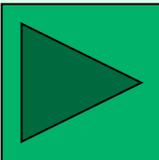


АМ

Алгоритмы обработки массивов



Одномерные массивы

АМ

Массив это - последовательность элементов одного типа.

Свойства массива:

- Массив имеет *имя*
- Массив имеет *размер*.

Например:

A : Array [1.. 10] of Integer;



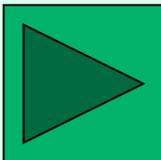
Имя массива



Размер



Тип элементов



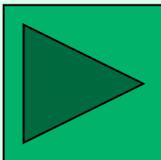
Ввод элементов массива

```
Read (N);  
For I :=1 to N do  
begin  
    Write ('A(', I, '=');  
    Read (A[I])  
end;
```

Номер (I) элемента	Значение элемента
1	$A(1) = 5$
2	$A(2) = 3$
3	$A(3) = 8$
4	$A(4) = 2$
5	$A(5) = 9$

$N=5$

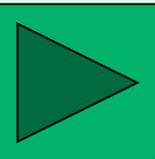
$A = \{5, 3, 8, 2, 9\}$



Алгоритмы обработки массивов

АМ

1. Суммирование элементов массива
2. Удаление K-го элемента из массива
3. Перестановка элементов в массиве
4. Поиск Max, Min



Полка с корзинками

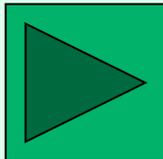


$N := 5;$

For $i := 1$ to N do

Read ($A[i]$);

I	1	2	3	4	5
A[I]=	5	3	8	2	9



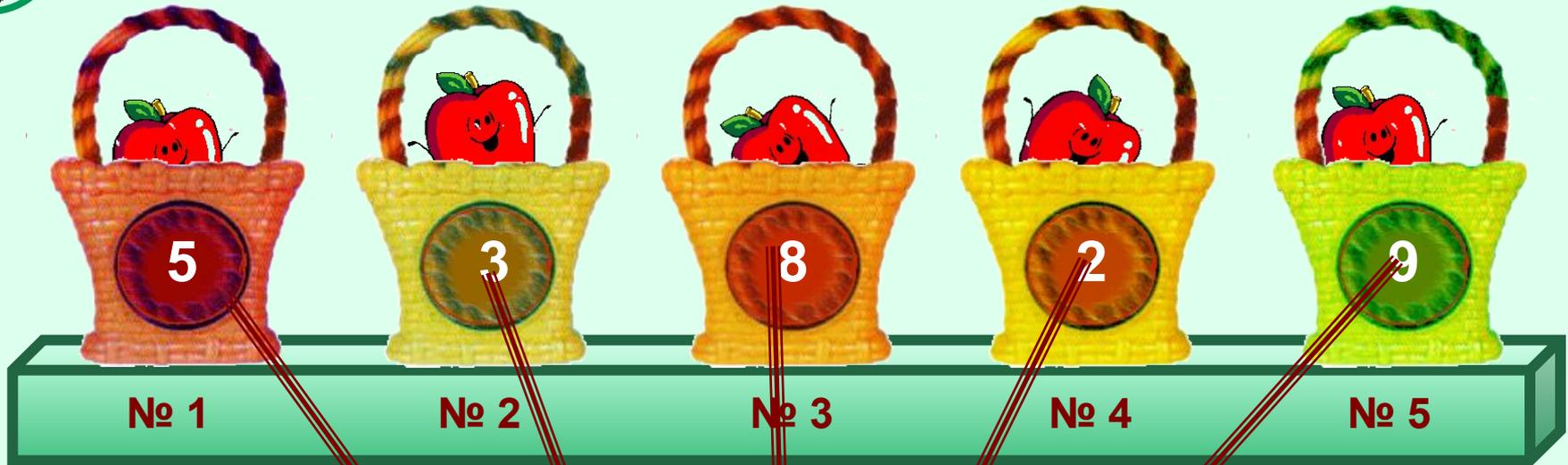
Алгоритм:

Суммирование элементов массива

- В цикле со счетчиком, поэлементно суммируем (добавляем) все элементы массива в переменную S.



