

C++

Урок 7

A hand-drawn blue oval frame with a double-line border, centered on the page. The text 'Разбор ДЗ' is written inside this frame in a bold, black, sans-serif font.

Разбор ДЗ

Разбор дз

1 задача:

На вход подается число, вывести да, если оно палиндром. (Решать через одномерный массив и сравнивать левую и правую часть через индексы.)

Число занести в массив через цикл while, по одной цифре.

Разбор дз

Алгоритм:

- Выделить память под одномерный массив и введенное число и нужные нам переменные.
- Узнать количество разрядов в числе и занести каждый разряд к своей ячейке.
- В цикле делать проверку первого и последнего разряда числа и тд.

Разбор дз

Листинг:

```
int arr[100] = {};  
long long int x, n, cnt, tmp;  
cin >> x; //получаем пользовательское значение  
cnt = 0;   //количество разрядов вводимого числа  
tmp = x;  
           //временная переменная для расчетов количества разрядов  
int i = 0; //для работы с массивом внутри while
```

Разбор дз

ЛИСТИНГ:

```
//считаем количество цифр и записываем цифры в массив
while (tmp != 0)
{
    arr[i] = tmp % 10; //записываем в массив в обратном порядке
    tmp /= 10;
    i++;
    cnt++;
}
```

Разбор дз

ЛИСТИНГ:

```
bool flag = 0; //для проверки числа
for (int i = 0; i < cnt; i++)
{
    if (arr[i] != arr[cnt - 1 - i])
    {
        flag = 1;
    }
}

if (flag)
    cout << "Not a Palindrom";
else
    cout << "Palindrom";
```

Разбор дз

ЛИСТИНГ:

```
bool flag = 0; //для проверки числа
for (int i = 0; i < cnt; i++)
{
    if (arr[i] != arr[cnt - 1 - i])
    {
        flag = 1;
    }
}

if (flag)
    cout << "Not a Palindrom";
else
    cout << "Palindrom";
```

Можно ли чуть оптимизировать?

Разбор дз

ЛИСТИНГ:

```
bool flag = 0; //для проверки числа
for (int i = 0; i < cnt / 2; i++)
{
    if (arr[i] != arr[cnt - 1 - i])
    {
        flag = 1;
    }
}

if (flag)
    cout << "Not a Palindrom";
else
    cout << "Palindrom";
```

Разбор дз

7 задача:

Пользователь вводит размер массива n и вводит значения.

Затем пользователь вводит размер второго массива и вводит в него значения. Вставьте значения из массива 2 на позицию индекса k первого.

Разбор дз

Алгоритм:

- Выделить память под 2 одномерных массива, их размерности и число (индекс вставки).
- Ввести 2 массива, индекс вставки.
- Реализовать вставку.
- Вывести получившийся массив.

Разбор дз

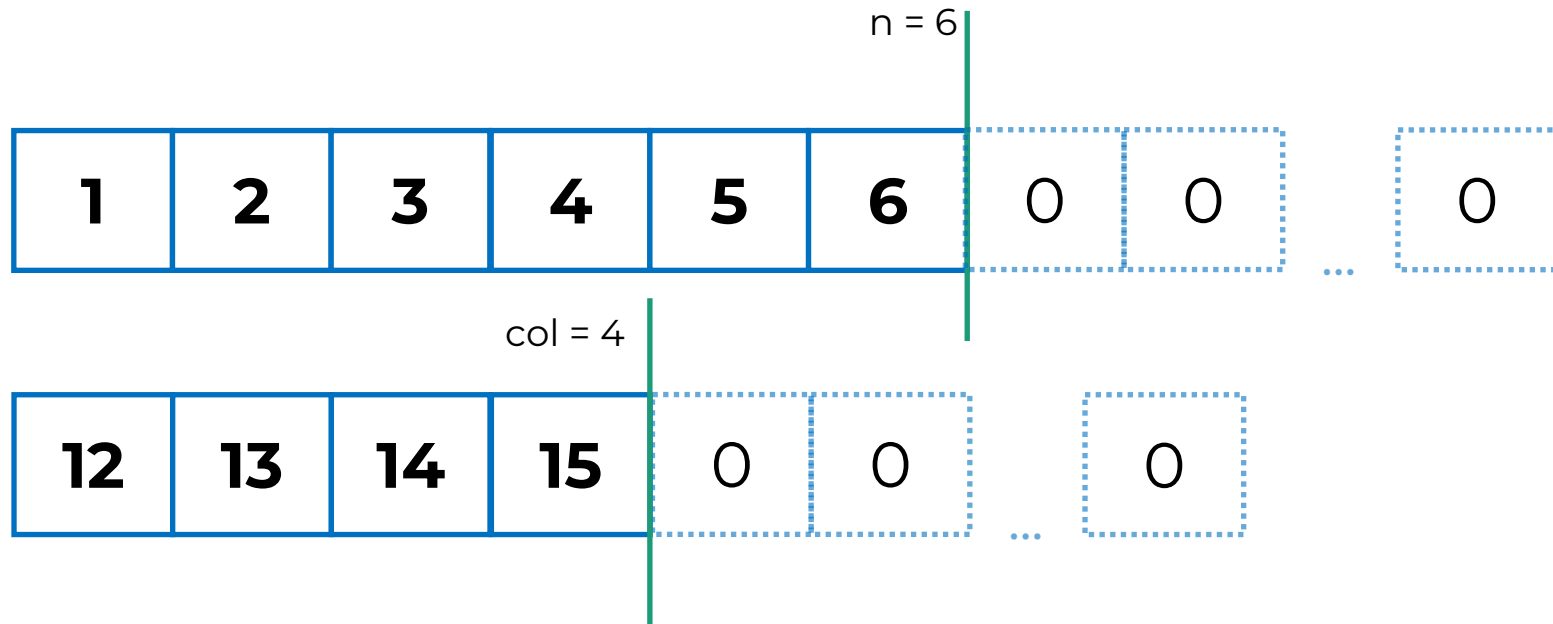
Листинг:

```
int arr[100] = {1, 2, 3, 4, 5, 6}; //изначальный массив
int arr2[100] = {12, 13, 14, 15}; //массив вставки
int n = 6; //размер первого
int pos = 3; //индекс вставки
int col = 4; //количество символов вставки (размер второго массива)
```

