A.2:

Argumentieren Sie, weshalb der Algorithmus $insertion\_sort$ korrekt ist. Weisen Sie dazu nach, dass für i ≥ 1 die Schleifeninvariante

> $Inv(i)$: Vor dem i–ten Durchlauf durch den Rumpf der for-Schleife besteht $a[0:i]$ aus den ursprünglichen Elementen $a[0:i]$, aber in sortierter Reihenfolge.

gilt. Zerlegen Sie Ihre Argumentation in folgende drei Schritte:

1. Initialisierung: Es gilt $Inv(1)$ vor der ersten Ausführung des Rumpfes der for-Schleife.

- Aussage ist trivial, da $a[0:1]$ nur ein Element enthält

2. Invarianz: Falls für ein i ≥ 1 die Invariante $Inv(i)$ gilt, dann gilt auch $Inv(i\ +\ 1)$.

- Wenn Invariante $Inv(i)$ gilt, ist der Teil der Liste vor dem i-ten Element sortiert.
- Bei $Inv(i + 1)$ ist das i-te Element bereits sortiert, da $Inv(i)$ gilt

3. Terminierung: Wenn die for-Schleife terminiert ($i\ =\ m$), können wir aus $Inv(i)$ auf die Korrektheit des Algorithmus schließen.

- Korrektheit wurde durch vollständige Induktion bewiesen
- Schritt 1 stellt Induktionsanfang dar
- Schritt 2 stellt Induktionsschritt da
- Somit gilt $Inv(i)$ für alle $i \geq 1$