Three alternative representations for non-negative integers, $n$, are:

- **Peano**: values have the form $S(... S(Z) ...)$, applying $S$ $n$ times to $Z$ where $S$ and $Z$ are constructors or constants of some data type.

- **Binary**: values are of type `bool list` with 0 being represented as the empty list, and the least-significant bit being stored in the head of the list.

- **Church**: values have the form $\text{fn } f => \text{fn } x => f(... f(x) ...)$, applying $f$ $n$ times to $x$

(a) Write ML functions for each of these data types which take the representation of an integer $n$ as argument and return $n$ as an ML `int`.  [6 marks]

(b) Write ML functions for each of these data types which take representations of integers $m$ and $n$ and return the representation of $m + n$. Your answers must not use any value or operation on type `int` or `real`.  [Hint: you might it useful to write a function `majority`: `bool*bool*bool -> bool` (which returns true when two or more of its arguments are true) and to note that the ML inequality operator ‘<>’ acts as exclusive-or on `bool`.]  [10 marks]

(c) Letting `two` and `three` respectively be the Church representations of integers 2 and 3, indicate whether each of the following ML expressions give a Church representation of some integer and, if so what integer is represented, and if not giving a one-line reason.

(i) `two three`

(ii) `three two`

(iii) `two o three`

(iv) `three o two`

[4 marks]