A language \( \mathcal{L} \) has the following abstract syntax, where \( c \) ranges over integer constants, \( x \) ranges over a set of variables and \( \oplus \) ranges over binary operations:

\[
e = c \mid x \mid \lambda x.e \mid e_1 e_2 \mid \text{let } x = e_1 \text{ in } e_2 \mid \text{if } e_1 \text{ then } e_2 \text{ else } e_3 \mid e_1 \oplus e_2
\]

Consider the following program \( P \) in \( \mathcal{L} \):

\[
\text{let } x = 5 \text{ in } \\
\quad \text{let } f = \lambda x.2 \ast x \text{ in } \\
\quad \quad \text{if } x > 0 \text{ then } f \ x \text{ else } f (0 - x)
\]

This question asks you to perform 0CFA on \( P \).

(a) Draw the program \( P \) as a tree and label its program points. [4 marks]

(b) Give the space of flow values for \( P \). [2 marks]

(c) Each program point \( i \) in \( P \) has an associated flow variable \( \alpha_i \). Show the initial constraints on each \( \alpha_i \) that are generated when performing 0CFA. [4 marks]

(d) Show how the process of solving the constraints from part (c) leads to additional constraints being generated. [4 marks]

(e) Show the final solution after solving all constraints from parts (c) and (d) and simplifying binary terms. [4 marks]

(f) Explain whether your answer is a safe over- or under-approximation of the result of \( P \) and where the imprecision comes from. [2 marks]