Let say we have two polynomials

$3x^2 + 2x + 1$ ——— (1)

$5x^2 - x + 2$ ——— (2)

Addition of two polynomials involves combining like terms present in the two polynomials.

means adding terms having same variables and same exponents.

Adding (1) and (2)

$$\begin{array}{r} 3x^2 + 2x + 1 \\ 5x^2 - x + 2 \\ \hline 8x^2 + x + 3 \end{array}$$

In these two polynomials, $3x^2$ and $5x^2$ are like terms. Similarly, 2x and x are like terms.

Addition of two polynomials becomes easier if the terms are arranged in descending order of their exponents.

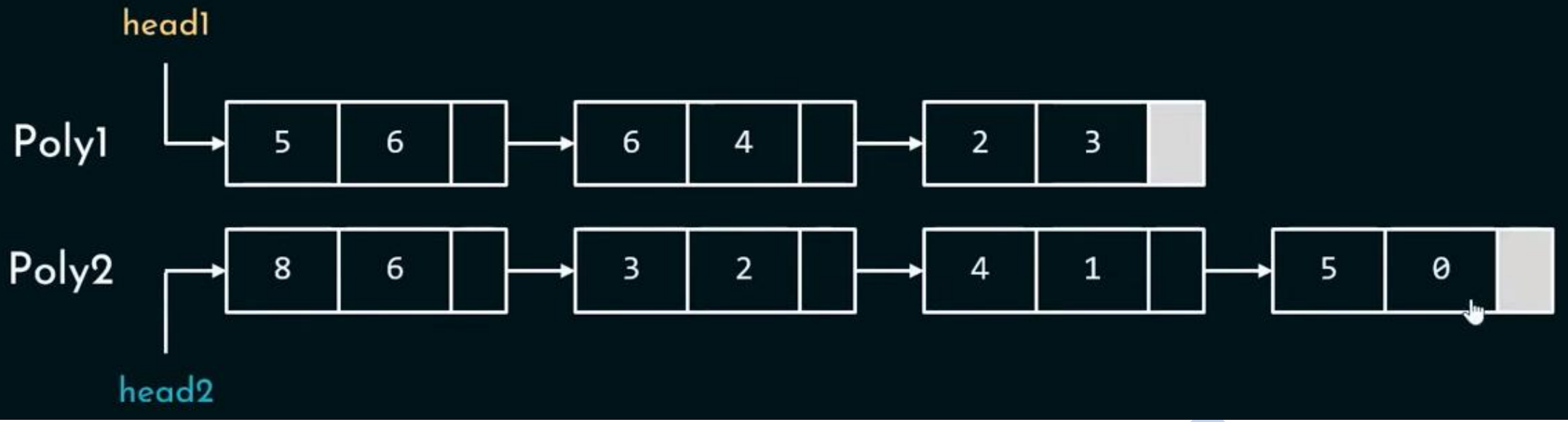
Let say we have the following two polynomials and our job is to add them.

$$5x^6 + 6x^4 + 2x^3$$

$$8x^6 + 3x^2 + 4x + 5$$

Terms are arranged in descending order of their exponents.

Let's represent the above two polynomials using linked lists.



Adding two polynomials means adding their like terms.

The only thing we have to do is to compare their exponents.

$$5x^6 + 6x^4 + 2x^3 \quad \text{(Poly 1)}$$

$$8x^6 + 3x^2 + 4x + 5 \quad \text{(Poly 2)}$$

$$13x^6$$

Compare the exponents of $5x^6$ and $8x^6$

$$6 = 6$$

$$5x^6 + 6x^4 + 2x^3 \quad \text{(Poly 1)}$$
$$8x^6 + 3x^2 + 4x + 5 \quad \text{(Poly 2)}$$

$$13x^6 + 6x^4$$

Compare the exponents of $6x^4$ and $3x^2$

$4 > 2$

$$5x^6 + 6x^4 + 2x^3 \quad \text{(Poly 1)}$$
$$8x^6 + 3x^2 + 4x + 5 \quad \text{(Poly 2)}$$

$$13x^6 + 6x^4 + 2x^3$$

Compare the exponents of $2x^3$ and $3x^2$

$3 > 2$

$$5x^6 + 6x^4 + 2x^3 \quad \text{(Poly 1)}$$
$$8x^6 + 3x^2 + 4x + 5 \quad \text{(Poly 2)}$$

$$13x^6 + 6x^4 + 2x^3$$

Terms of Poly 1 are finished.

