Beweis Stück für Stück aufgebaut, Conditions direkt mit zum Code

GruppeB

Ablauf überlegen -> aufteilen, paar Mal alles im Prinzip von vorne

Raten, wie Algorithmus funktioniert -> bauen -> Conditions finden

Struktur des Programms entwerfen -> Bedingungen formulieren -> Code schreiben -> Verifizieren

Schleife gebaut -> Statements gebaut

Wie bist du vorgegangen bei KeY?

GruppeA

Programm -> Verifikation

Code -> Loopinvariante mit decreases -> Verbesserungen

Algorithmus -> Invarianten hinzugefügt

Algorithmus -> Trial and Error bei Loopinvariante

Rechnung versucht an Nachbedingung abzuleiten

Algorithmus implementiert -> Schleifeninvariante geschrieben und getestet

GruppeB

Programm -> Schleifeninvariante setzen

Coden -> Postcondition anschauen -> Invariante raten

Struktur des Programms entwerfen -> Code schreiben -> Verifizieren -> Bedingungen anpassen -> Loop Verifizieren

Schleife gebaut -> Variable entfernt