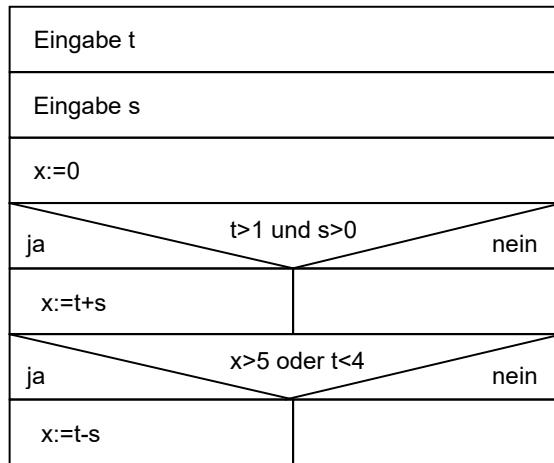


Aufgabenblatt 9

Aufgabe 1

Gegeben sei folgendes Programm in Form eines Struktogramms:



Geben Sie eine minimale Anzahl an Testfällen an, um vollständige Pfadüberdeckung zu erreichen.

Geben Sie die jeweiligen Testfälle in der Form (i, j) an, falls die Eingabeparameter t und s mit den konkreten Zahlenwerten i und j zu belegen sind, wobei $i, j \in \{0, 1, 2, 3, 4, 5\}$.

Aufgabe 2

Folgende Java-Methode überprüft, ob eine nicht-leere Zeichenkette namens source eine nicht-leere Zeichenkette namens s vollständig enthält. Ihr Rückgabewert ist wie folgt spezifiziert:

- i, falls i die Startposition des ersten Auftretens der Zeichenkette s innerhalb der Zeichenkette source ist,
- -1, falls Zeichenkette s mehr Zeichen als Zeichenkette source enthält,
- -2, sonst.

```
public static int isSubString(String s, String source) {  
    int result = -2;  
    if (s.length() > source.length())  
        result = -1;  
  
    int i = 0;  
    boolean prefixFound = false;  
    int diff = source.length() - s.length();  
  
    while (!prefixFound && i <= diff) {  
        int j = 0;  
        prefixFound = true;  
        while (prefixFound && j < s.length()) {  
            if (s.charAt(j) != source.charAt(i + j))  
                prefixFound = false;  
            j = j + 1;  
        }  
        if (prefixFound)  
            result = i;  
        i = i + 1;  
    }  
    return result;  
}
```

Teilaufgabe a

Geben Sie zum obigen Code den dazugehörigen Kontrollflussgraphen an.

Teilaufgabe b

Geben Sie eine Testdatenmenge mit einer minimalen Anzahl von Elementen an, die eine vollständige Verzweigungsüberdeckung erreicht. Begründen Sie, warum eine solche Menge einen Testfall (s, source) mit s.length() < source.length() enthalten muss.