To save memory space, a polynomial may be represented as a linked list where each node contains the **coefficient** and **exponent** of a term of the polynomial. For example, the polynomial 4x3 + 3x2 - 5, would be represented as

shown at right. Write a program that works with polynomials whose coefficients and exponents are integers using linked lists, where each node in the list can be represented as in the following example:

**typedef struct {**

**int coef;**

**int exp;**

**} elementtype;**

**typedef struct node {**

**elementtype term;**

**struct node \* next;**

**} node;**

The program continuously interacts with the user through the sequence of commands described in the table below, until the command \*\*\* is encountered. The parameters have the following meanings:

* <LIST\_ID> are integers 1, 2, 3, 4 indicating the first polynomial, the second polynomial, the sum polynomial of polynomials 1 and 2, and the product polynomial of polynomials 1 and 2.
* coef and exp are integer values.

|  |  |  |
| --- | --- | --- |
| **Index** | **Command** | **Description** |
| 1 | insert < LIST\_ID > <coef> | Insert an element with coefficient coef and |
|  | <exp> | exponent exp as a node **at the beginning** of the |
|  |  | list corresponding to the LIST\_ID polynomial of |
|  |  | **1 or 2**. It should be noted that the user can insert |
|  |  | elements with the same exponent value more than |
|  |  | once. **The program must operate on the list so** |
|  |  | **that the number of nodes in the list is minimal.** |
| 2 | display < LIST\_ID > | Display the polynomial <LIST\_ID> on a single |
|  |  | line as a series of integers that are the coefficients |
|  |  | and exponents for each element, each value |
|  |  | separated by a space. **The output ends with a** |
|  |  | **space** followed by a newline. Eg: |

|  |  |  |
| --- | --- | --- |
|  |  | 3<space>2<space>4<space>3<space>- |
|  |  | 5<space>0<space> |
| 3 | sort < LIST\_ID > | Arrange (sort) the nodes in the linked list so that it |
|  |  | represents a polynomial with elements with |
|  |  | **decreasing exponents**. Display the polynomial |
|  |  | after sorting as described in the display command |
|  |  | to the screen. Eg: |
|  |  | 4<space>3<space>3<space>2<space>- |
|  |  | 5<space>0<space> |
| 4 | add | Sum the two polynomials 1 and 2 and store the |
|  |  | resulting polynomial with a LIST\_ID of 3 in a |
|  |  | linked list. Display the resulting polynomial **after** |
|  |  | **sorting** to the screen as described in the display |
|  |  | command |
| 5 | multiply | Calculate the product of the two polynomials 1,2 |
|  |  | and store the resulting polynomial with a |
|  |  | LIST\_ID of 4 into a linked list. Display the |
|  |  | resulting polynomial **after sorting** on the screen. |
| 6 | space < LIST\_ID > | Displays the number of bytes that the list |
|  |  | corresponding to the <LIST\_ID> polynomial |
|  |  | occupies in memory, given that each node in the |
|  |  | list occupies 10 bytes (regardless of how the |
|  |  | program actually represents the list). |
| 7 | reset <LIST\_ID> | Free the list corresponding to the polynomial |
|  |  | <LIST\_ID>. The polynomial <LIST\_ID> is |
|  |  | initialized to the state at the start of the program. |
| 8 | \*\*\* | Exit the command waiting loop, free the memory |
|  |  | being allocated for the polynomials, and exit the |
|  |  | program. |

**Input**

* Each line contains a command with above format, terminated by a line \*\*\*

**Output**

* Each line contains a result printed out (by commands 1, 2, 3, 4, 5, 6, 7, 8 described in the table above)

**Example for student**

|  |  |  |
| --- | --- | --- |
|  | **Stdin** | **illustration** |
| insert 1 | -6 3 | L1: -6x^3 |
| insert 1 | 4 2 | L1: 4x^2 - 6x^3 |

|  |  |
| --- | --- |
| insert 1 1 0 | L1: 1 + 4x^2 - 6x^3 |
| insert 1 2 3 | L1: 1 + 4x^2 -4x^3 |
| display 1 |  |
| sort 1 |  |
| insert 2 -1 1 | L2: -x |
| insert 2 -2 2 | L2: -2x^2 - x |
| display 2 |  |
| add |  |
| space 3 |  |
| multiply |  |
| \*\*\* |  |
|  |  |
| **Stdout** | **illustration** |
| 1 0 4 2 -4 3<space> | 1 + 4x^2 -4x^3 |
| -4 3 4 2 1 0<space> | -4x^3 + 4x^2 +1 |
| -2 2 -1 1<space> | -2x^2 - x |
| -4 3 2 2 -1 1 1 0<space> | L3: -4x^3+2x^2- x+1 |
| 40 | 40 byte |
| 8 5 -4 4 -4 3 -2 2 -1 1<space> | L4:8x^5-4x^4-4x^3-2x^2-x |