Polynomials is an important application of arrays and linked lists. A polynomial is composed of different terms where each of them holds a coefficient and an exponent. A polynomial p(x) is the expression in variable x which is in the form (anxn + an-1xn-1 + .... + a1x+ a0), where an, an-1, ...., a0 fall in the category of real numbers and 'n' is the non-negative integer, which is called the degree of polynomial.

Example: 10x2 + 26x, here 10 and 26 are coefficients and 2, 1 is its exponential value.

Points to keep in Mind while working with Polynomials:  
- The sign of each coefficient and exponent is stored within the coefficient and the exponent itself.  
- The storage allocation for each term in the polynomial must be done in descending order of their exponent.

In this question, complete SLinkedList class is included. You should use this class to complete your Polynomial class with initialized frame as following. This task is implement insertTerm to insert a term into a polynomial.

class Polynomial;  
class Term {  
private:  
    double coeff;     
    int exp;  
    friend class Polynomial;  
public:  
    Term(double coeff = 0.0, int exp = 0) {  
        this->coeff = coeff;  
        this->exp = exp;  
    }  
    bool operator==(const Term& rhs) const {  
        return this->coeff == rhs.coeff && this->exp == rhs.exp;  
    }  
    friend ostream & operator<<(ostream &os, const Term& term) {  
        cout << endl;  
        cout << "Term: " << "(" << term.coeff << " " << term.exp << ")";  
        return os;  
    }  
};  
class Polynomial {  
private:  
    SLinkedList<Term>\* terms;  
public:  
    Polynomial() {  
        this->terms = new SLinkedList<Term>();  
    }  
    ~Polynomial() {  
        this->terms->clear();  
    }  
    void insertTerm(const Term& term);  
    void insertTerm(double coeff, int exp);  
    void print() {  
        SLinkedList<Term>::Iterator it;  
        cout << "[";  
        for (it = this->terms->begin(); it != this->terms->end(); it++) {  
            cout << (\*it);  
        }  
        cout << endl << "]";  
    }  
};

**For example:**

| **Test** | **Result** |
| --- | --- |
| Polynomial \*poly = new Polynomial();  poly->insertTerm(6.0, 2);  poly->insertTerm(4.0, 5);  poly->insertTerm(4.0, 3);  poly->insertTerm(6.0, 5);  poly->insertTerm(-1.0, 0);  poly->insertTerm(-6.0, 6);  poly->insertTerm(6.0, 6);  poly->print(); | [  Term: (10 5)  Term: (4 3)  Term: (6 2)  Term: (-1 0)  ] |
| int n = 5;  int coeff[] = { 1, 2, 3, 4, 5 };  int exp[] = { 1, 2, 3, 4, 5 };  Polynomial p1;  for (int i = 0; i < n; ++i)  p1.insertTerm(coeff[i], exp[i]);  p1.print(); | [  Term: (5 5)  Term: (4 4)  Term: (3 3)  Term: (2 2)  Term: (1 1)  ] |