Объявление + инициализация

Разбор дз

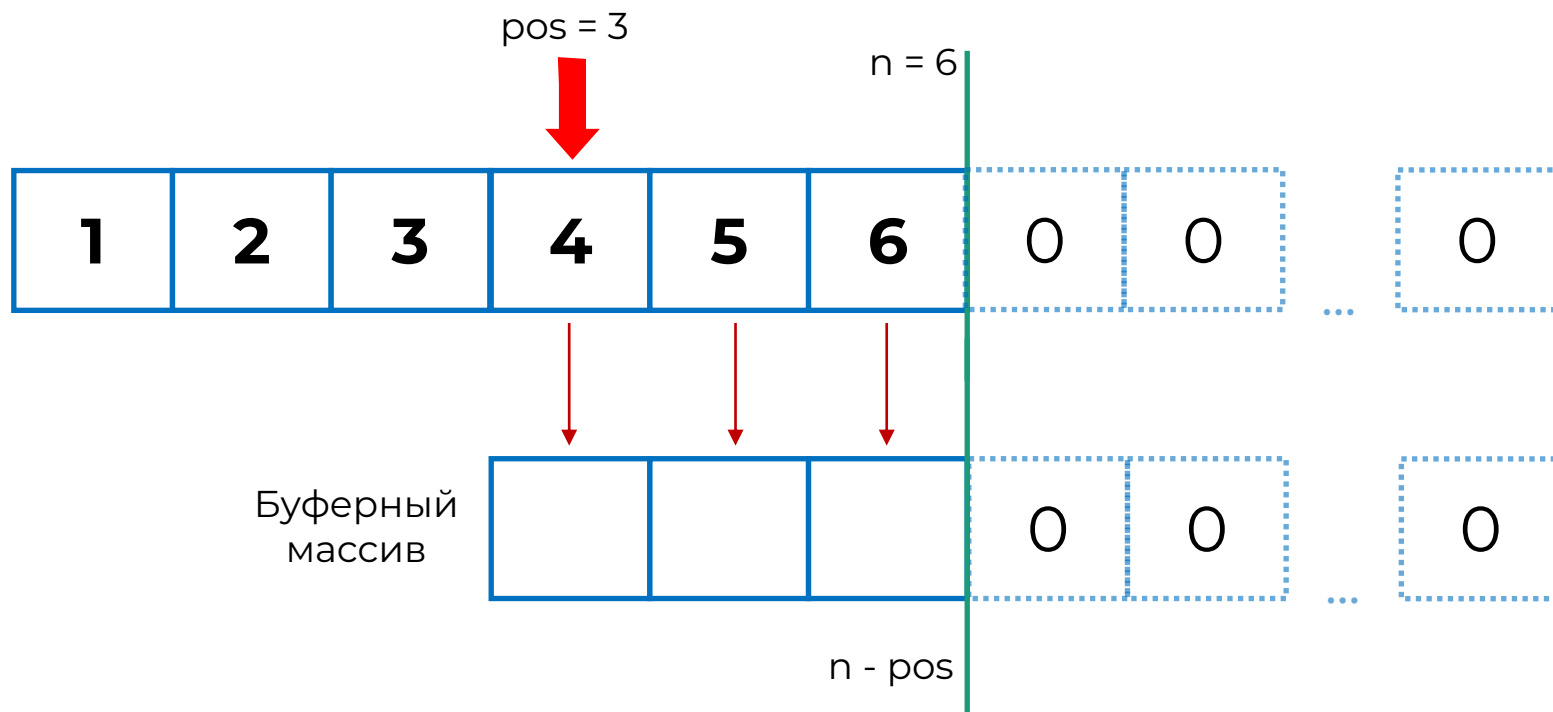
1 вариант:



Исходные данные

Разбор дз

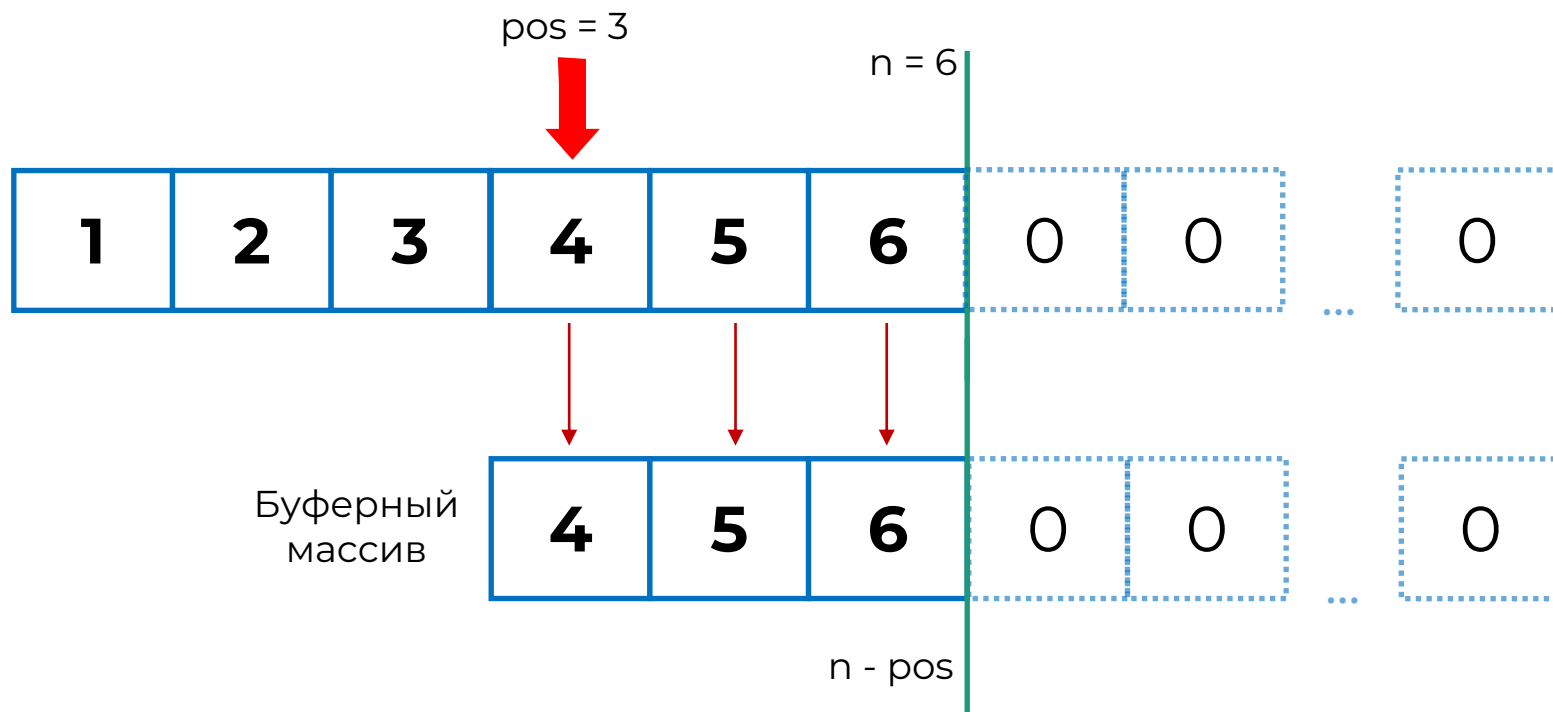
1 вариант:



Копирование значение с позиции вставки

Разбор дз

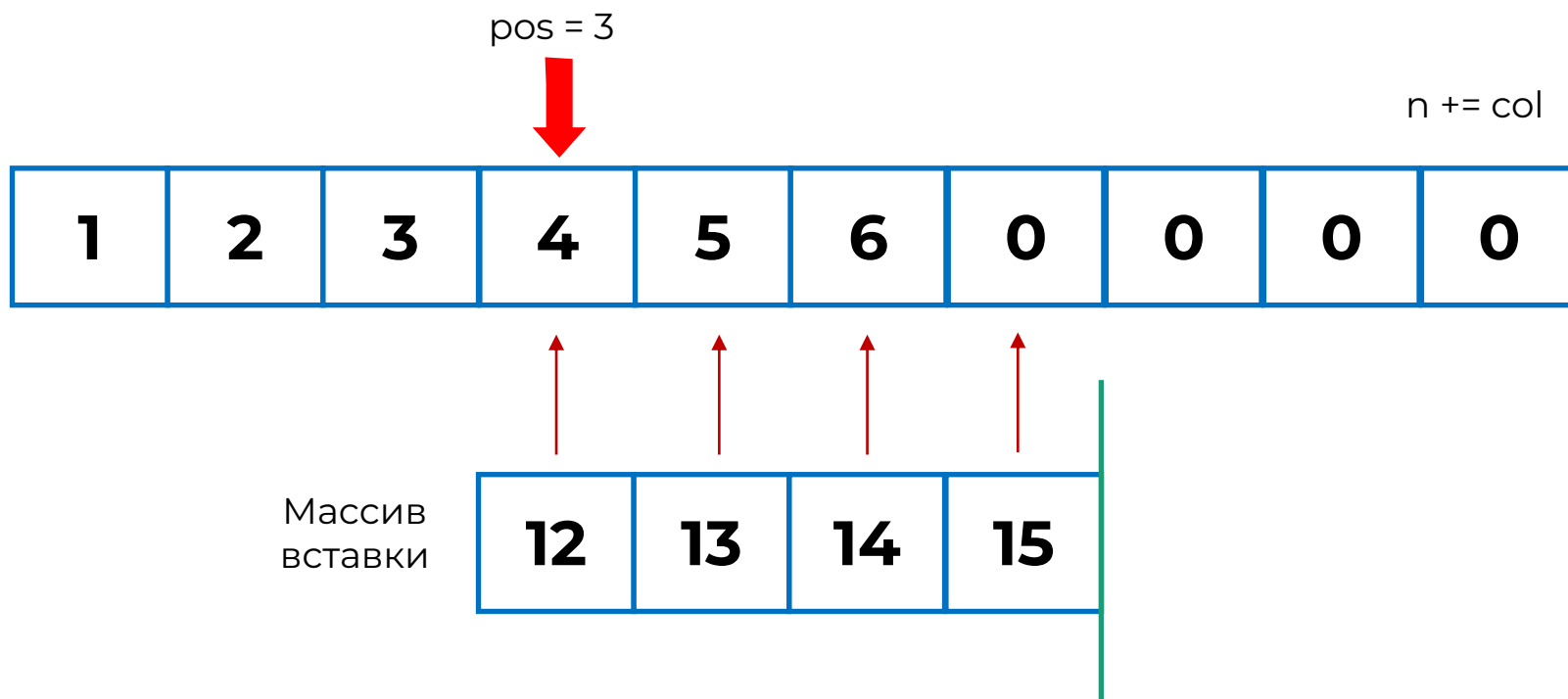
1 вариант:



Копирование значение с позиции вставки

Разбор дз

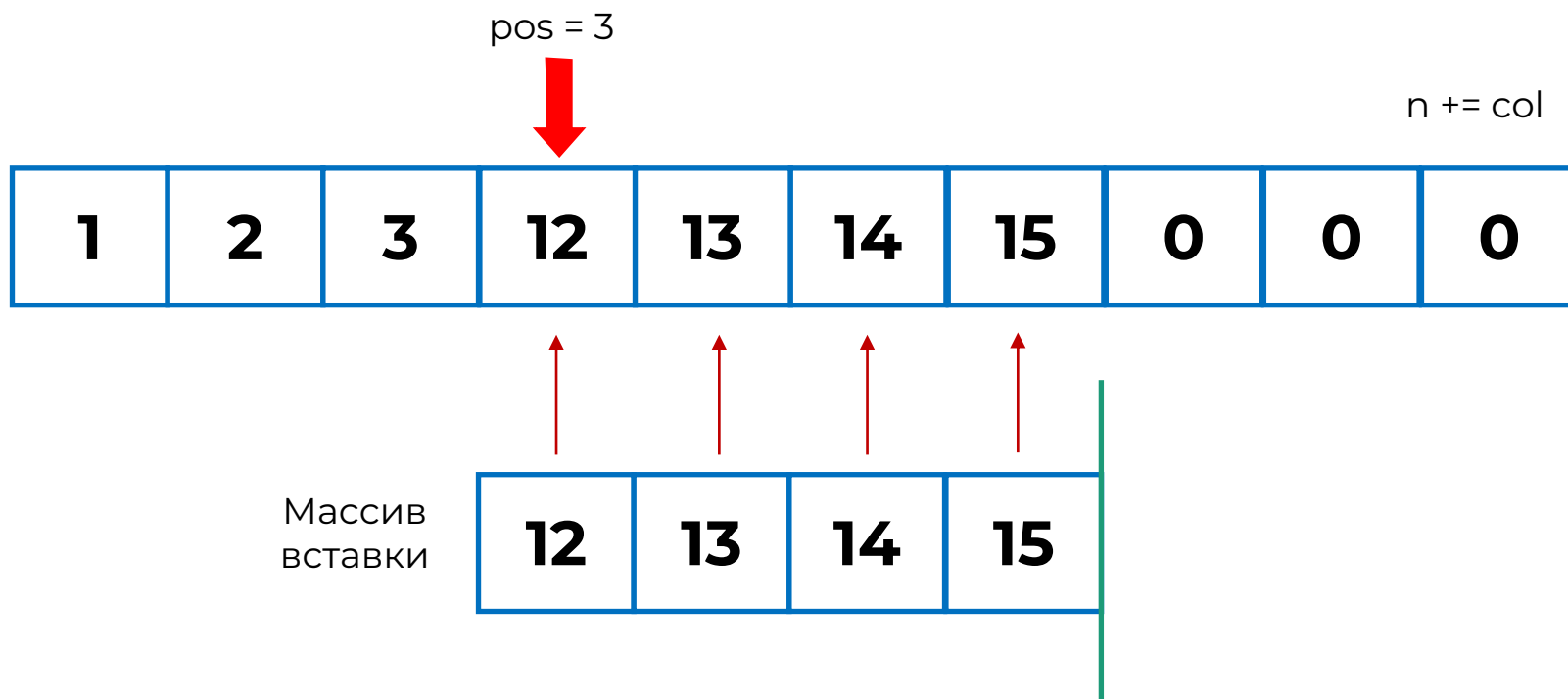
1 вариант:



Вставка значений в исходный массив

Разбор дз

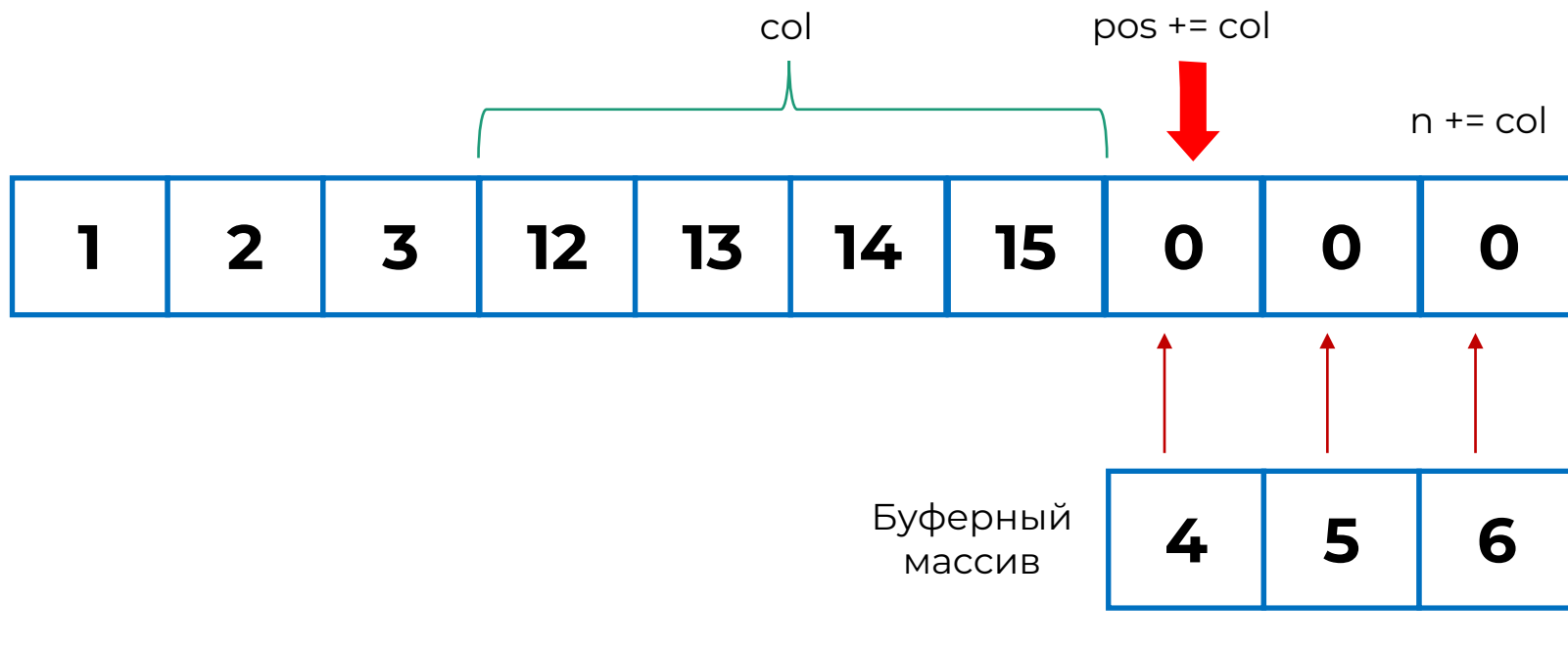
1 вариант:



Вставка значений в исходный массив

Разбор дз

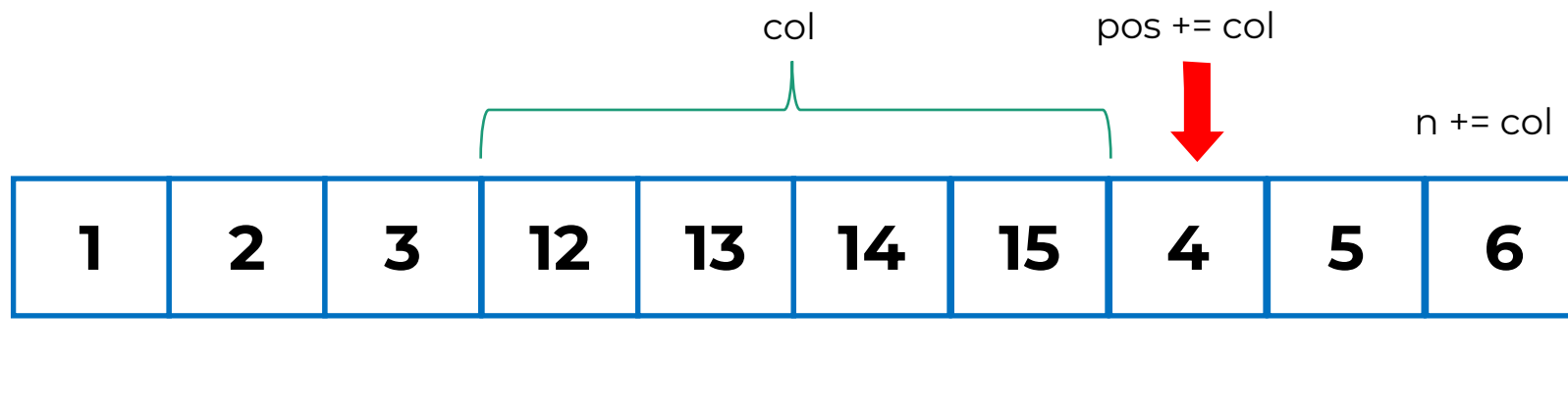
1 вариант:



Вставка значений в исходный массив
из буфера

Разбор дз

1 вариант:



Вывод получившегося массива

Разбор дз

Листинг 1 вариант:

```
int arr3[100] = {};  
for (int i = pos; i < n; i++)  
{  
    arr3[i - pos] = arr[i];  
    //запись в новый массив всех значений после данного индекса  
}  
  
n += col;  
    //увеличиваем размер 1 массива на кол-во эл - тов вставки  
  
for (int i = pos; i < n; i++)  
{  
    arr[i] = arr2[i - pos]; //вставка значений из массива 2  
}  
pos += col; //увеличиваем позицию для вставки из массива 3  
  
for (int i = pos; i < n; i++)  
{  
    arr[i] = arr3[i - pos]; //запись в исходный массив  
}
```

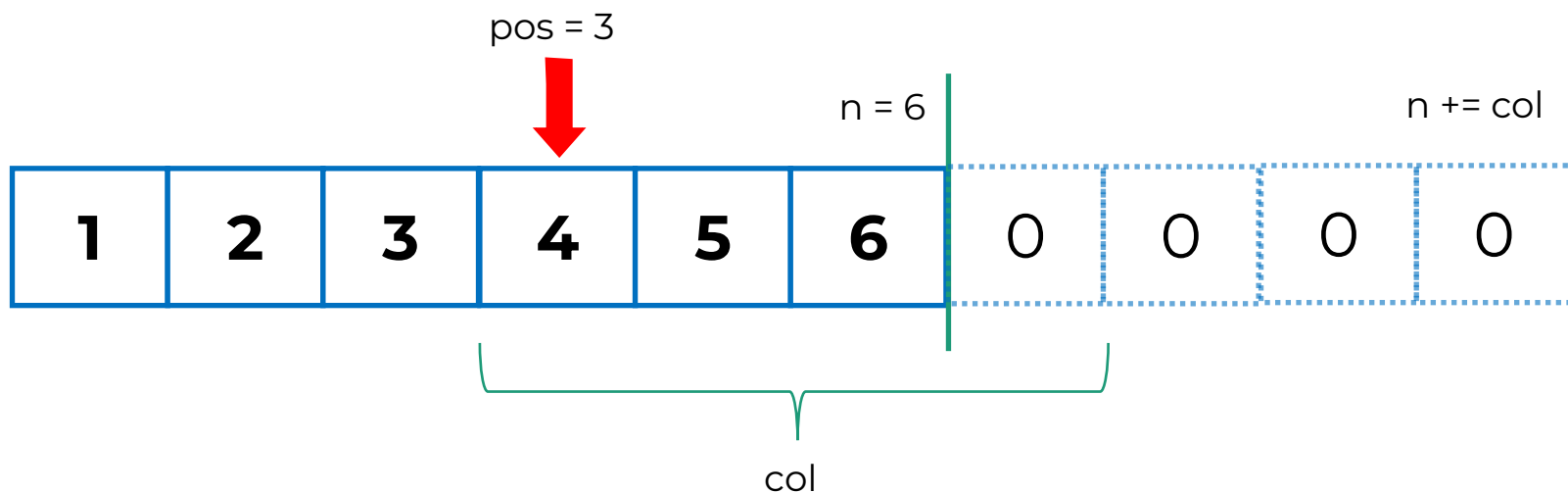
Через дополнительный массив

**Можно ли обойтись без
буферного массива?**

Если да, то как?