Суммирование элементов массива



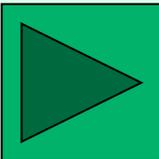
Сумма = $5 + 3 + 8 + 2 + 9 = 27$

S

N:=5;

For I:= 1 to N do

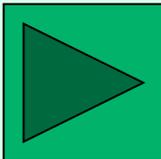
S:=S+ (A[I]);



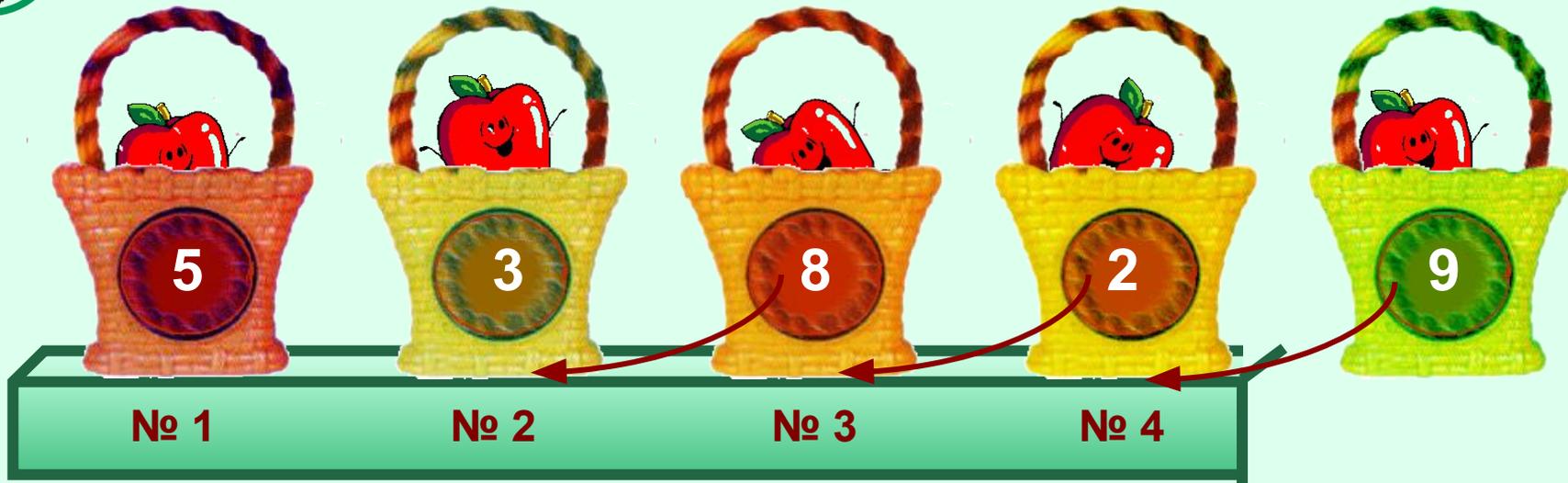
Алгоритм:

Удаление K-го элемента из массива

- ввести номер удаляемого элемента k ;
- в цикле со счетчиком начиная с k -го элемента до $N-1$ элемента - сдвинуть все элементы, на один элемент влево.

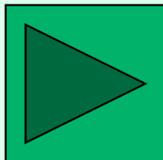


Удаление элемента из массива



$N := N - 1; k := 2;$
 For $j := k$ to N do
 $A[j] := A[j + 1];$

l	1	2	3	4
A[l]=	5	3	8	2

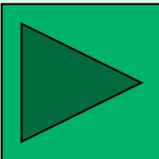


Алгоритм:

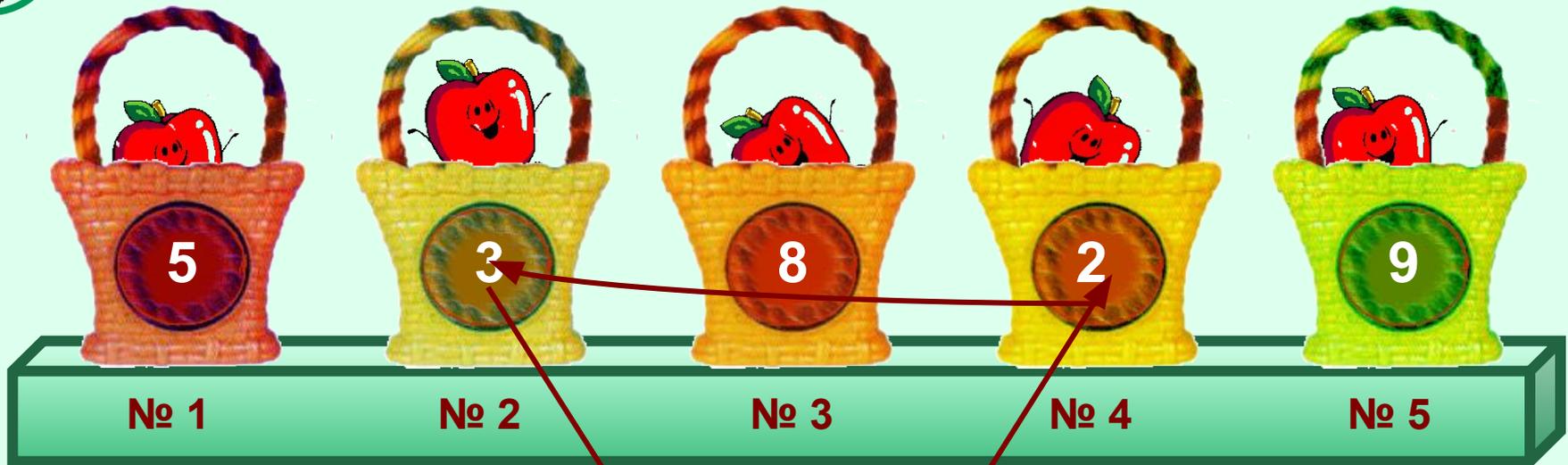
Перестановка элементов в массиве

Два элемента массива k -ый и J -ый поменять местами.

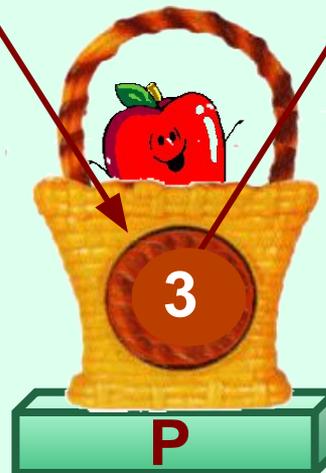
- дополнительной переменной P присвоить значение k -го элемента;
- k -му элементу присвоить значение J -го элемента;
- J -му элементу присвоить значение дополнительной переменной.



