

# Задание по теме: Детская считалка

Выполнили  
Студенты группы Б8219а  
Демченко Анастасия  
Пакичев Тимофей

# Выбор структуры данных

□ Требуется динамическая структура данных, которая обеспечила бы возможность:

□ Задать некоторую линейную последовательность однотипных элементов(участников считалки)

□ Получать доступ к следующему элементу за текущим(каждому следующему слову считалки - соответствует следующий участник)

□ Иметь доступ от последнего элемента к первому (так как участники стоят по кругу)

□ Иметь возможность простого и быстрого удаления элемента (проигравший выбывает)

Связный список

Циклический  
связный список

Двунаправленный  
циклический  
связный список

# Алгоритм(краткое описание)

1. Добавляем необходимое количество элементов в список
  2. Пока не кончится считалка: считываем очередное слово считалки и переходим к следующему элементу списка
  3. Удаляем элемент на котором закончилась считалка, возвращаемся в начало считалки
  4. Повторяем 2-3 пока не останется только один элемент в списке
- Сложность по времени  $O(k*n)$ , где  $k$  - количество слов в считалке,  $n$  - число участников. Сложность по памяти  $O(n)$

# Алгоритм

1 Процедура Добавление()

2 Начало процедуры

3 Создать новый элемент списка T

4 Переменной T.name присвоить значение переменной Name

5 Если значение указателя A равно nil, то:

5.1 Указателю A присвоить значение T

5.2 Указателю T.next присвоить значение T

5.3 Указателю T.prev присвоить значение T

6 Иначе:

6.1 Указателю T.next присвоить значение A.next

6.2 Указателю A.next.prev присвоить значение T

6.3 Указателю T.prev присвоить значение A

6.4.Указателю A.next присвоить значение T

7 Конец процедуры

8 Процедура Удаление(входные данные:P - указатель)

9 Начало процедуры

# Алгоритм(продолжение)

14 Начать работу алгоритма

15 Присвоить значение nil указателю A.

16 Вывести сообщение - "Хотите добавить нового участника?"

17 Считать ответ пользователя в переменную Answer

18 Если значение переменной Answer = "да" то:

18.1 Вывести сообщение - "Введите имя участника"

18.2 Считать ответ пользователя в переменную Name

18.3 Вызвать процедуру Добавление()

18.4 Вернуться к пункту 16

19 Вывести сообщение - "Выберите номер считалки"

20 Считать в переменную number номер считалки

21 Открыть текстовый файл соответствующий номеру считалки

22 Пока A.next<>A делать:

22.1 Считать очередное слово считалки

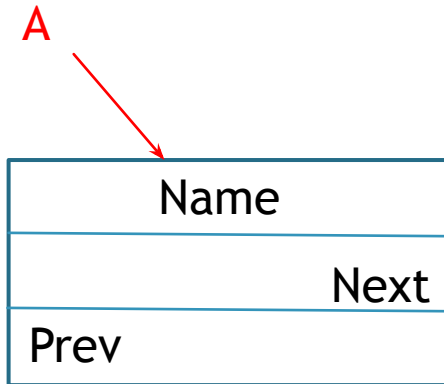
22.2 Если встречен конец текстового файла со считалкой, то:

22.2.1 Указателю A присвоить значение A.next

22.2.3 Вывести сообщение "A.prev.name выбывает"

22.2.4 Вывести сообщение "Участник (A.name) выбывает"