Put all the remaining terms of Poly 2 in the resultant polynomial

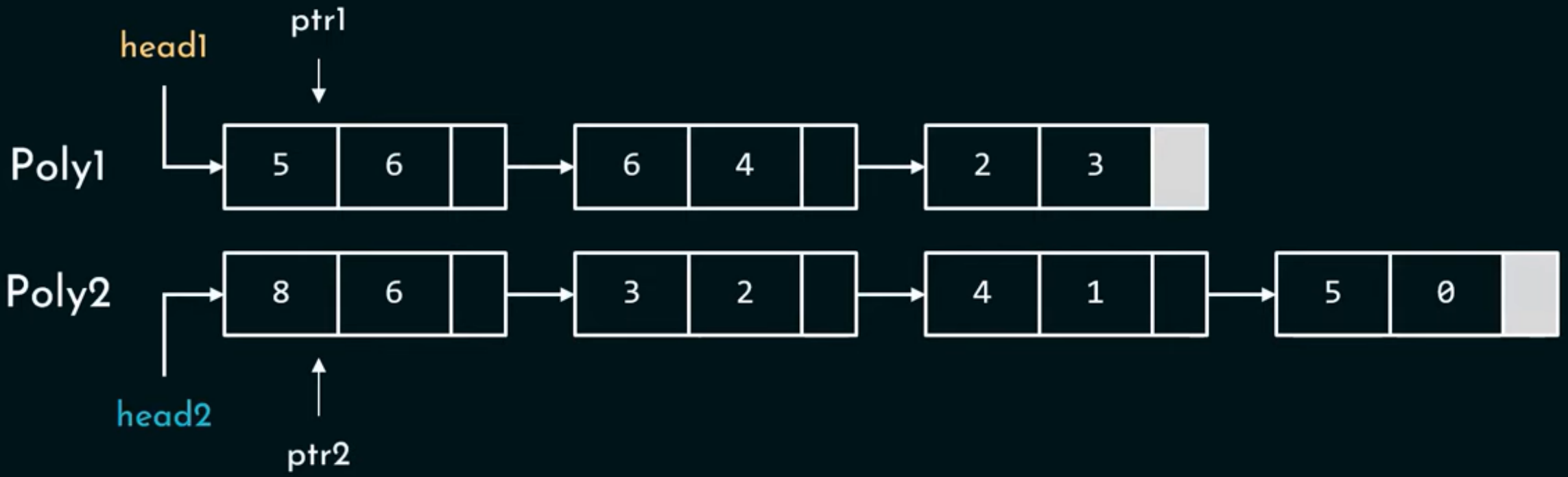
Adding two polynomials means adding their like terms.

The only thing we have to do is to compare their exponents.

$$5x^6 + 6x^4 + 2x^3 \quad \text{(Poly 1)}$$

$$8x^6 + 3x^2 + 4x + 5 \quad \text{(Poly 2)}$$

$$13x^6 + 6x^4 + 2x^3 + 3x^2 + 4x + 5$$



Algorithm:

Repeat the following until ptr1 or ptr2 becomes NULL

if(ptr1->expo == ptr2->expo)