Перестановка 2-х элементов в массиве



№ элемента	1	2	3	4	5
Исходный массив	5	3	8	2	9
Полученный массив	5	2	8	3	9

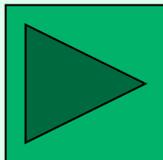


$k := 2 ; J := 4 ;$

$P := A [K] ;$

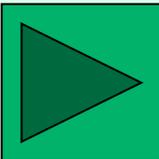
$A [K] := A [J] ;$

$A [J] := P ;$

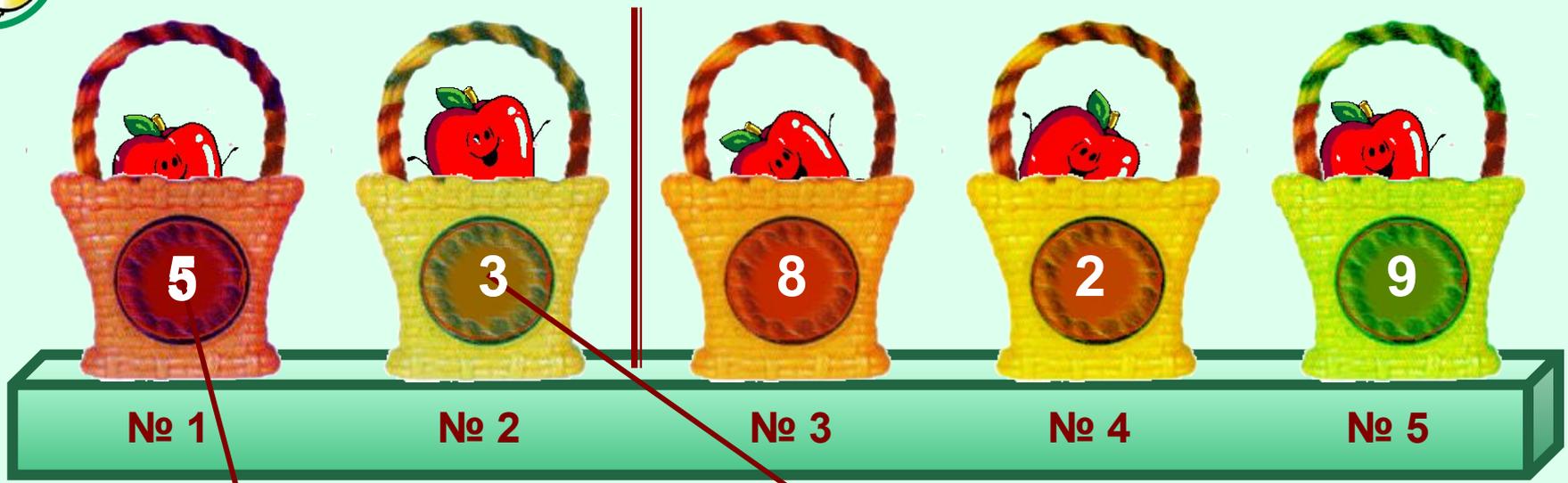


Алгоритм: Поиск Min

- присвоить переменной Min значение первого элемента.
- в цикле со счетчиком, начиная со второго элемента до последнего сравнить каждый элемент с Min.
- если элемент \leq Min, его значение присваивается переменной Min.

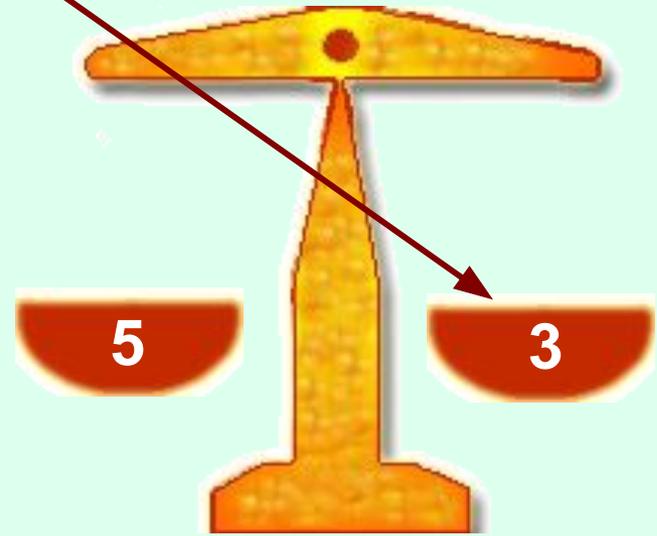


Поиск минимального элемента

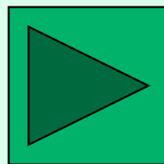


MIN := 5 3

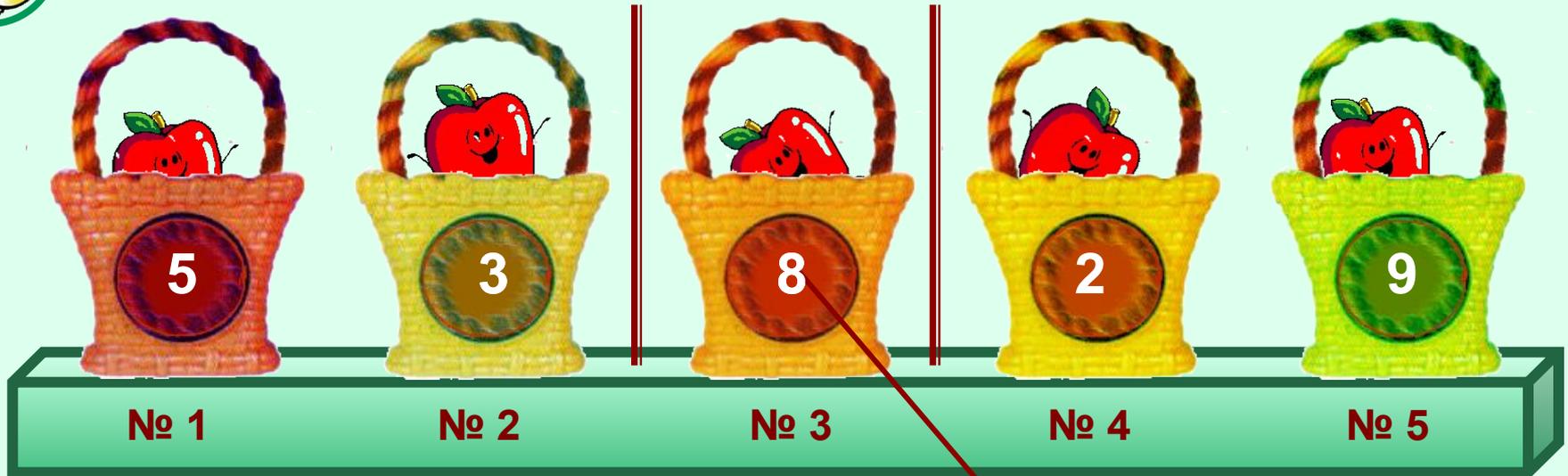
Min := A [2]



