Assignment 3: WCET Analysis

Consider the following piece of code:

```
A: i = 0
B: s = 0
C: while i < 100 do
D:   if a[i] > 0 then
E:     s = s + a[i]
     else
F:     s = s - a[i]
     end
G:   if a[i] > 2 then
H:     s = s*2
     end
I:   i = i + 2
   end
```

1. Draw a control-flow graph for the code!
2. Give a tight upper bound for the nmber of loop iterations! Explain how you achieved it.
3. Is there any infeasible path? If yes, which?
4. Assume that we have the following local WCETs (in machine cycles) for the basic blocks in the control-flow graph: $t_A = t_B = 5$, $t_C = 7$, $t_D = 15$, $t_E = 8$, $t_F = 10$, $t_G = 15$, $t_H = 14$, and $t_I = 5$. Calculate a WCET bound for the code through tree-based calculation!
5. Now calculate a WCET bound using path-based calculation! Try to utilize any infeasible path information that might be present. Is the new WCET bound smaller or larger than the one calculated with the tree-based method?