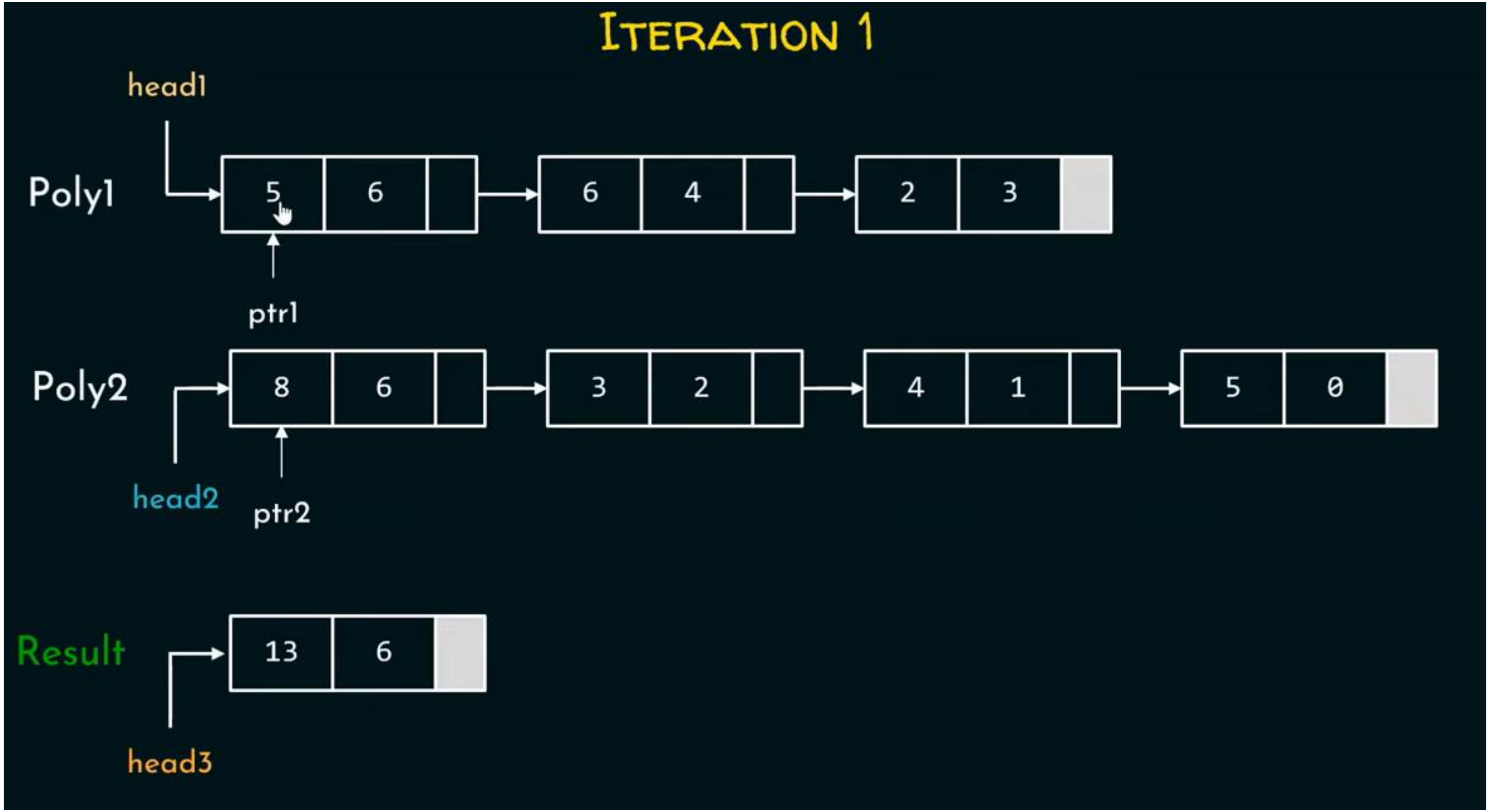
Add the coefficients and insert the newly created node in the resultant linked list and make ptr1 and ptr2 point to the next nodes.

