

# 1 Dynamischer Test: White-Box

## 1.1 Code-Überdeckung

Gegeben sei der folgende abstrakte Code einer (nicht notwendigerweise sinnvollen) Methode als Pseudocode

```
1 public int h (int x, y; boolean b1){
2     int res = 10;
3     boolean b2 = false;
4     for (i = 1; i < 5; i=i+2){
5         if (x < y) {
6             i = i + 5;
7             res = res + x;
8         }
9         else {
10            res = 0;
11        };
12        if (res < 10) {
13            res = res + 1;
14            b2 = true;
15        }
16    }
17    if ((res < 10) and b1 and b2) {
18        res = res * res;
19    }
20    else {
21        println{'ELSE-Teil'};
22        res = 1;
23    };
24    println('final result:' res);
25    return res;
26 }
```

$h(7, 10, true)$

$i = 6$   
 $res = 10 + 7$

$if (17 < 10) // false$

$(17 < 10) // false$

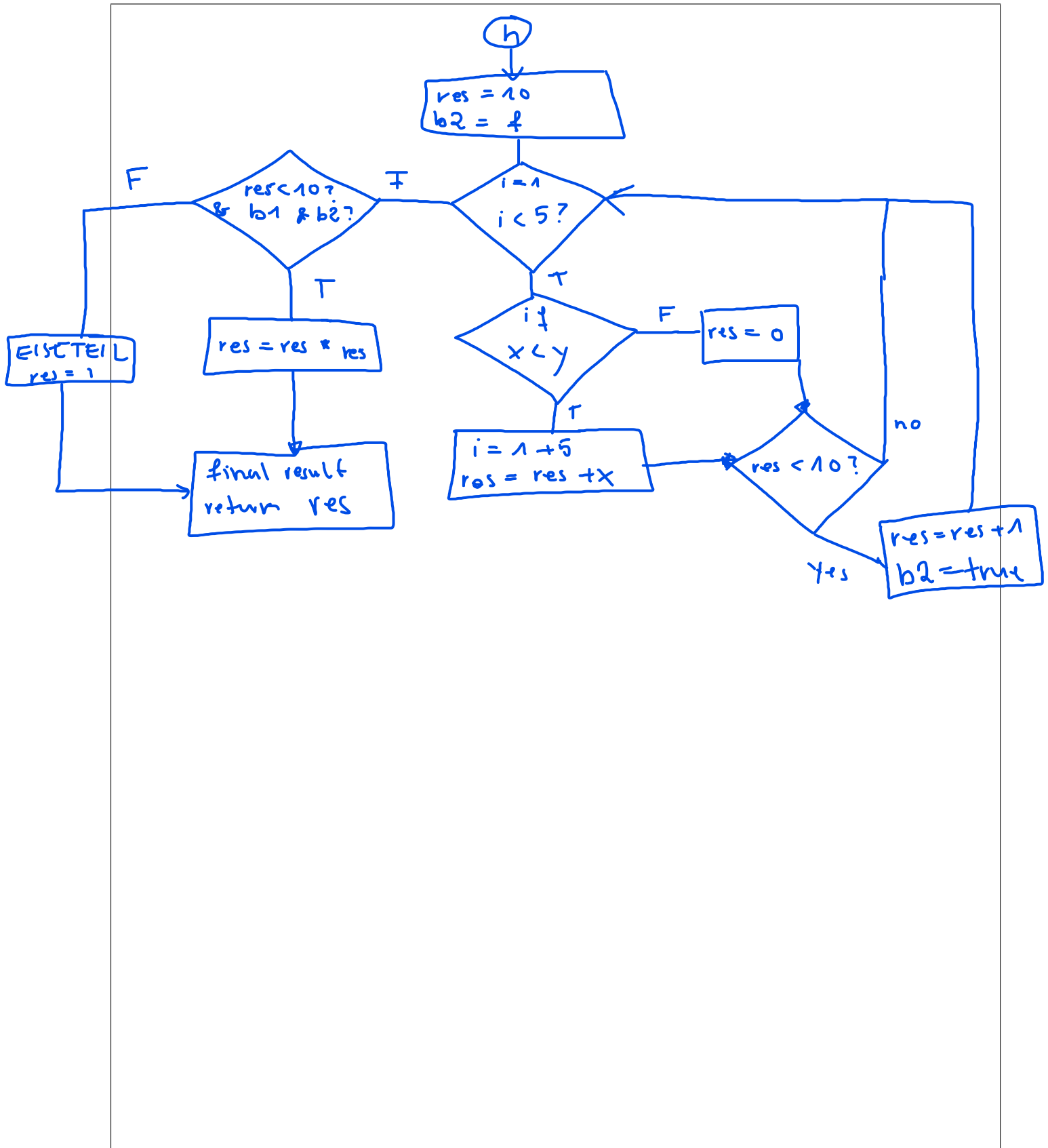
print  
 $res = 1$

"final result : 1"

return 1

Bearbeiten Sie die folgenden Teilaufgaben

a) Bestimmen Sie zunächst den Kontrollflussgraphen für Methode h.



<b>WP-CT WiSe 22</b>	<b>Praktikum Aufgabenblatt 3 Whitebox Coverage (3.1)</b>	<b>Prof. Dr. B. Buth 15.11.2022</b>
--------------------------	--	---

b) Bestimmen Sie die Anweisungsüberdeckung zu folgendem Testfall:

- TF 1:  $x == 7$ ;  $y = 10$ ;  $b1 == \text{true}$  (Aufruf von  $h(7,10, \text{true})$ )

	Folge der durchlaufenen Zeilen (Zeilennr)
TF 1	
Anzahl Anweisungen gesamt in $h$	19
Anweisungsüberdeckung durch TF 1 als Bruch	$19/22 \approx 86\%$

Anmerkung: bei den Anweisungen zählen der Methodenkopf (Startknoten) und die schließende Klammer des Rumpfs (Endknoten) auch als je 1 Anweisung.

c) Geben Sie weitere Testfälle die zusammen mit TF 1 eine 100% -ige Entscheidungsüberdeckung sicherzustellen.

	x	y	b1	durchlaufene Folge von Zeilen
TFZ 1	0	0	+	1, 2, 3, 4, 5, 8, 10, 12, 13, 14 15, 16, 17, 18, 19, 21, 25, 26 res = 1
TFZ 2	0	0	f	1, 2, 3, 4, 5, 8, 10, 12, 13, 14 15, 16, 17, 21, 22, 23, 24, 25, 26
TFZ 3	10	5	f	siehe TFZ 1
TFZ 4	10	5	T	siehe TFZ 2
TFZ 5	1	0	f	1-8, 12, 15, 16, 17, 19, 21-26

Anmerkung: unter Umständen werden auch weniger als die vorgesehenen Testfälle reichen.

- d) Ist Ihrer Meinung nach eine 100 %ige Pfadüberdeckung (gegenüber den Pfaden im Kontrollflussgraphen) für diesen Code erreichbar? Begründen Sie Ihre Aussage.

100 % möglich? (Ja / Nein):

Begründung:

Durch die else Zweige in 5 und 19  
gibt es immer Pfade die nicht ausgeführt  
werden können wenn  $(x < y) == \text{true} / \text{false}$   
und  $(res < 10) \&\& b1 \&\& b2 == + / \text{false}$ .  
Man entscheidet man sich für bestimmte Parameter  
sind die anderen Pfade durch die if else Bedingung  
unerreichbar während der Laufzeit.