Разбор дз

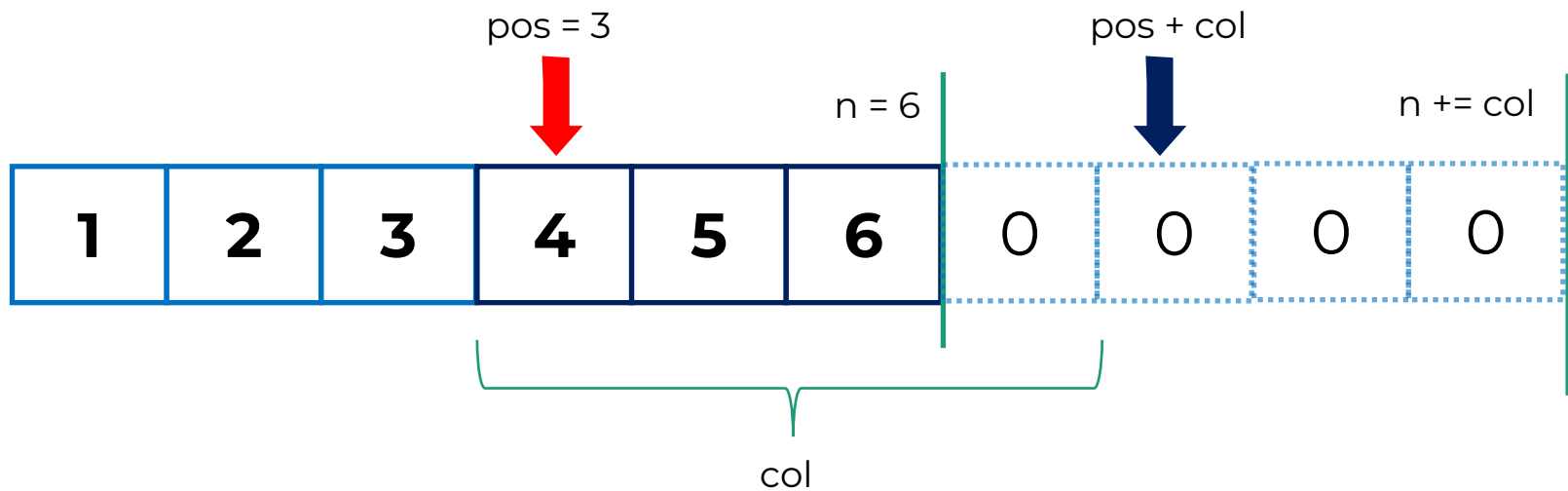
2 вариант:



Копия значений с pos на pos + col

Разбор дз

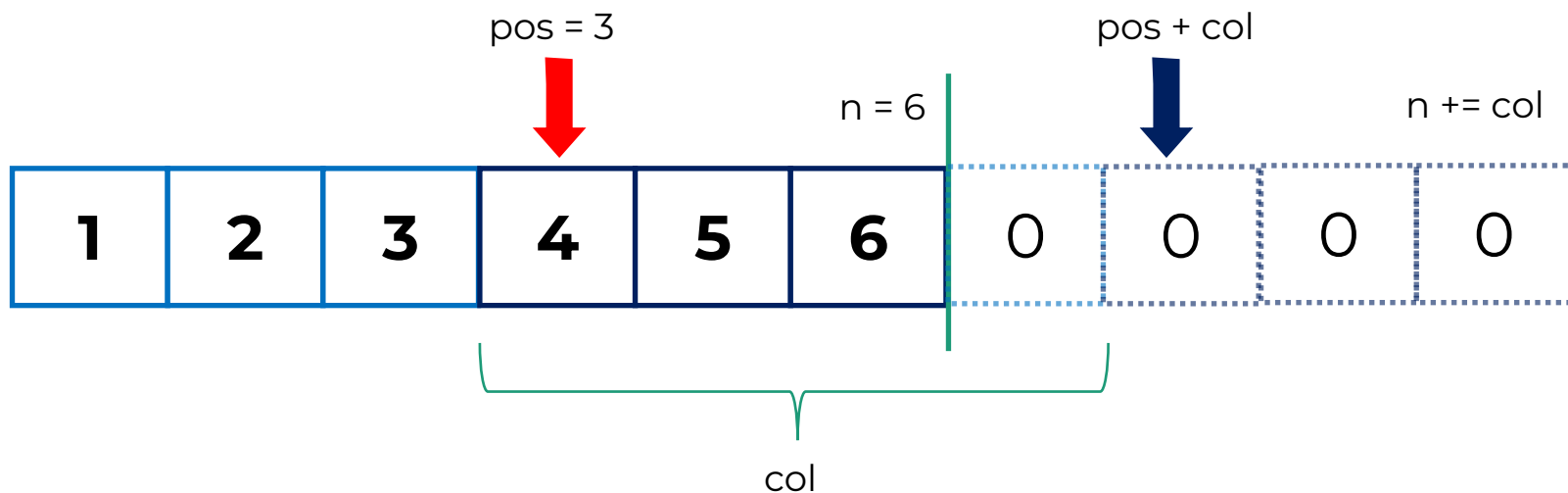
2 вариант:



Копия значений с pos на $pos + col$

Разбор дз

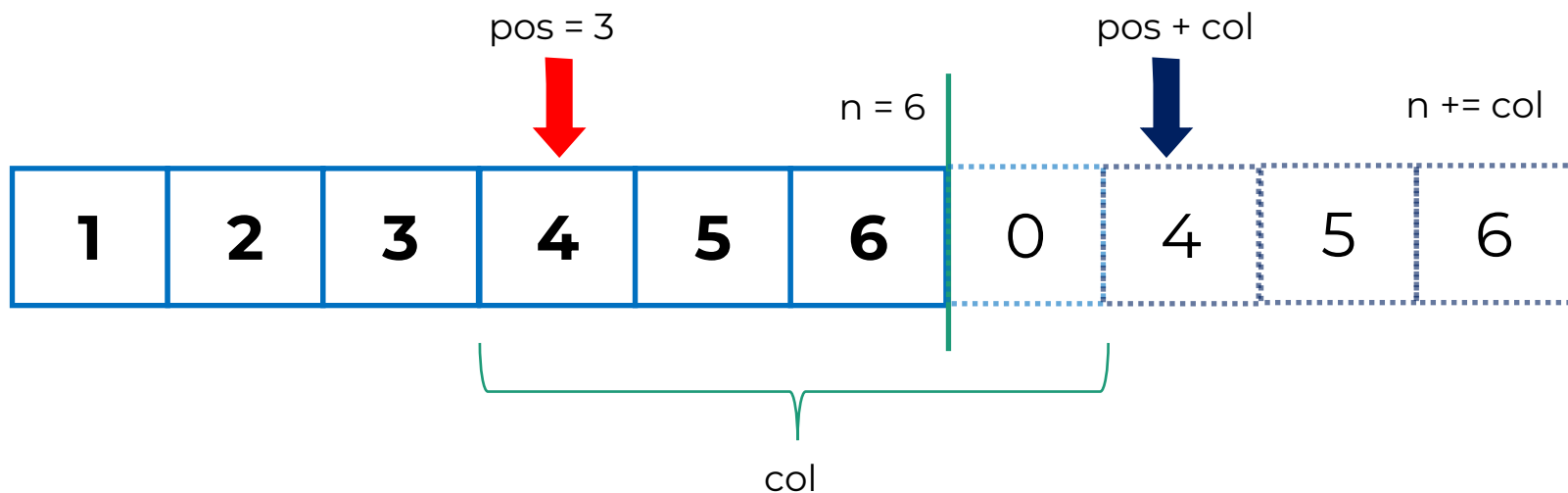
2 вариант:



Копия значений с pos на pos + col

Разбор дз

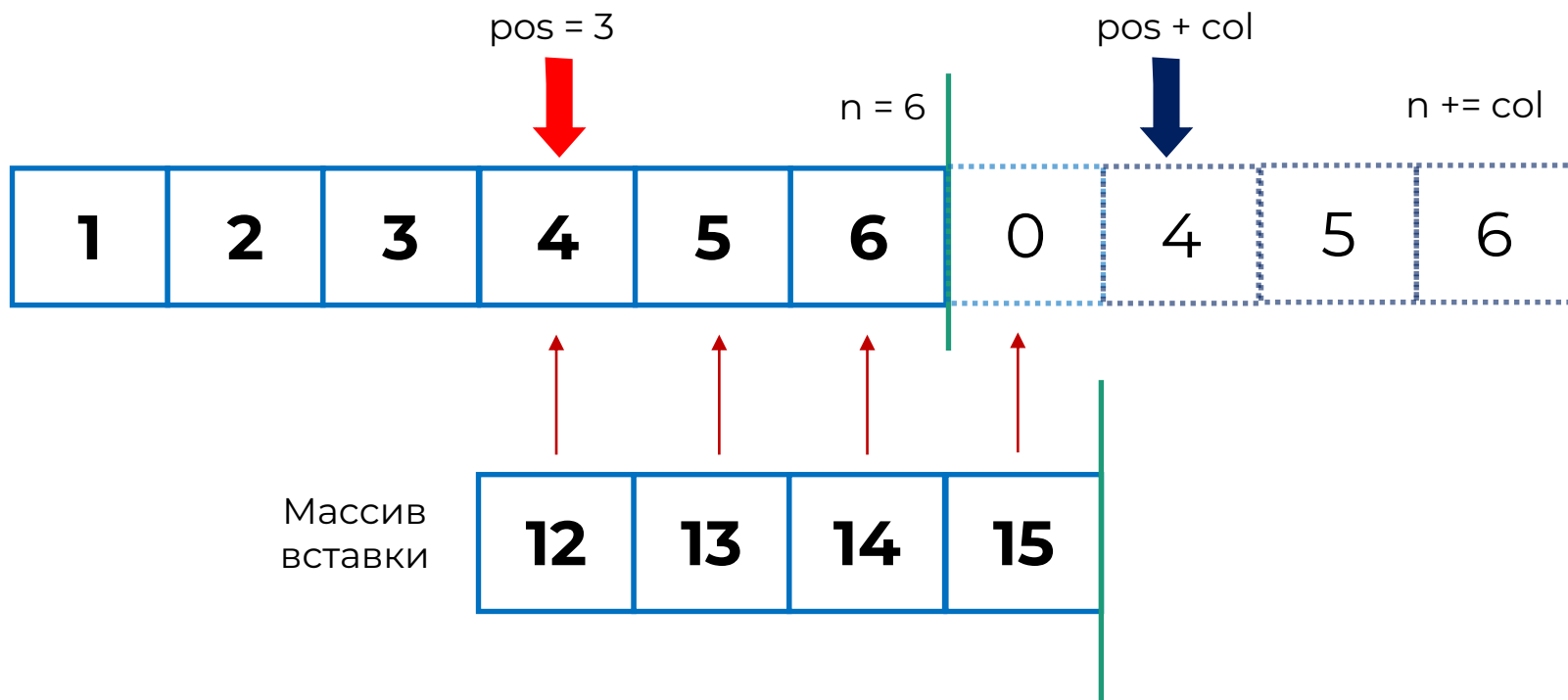
2 вариант:



Копия значений с pos на pos + col

Разбор дз

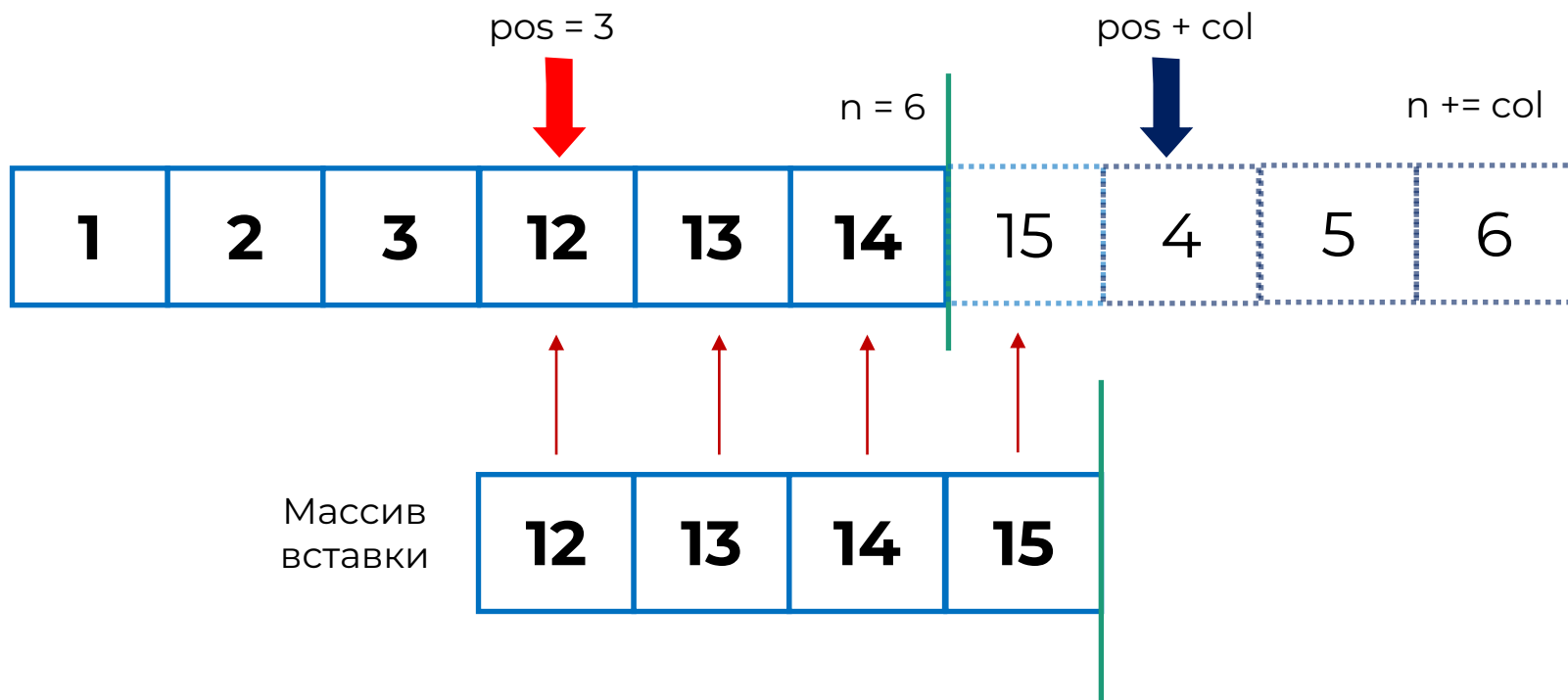
2 вариант:



Вставка значений в исходный массив

Разбор дз

2 вариант:



Вставка значений в исходный массив

Разбор дз

2 вариант:

n += col									
1	2	3	12	13	14	15	4	5	6

Вывод получившегося массива

Разбор дз

Листинг 2 вариант:

```
n+= col;
for (int i = n; i >= pos + col; i--){
    arr[i] = arr[i - col]; //делаем копию
}

for (int i = 0; i < col; i++)
{
    arr[i + pos] = arr2[i]; //реализуем вставку
}
```

Без дополнительного массива

A hand-drawn blue oval frame with a slightly irregular, sketchy border, centered on the page. The word "Повторение" is written inside this frame in a bold, black, sans-serif font.

Повторение

Что такое массив?

На что похож?

Какие виды массивов узнаете?

**Как лучше инициализировать
одномерный массив?**



Что такое вложенный цикл?

Задача

Пользователь вводит в консоль число - n (размер одномерного массива) и целое положительное число k .

Задача: Вставьте в массив число 12 по данному индексу.

Задача

Пользователь вводит в консоль число - n (размер одномерного массива) и целое положительное число k .

Задача: Удалите из массива число по данному индексу.

Задача

Пользователь вводит размер массива n и вводит значения. Затем пользователь вводит размер второго массива и вводит в него значения.

Задача: Удалите из массива 1 те элементы, значение которых совпадает со значениями из массива 2.

A hand-drawn blue oval frame with a slightly irregular, sketchy border, centered on the page. The word 'Сортировка' is written inside this frame in a bold, black, sans-serif font.

Сортировка



Что такое сортировка?

Сортировка

Сортировка – это алгоритм для упорядочивания данных по некоторым признакам.



Где применяется сортировка?

Сортировка

Для того, чтобы отсортировать массив по **возрастанию**:

```
for (int i = 0; i < n - 1; i++)  
{  
    for (int j = 0; j < n - 1; j++)  
    {  
        if (arr[j] > arr[j + 1])  
            swap(arr[j], arr[j + 1]);  
    }  
}
```

Проходим в циклах до **n - 1**

Сортировка

Для того, чтобы отсортировать массив **не по возрастанию**:

```
for (int i = 0; i < n - 1; i++)  
{  
    for (int j = 0; j < n - 1; j++)  
    {  
        if (arr[j] < arr[j + 1])  
            swap(arr[j], arr[j + 1]);  
    }  
}
```

Проходим в циклах до **n - 1**

Почему до $n - 1$?



Можно ли оптимизировать?

Сортировка

Для того, чтобы отсортировать массив по **возрастанию**:

```
for (int i = 0; i < n - 1; i++)  
{  
    for (int j = 0; j < n - i - 1; j++)  
    {  
        if (arr[j] > arr[j + 1])  
            swap(arr[j], arr[j + 1]);  
    }  
}
```

Проходим во внутреннем
цикле до **$n - i - 1$**



Задача

Умпа-Лумпы решили взять с собой в лилипутию песелей. При прохождении инопланетного таможенного контроля нужно выполнять взвешивание домашних животных. (массив весов)

Задача: Отсортируйте по возрастанию веса всех песелей Умпа-Лумпов и выведите их на экран таможенника.



Размер массива

Размер массива

Для того, чтобы объявить и инициализировать:

```
int arr[] = {1,5,7,8,12}; //сам определяет размер
```

Размер массива

Для того, чтобы объявить и инициализировать:

```
int arr[] = {1,5,7,8,12}; //сам определяет размер
```

```
for (int i = 0; i < ? ;i++){  
    cout <<arr[i]<<" ";  
}
```

Но что записать тут?

Размер массива

ЛИСТИНГ:

```
int arr[] = {1,5,7,8,12};  
    //сам определяет размер  
int n = sizeof(arr) / sizeof(arr[0]);  
  
for (int i = 0; i < n ;i++){  
    cout <<arr[i]<<" ";  
}
```

Задача

Инициализируйте в момент объявления массива значения роста песелей.

Задача: Узнайте количество введенных значений.