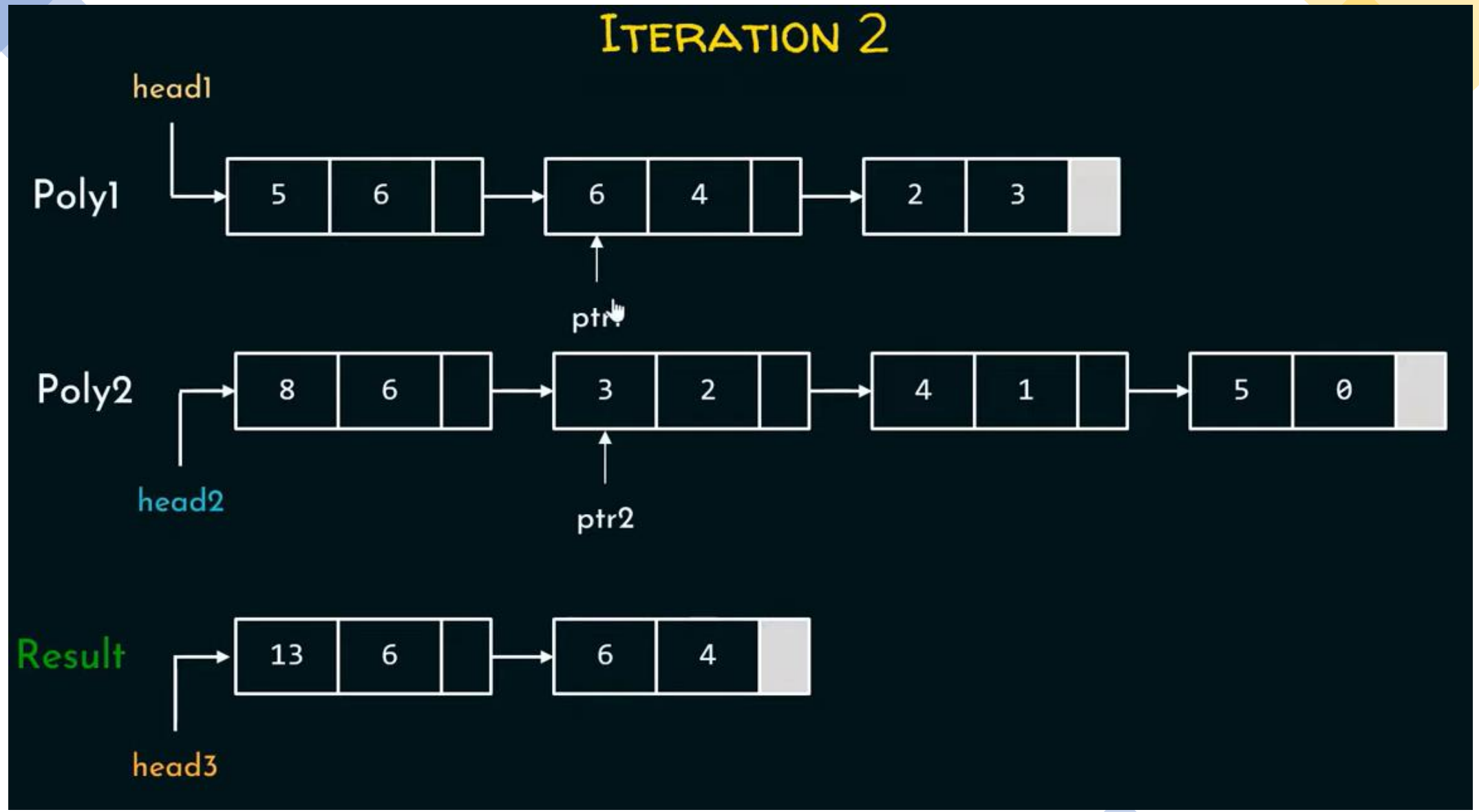
if(ptr1->expo > ptr2->expo)