Шаг 1



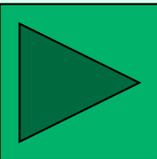
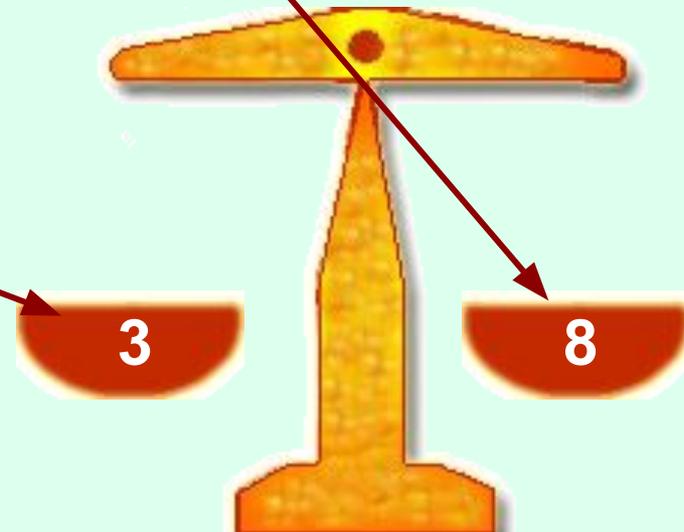
Поиск минимального элемента



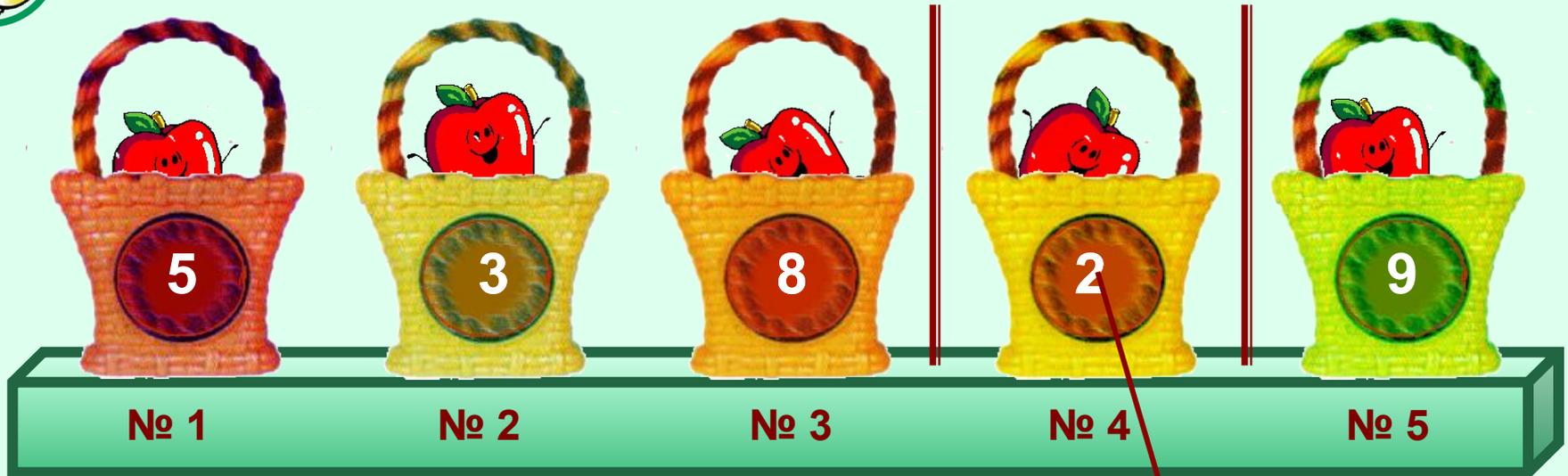
MIN := 3

Min := A [2]

