Foundations of Computer Science

This question has been translated from Standard ML to OCaml

To represent the power series $\sum_{i=0}^{\infty} a_i x^i$ in a computer amounts to representing the coefficients $a_0, a_1, a_2, \ldots$ One possible representation is by a function of type `int -> real` that returns the coefficient $a_i$ given $i$ as an argument. An alternative representation is the following type:

```
type power = Cons of float * (unit -> power)
```

(a) Demonstrate the two representations by using each of them to implement these two power series:

(i) The constant power series $c$, with $a_0 = c$ and $a_i = 0$ for $i > 0$. [3 marks]

(ii) The Taylor series $\sum_{i=0}^{\infty} x^i/i!$ for the exponential function. [4 marks]

(b) Also implement (using both representations) each of the following operations on power series:

(i) Product with a scalar, given by $c \cdot (\sum_{i=0}^{\infty} a_i x^i) = \sum_{i=0}^{\infty} (ca_i) x^i$. [3 marks]

(ii) Sum, given by $(\sum_{i=0}^{\infty} a_i x^i) + (\sum_{i=0}^{\infty} b_i x^i) = \sum_{i=0}^{\infty} (a_i + b_i) x^i$. [4 marks]

(iii) The product $(\sum_{i=0}^{\infty} a_i x^i) \times (\sum_{i=0}^{\infty} b_i x^i)$, where the $i$th coefficient of the result is $a_0 b_i + a_1 b_{i-1} + \cdots + a_i b_0$. [6 marks]

You may assume the OCaml function `float_of_int` of type `int -> float` that maps an integer to the equivalent real number.