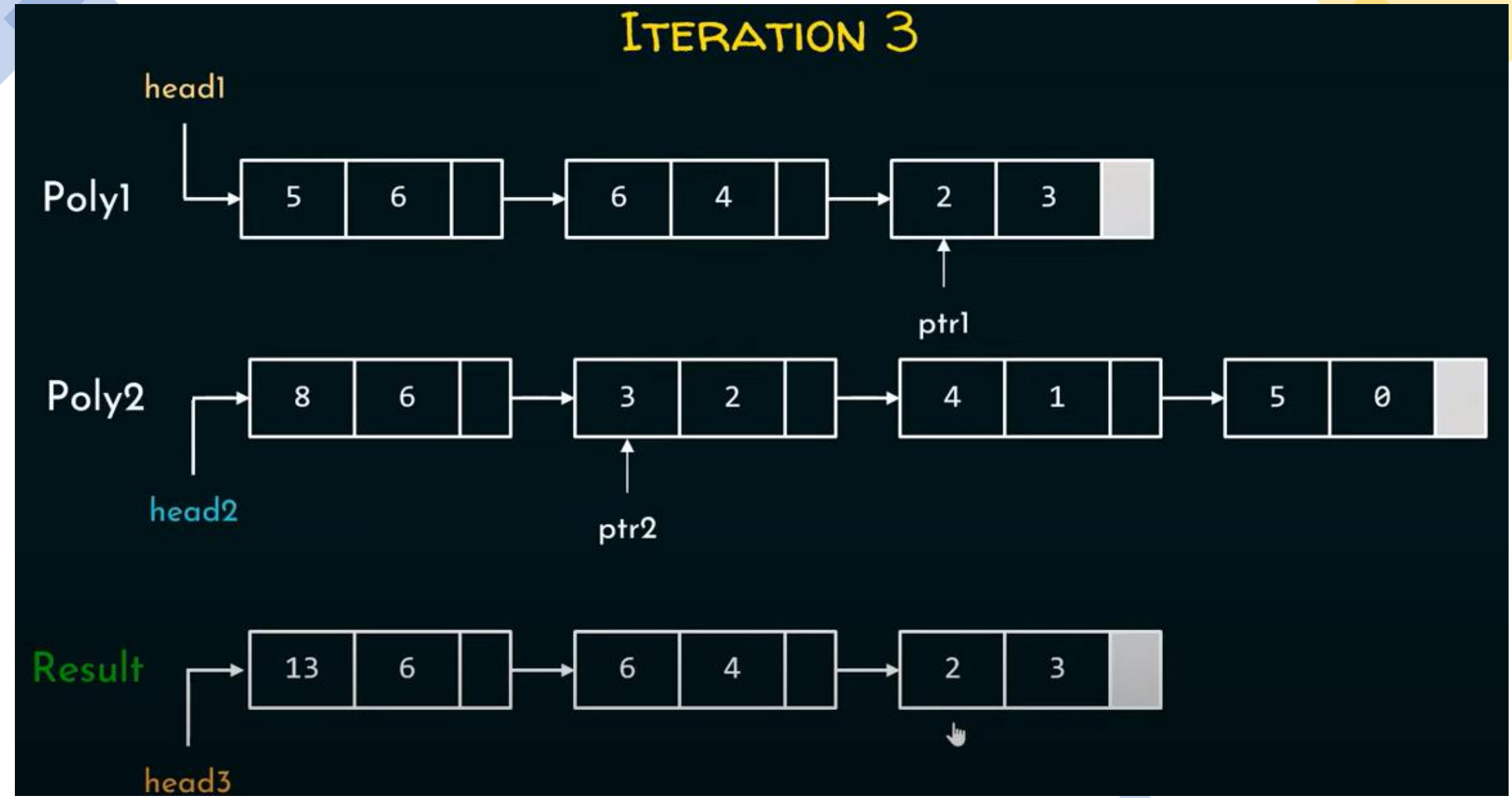
Insert the node pointed by ptr1 in the resultant linked list and make ptr1 point to the next node

