Program 1

	0	1	2	3	4
① data | 7 | 0 | 1 | -3 | 12 |

size [5]

i   0̸ 1̸ 2̸ 3̸ 4̸ 5

result 0̸ 1

	0	1	2	3	4
② data | 0 | 2 | 0 | 11 | -1 |

size [5]

i   0̸ 1̸ 2̸ 3̸ 4̸ 5

result 0̸ 1̸ 2

double count_Positive_2 (const vector < double> &data)

Program 2

① data

	0	1	2	3	4
	2	-5	0	-3	23

size | 5 |

i   Ø 1̸ 2̸ 3̸ 4 5

data[i] - 1̸ 5̸ 3

② data

	0	1	2	3	4
	0	-2	-4	11	-1

size | 5 |

i   Ø 1̸ 2̸ 3̸ 4

data[i]   Ø 7̸ 4̸ 1

void update_data (vector < double > & data).

Program 3

① data

	0	1	2	3	4
	2	-5	0	1	23

data [idx] | 1 |
             |---|
             | -5 |

data [last] | 4 |
              |---|
              | 23 |

data[idx] = data[last]

data[idx] = | 1 |
              |---|
              | 23 |

data

2	23	0	1	

② data

	0	1
	0	-2

data [idx] | 1 |
             |---|
             | -2 |

data [last] | 1 |
              |---|
              | -2 |

data [idx] = data [last]

data [idx] = | 1 |
               |---|
               | -2 |

data

data | 0 |
       |---|
       | 0 |

③ data $\quad\overset{0\quad1}{\boxed{4\ |\ 3}}$

data [idx] $\quad\overset{4}{\boxed{0}}$

data [last] $\quad\overset{1}{\boxed{3}}$

data $\quad\overset{0\quad1}{\boxed{4\ |\ 3}}$

void Replace_Remove (vector < double > & data, int data)

Program 4

data

0	1	2	3	4
4	-6	0	8	-7

data [idx] =

2
0

$(idx >= data.size() - 1)$

For (int i = data.size() - 1; i > idx; i--)

    data[i] = data[i-1];

    data[4] = data[3]
    data[3] = -7

    data[3] = data[2]
    data[2] = 8

data

4	-6	8	-7	-7

void Insert_Shift (vector <double> &data, int idx, double value