Шаг 2



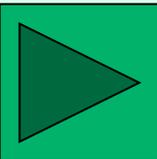
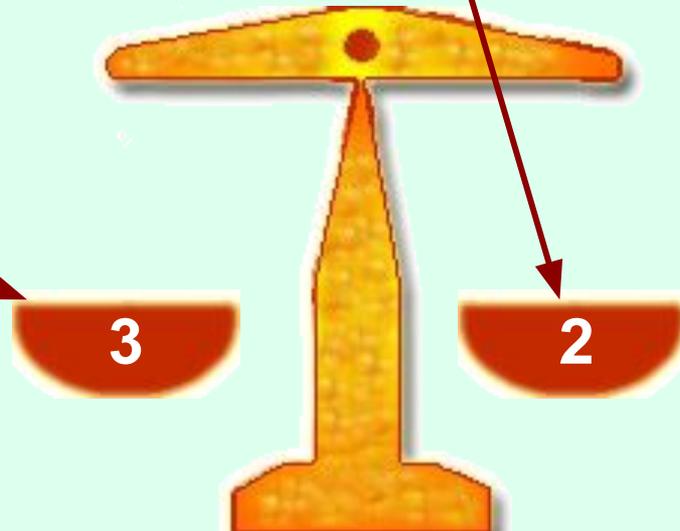
Поиск минимального элемента



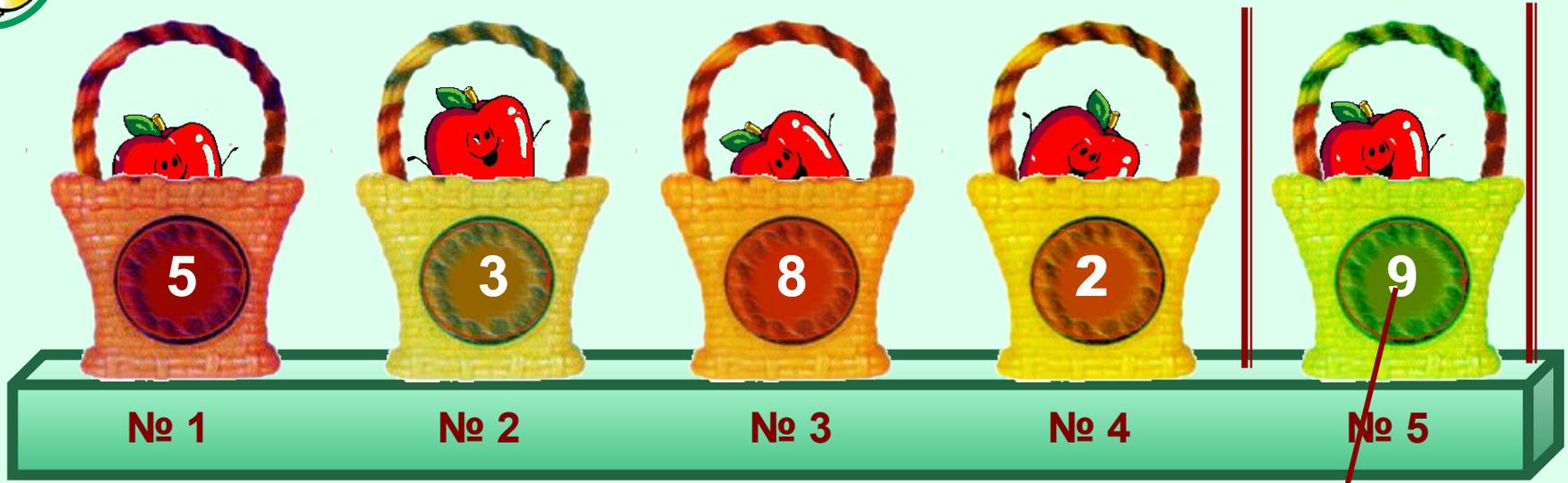
MIN := 2

Min := A[2]