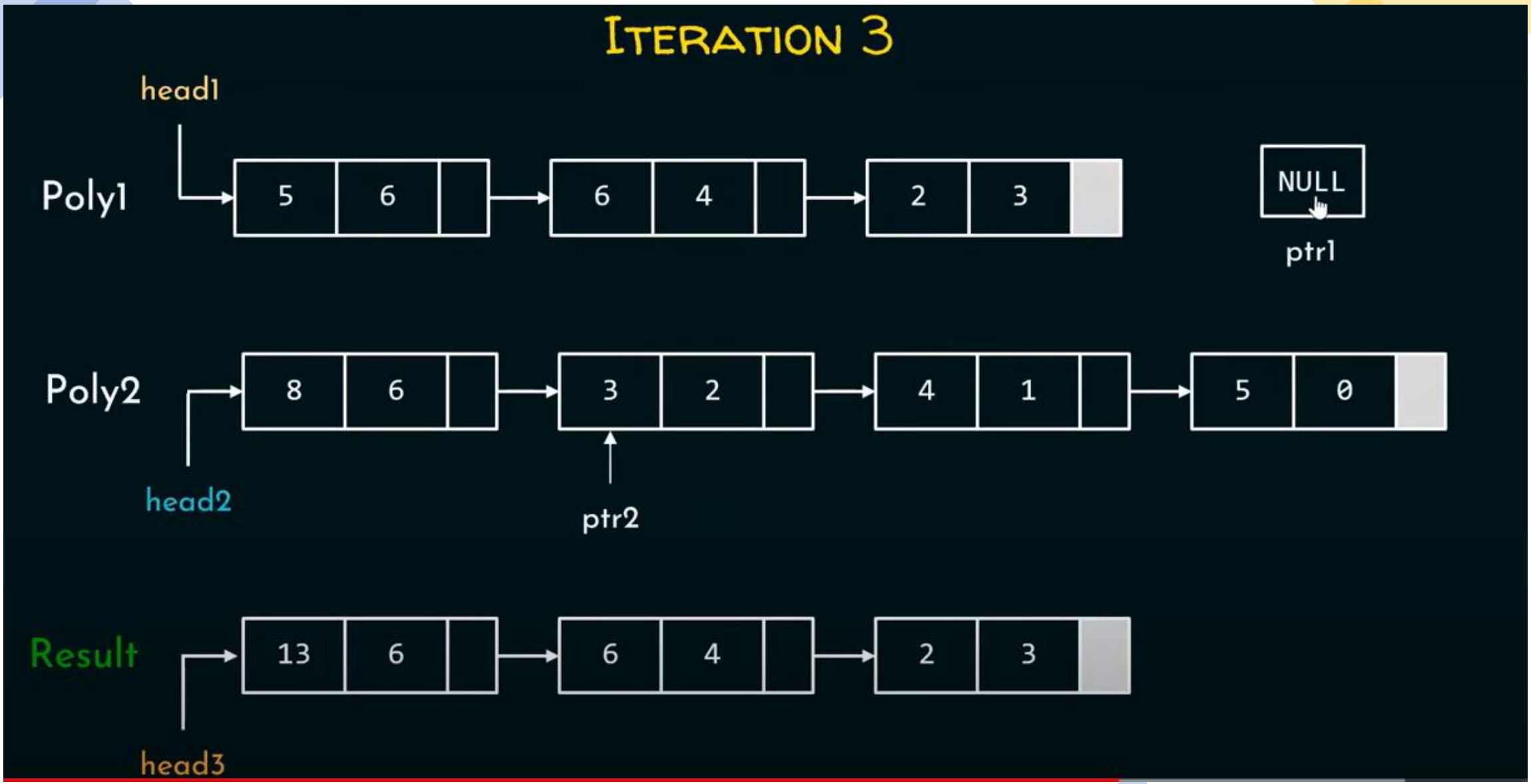
if(ptr1->expo < ptr2->expo)

Insert the node pointed by ptr2 in the resultant linked list and make ptr2 point to the next nod


ITERATION 1
head1
Poly1
5
6
6
4
2
3
ptr1
Poly2
8
6
3
2
4
1
5
0
head2
ptr2
Result
13
6
head3


ITERATION 2
head1
Poly1
5
6
6
4
2
3
ptr1
Poly2
8
6
3
2
4
1
5
0
head2
ptr2
Result
13
6
6
4
head3

