

8.5)

```
(1) z=0;  
(2) b=true;  
(3) do  
(4) {  
(5)     z=z+1;  
(6)     b=!b;  
(7) }  
(8) while(z != x);
```

Bedingungen: $x > 0$, x , z (Ganzzahlen), b (Wahrheitswert)

Anmerkung: Im weiteren Text setze ich für Gleichheit einfache =-Zeichen.

Klären wir zuerst, warum das Programm terminiert. Zumindest nach hinzufügen des Semikolons in Funktionszeile (8) ist eine Terminierung möglich, da sonst ein Syntaxfehler auftritt.

Wie in den Bedingungen geklärt ist x ein ganzzahliger INT-Wert größer Null, also eins und größer. Betrachten wir nun den Ausführungsparameter $z \neq x$, sehen wir, dass die Schleife mindestens einmal ausgeführt wird, da es eine DO-WHILE-Schleife ist, und sie erst dann abbricht, wenn z gleich x ist. Innerhalb der Schleife wird z , welches mit 0 initiiert wurde, pro Schleifendurchlauf inkrementiert und der Wahrheitswert von b negiert. Durch das inkrementieren von z innerhalb der Schleife läuft z den gesamten positiven Wertebereich größer 0 ab. Da x ebenfalls größer 0 ist, müssen beide INT-Werte zwangsläufig irgendwann den selben Zahlenwert annehmen und somit muss die Schleife irgendwann terminieren.

Kommen wir nun zur These, dass $(b = \text{true} \Leftrightarrow x \text{ ist gerade})$ ist. Hierzu hilft wieder eine Betrachtung der Schleifenfunktionalität. Nach dem ersten Schleifendurchlauf hat z den Wert 1 und b den Wert false. Beim nächsten Durchlauf wird z erneut inkrementiert und b negiert. Somit hat z den Wert 2 und b den Wert true.

Da die Negation eines Wahrheitswertes äquivalent einer Wertangabe modulo 2 (Restwertangabe nach Teilen durch 2, also 0 oder 1) ist, sieht man, dass sich induktiv ergibt, dass wenn gilt $z = 2*n$ ist auch $b = \text{true}$, da $b = \text{true}$ bei $z = 0, 2, \dots$ und ebenso, wenn gilt $z = 2*n + 1$ ist $b = \text{false}$, da b bei $z = 1, \dots$ false ist. Da nun wiederum die Schleife nur terminiert, wenn z gleich x ist, gilt, dass gilt:

- $z = 2*n = x \Leftrightarrow b = \text{true}$
- $z = 2*n + 1 = x \Leftrightarrow b = \text{false}$

q.e.d.