Шаг 3



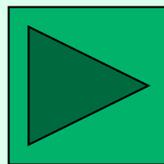
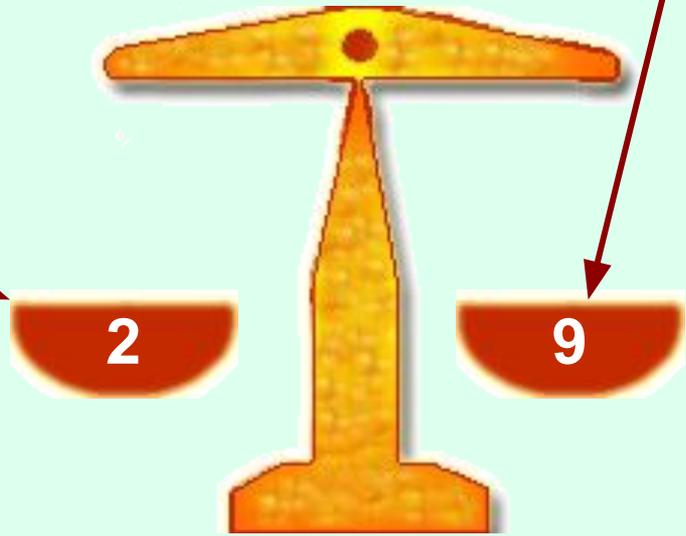
Поиск минимального элемента



MIN := 2

Min := A[4]

Шаг 4

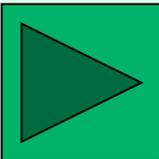


Определите какой из типов алгоритмов может быть использован для решения задачи:

Информация о **количестве** выпадавших в течение месяца **осадков** задана в виде массива.

Определить общее количество осадков за месяц.

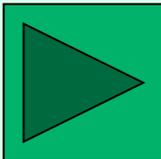
Сумма элементов массива



Определите какой из типов алгоритмов может быть использован для решения задачи:

Информация о **температуре воздуха** за месяц задана в виде массива. Определить, максимальную, минимальную и среднемесячную температуру воздуха.

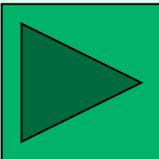
Поиск Max, Min и среднего арифметического элементов массива.



Определите какой из типов алгоритмов может быть использован для решения задачи:

В классе **ученики** рассажены за партами в определенном порядке. Необходимо **поменять** местами самого высокого ученика и самого низкого ученика.

Поиск Max, Min и перестановка элементов в массиве

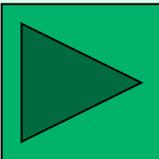


Определите какой из типов алгоритмов может быть использован для решения задачи:

При **поступлении на профиль** «Программирование» в ПК вводится фамилия ученика и входной балл тестирования.

Выяснить **проходной балл** и составить список поступивших учеников. (Все, кто набрал балл ниже проходного в список не попадают)

Поиск среднего и удаление элементов из массива, не удовлетворяющих условию.



Группы типичных ошибок в работах уч-ся по теме “Массивы”



1. Алгоритмические:

- отсутствие составного оператора
- отсутствие вывода результата
- нарушение порядка действий в алгоритме

2. Синтаксические ошибки:

- неверная запись ключевых слов языка Pascal
- неверно расставлены разделители(точка с запятой, запятая, двоеточие)

3. Ошибки описания:

- констант, переменных, массивов
- описание не всех переменных

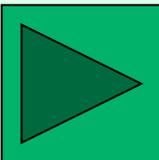
4. Несогласование типов переменных

5. Ошибки имен переменных и программ

6. Ошибки в оформлении процедур ввода/вывода

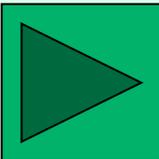
7. Неверная организация цикла с параметром

8. Неверное использование условного оператора



Найти ошибки

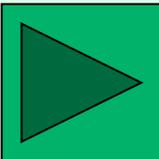
1. Program SR;
2. Var I,S:real;
3. Kol : integer;
4. B: array [1...N] of real;
5. BEGIN
6. S:=0; Kol := 0;
7. For I=1 to N do;
8. read (A[I]);
9. if A[I]>0, then
10. S:=S+A[I]; Kol:=Kol+1;
11. SR:=S/Kol
12. Writeln('ср.арифметическое=', SR)
13. END.



Проверь себя! Вам нужно:

- Получить по почте задания.
- Получить по почте таблицу ответов.
- Выполнить задания, заполняя таблицу ответов (по технологии Верю не Верю).
- Отправить таблицу ответов на ПК учителя.

Удачи!





-
- Спасибо за внимание