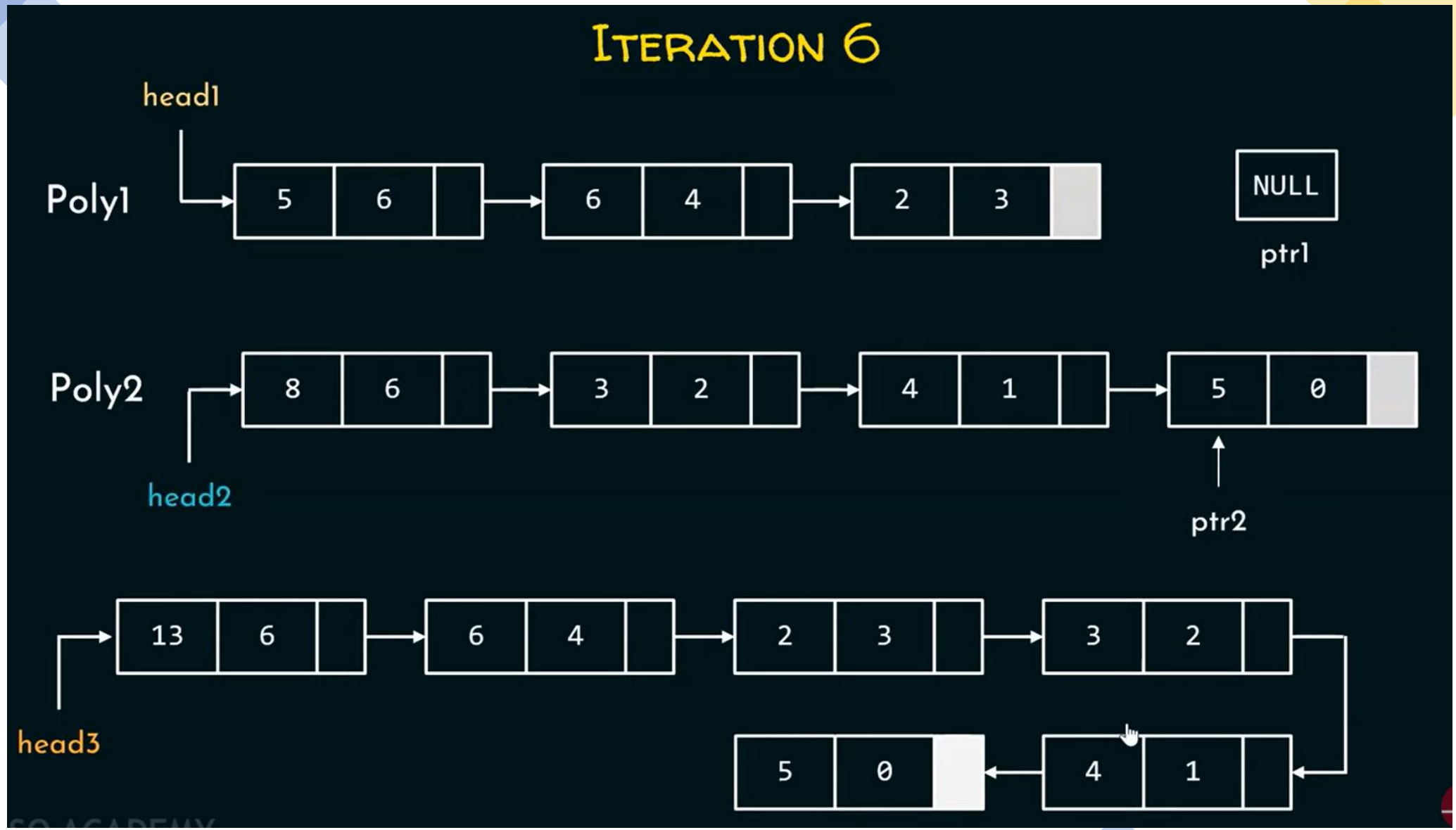

ITERATION 3
head1
Poly1
5
6
6
4
2
3
ptr1
Poly2
8
6
3
2
4
1
5
0
head2
ptr2
Result
13
6
6
4
2
3
head3


ITERATION 3
head1
Poly1
5
6
6
4
2
3
NULL
ptr1
Poly2
8
6
3
2
4
1
5
0
head2
ptr2
Result
13
6
6
4
2
3
head3

After iteration 3, ptr1 becomes NULL.
Now, there is nothing to compare with ptr2->expo.

At this stage, we must add all the remaining nodes of the second linked list at the end of the resultant linked list.

```
repeat until ptr2 != NULL
        Insert the remaining nodes
repeat until ptr1 != NULL
        Insert the remaining nodes
```


ITERATION 6
head1
Poly1
5
6
6
4
2
3
NULL
ptr1
Poly2
8
6
3
2
4
1
5
0
head2
ptr2
13
6
6
4
2
3
3
2
head3
5
0
4
1

```
nodeptr polyAdd(nodeptr head1,nodeptr head2)
{
    nodeptr ptr1=head1;
    nodeptr ptr2=head2;
    nodeptr head3 = NULL;
    while(ptr1!=NULL && ptr2!=NULL)
    {
        if(ptr1->expo == ptr2->expo)
        {
            head3 = insertpoly(head3,ptr1->coef+ptr2->coef,ptr1->expo);
            ptr1= ptr1->next;
            ptr2= ptr2->next;
        }
        else if(ptr1->expo > ptr2->expo)
        {
            head3 = insertpoly(head3,ptr1->coef,ptr1->expo);
            ptr1=ptr1->next;
        }
        else if(ptr1->expo < ptr2->expo)
        {
            head3 = insertpoly(head3,ptr2->coef,ptr2->expo);
            ptr2=ptr2->next;
        }
    }
     while(ptr1!=NULL)
     {
         head3 = insertpoly(head3,ptr1->coef,ptr1->expo);
         ptr1=ptr1->next;
     }
     while(ptr2!=NULL)
     {
         head3 = insertpoly(head3,ptr2->coef,ptr2->expo);
         ptr2=ptr2->next;
     }
     return head3;
}
```