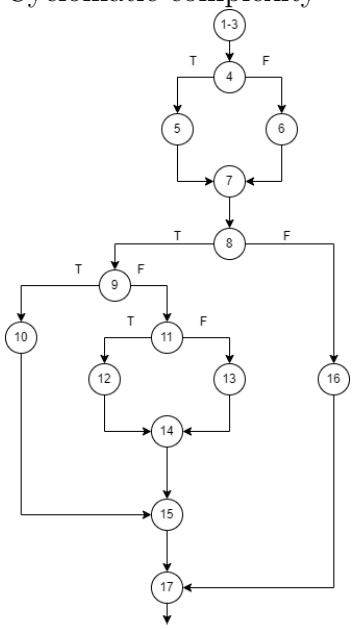
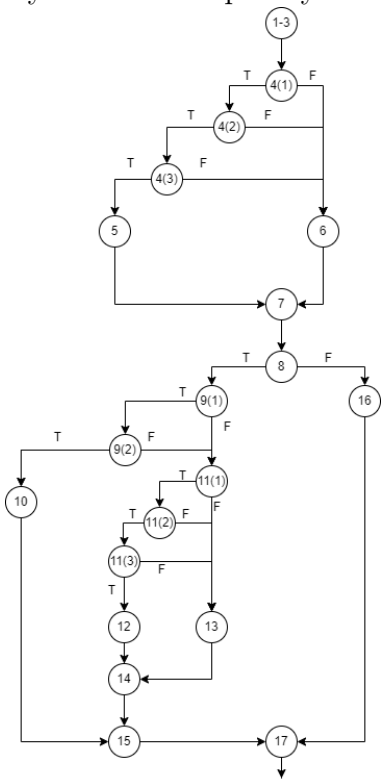
Q1

a) Compound decisions are treated *en bloc*
Cyclomatic complexity = 5

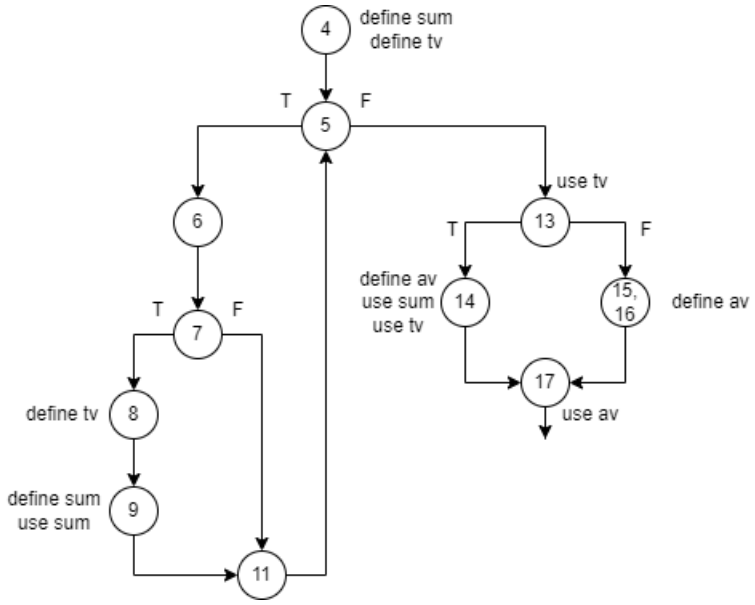


b) Compound decisions are treated separately
Cyclomatic complexity = 10



Q2

```
1  public static double ReturnAverage(int value[], int AS, int MIN, int MAX) {
2      int i, ti, tv, sum;
3      double av;
4      i = 0; ti = 0; tv = 0; sum = 0;
5      while (ti < AS && value[i] != -999) {
6          ti++;
7          if(value[i] >= MIN && value[i] <= MAX) {
8              tv++;
9              sum = sum + value[i];
10         }
11         i++;
12     }
13     if (tv > 0)
14         av = (double) sum/tv;
15     else
16         av = (double) -999;
17     return (av);
18 }
```



Some *def-clear* paths are as following:

For tv:

1. $(4 \rightarrow 5 \rightarrow 6 \rightarrow 7)$
2. $(8 \rightarrow 9 \rightarrow 11 \rightarrow 5 \rightarrow 13)$
3. $(8 \rightarrow 9 \rightarrow 11 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 11 \rightarrow 5 \rightarrow 13)$

4. $(4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 11 \rightarrow 5 \rightarrow 13)$

For av:

1. $(14 \rightarrow 17)$
2. $(16 \rightarrow 17)$

For sum:

1. $(4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8)$
2. $(4 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 11 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 8)$
3. $(9 \rightarrow 11 \rightarrow 5 \rightarrow 6 \rightarrow 7 \rightarrow 11 \rightarrow 5 \rightarrow 13 \rightarrow 14)$

Q3

```
1  def testStrategies(a, b):
2      x = 0
3      y = 0
4      if a < b:
5          return x+y
6      else:
7          return y-x
```

Q4

Test #	a	b	c	d	Expected
1	T	F	T	F	T
2	T	F	F	T	T
3	F	F	F	F	F
4	T	T	T	T	F