CPS 710 HL OPERATIONS F2007

There are 6 operators in HL: +, -, *, /, ^, and evaluation expressed with []. They are overloaded as described in the sections below. They all work with HL ints and polyns as follows.

ADDITION, SUBTRACTION AND MULTIPLICATION

When two ints are combined with one of these operations the result is an int.
When two polyns, or one int and one polyn are combined with +, -, or *, the result is a polyn.

DIVISION

Since HL does not have any number types other than ints all divisions involving two ints is an integer division: the result is the quotient of that division.

Because this is only an assignment and not a real system, we will restrict divisions with polynomials to be divisions where the dividend is a monomial: i.e. the dividend is of the form ax^b. This includes integers which are of the form ax^0.

- Dividing a polyn by an int divides all the coefficients of that polynomial by the int. Division by zero will throw an exception in assignment 4.
- Dividing a polyn by a monomial where the power of x is not zero divides each term of that polynomial by the monomial.

For example, if a = $3x^2+10x+1$ then
a/x is the polyn $3x+10+1x^{-1}$
a/(3x) is the polyn x+3 (because of the integer divisions)
a/($5x^2$) is the polyn $2x^{-1}$ (again, integer division nullifies some of the terms)

EXPONENTIATION

Generally speaking HL exponentiation works like mathematical exponentiation. The values of exponents can only be ints. So even though exponents can be specified as complicated expressions they must evaluate to ints.

- For any value a, $a^0=1$ and $a^1=a$. Also $0^0 = 1$.
- The exponentiation of a simple monomial (including ints) is calculated using general exponentiation rules. Exponentiations of polynomials with more than one term are not allowed.

POLYNOMIAL EVALUATION

The a[b] evaluation operator works as follows:

- b must be an int
- if a is an int, then a[b] = a;
- if a is a polyn, then a[b] is the int calculated by replacing all occurences of x in a by b. Negative powers of any integer will evaluate to 0.

RULES OF EVALUATION

The operations just described follow these precedence rules:

Highest	( )	^	[ ]	* /	+ -	Lowest

All the HL operations expressed with these operators are left-associative.