# Пример



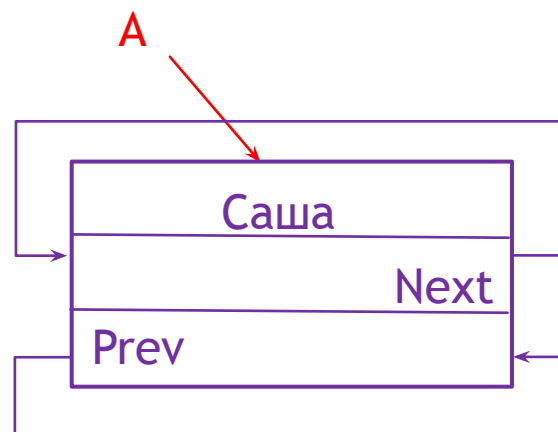
Name - имя участника

Next - ссылка на следующего участника

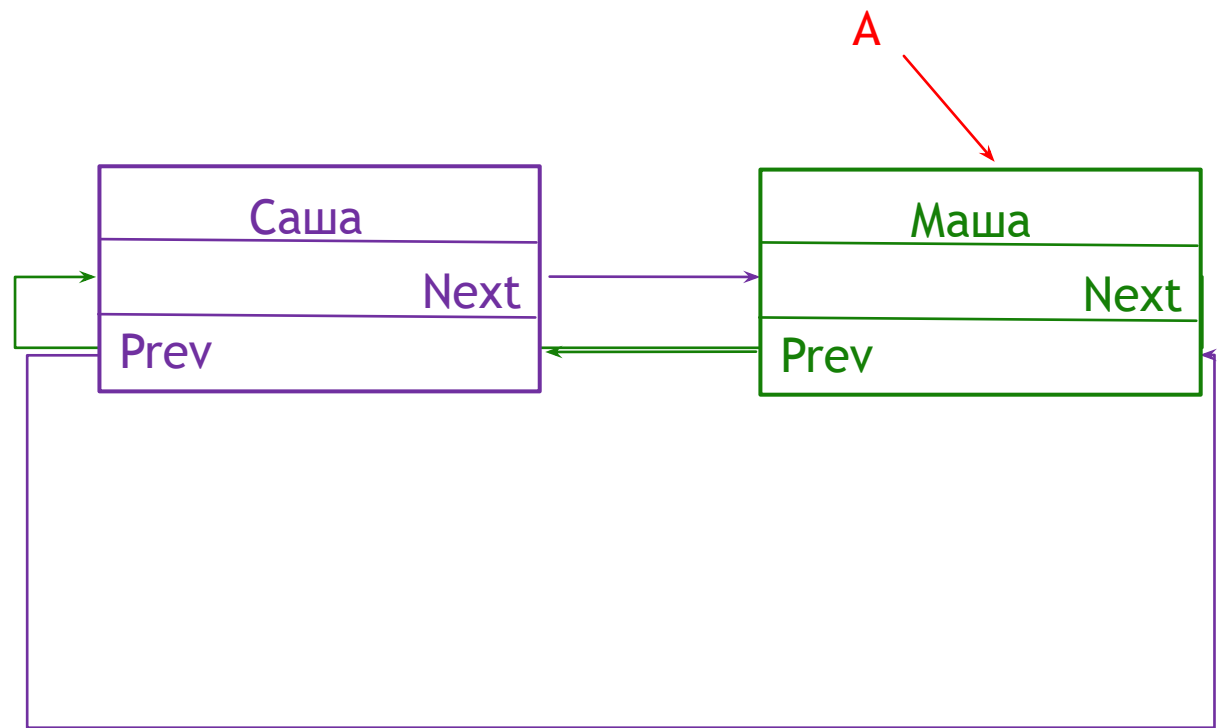
Prev - ссылка на предыдущего участника

A - указатель на текущий элемент

Добавляем 1й элемент в список.

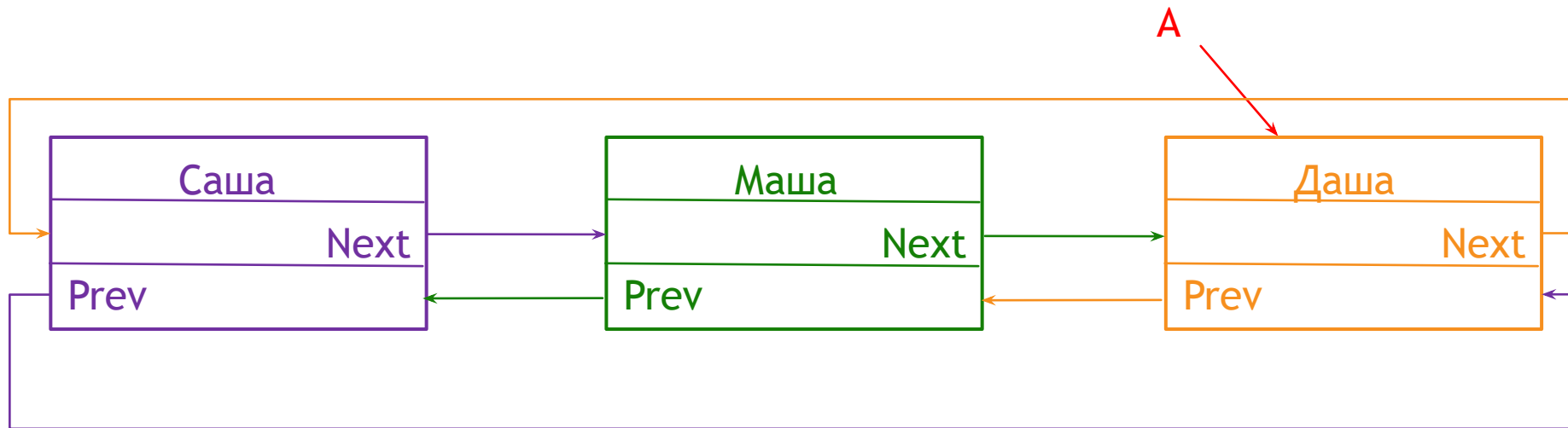


Добавляем 2й элемент в список.



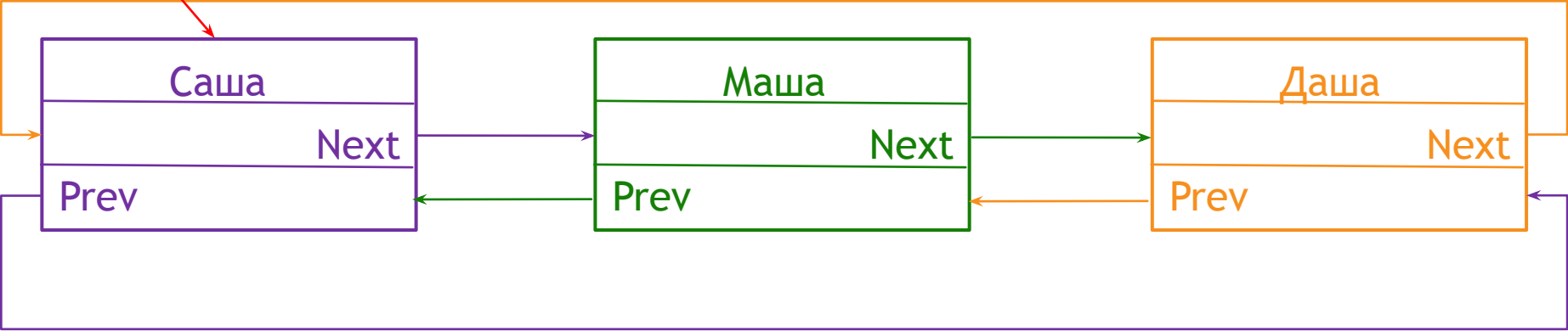


Добавляем 3й элемент в список.

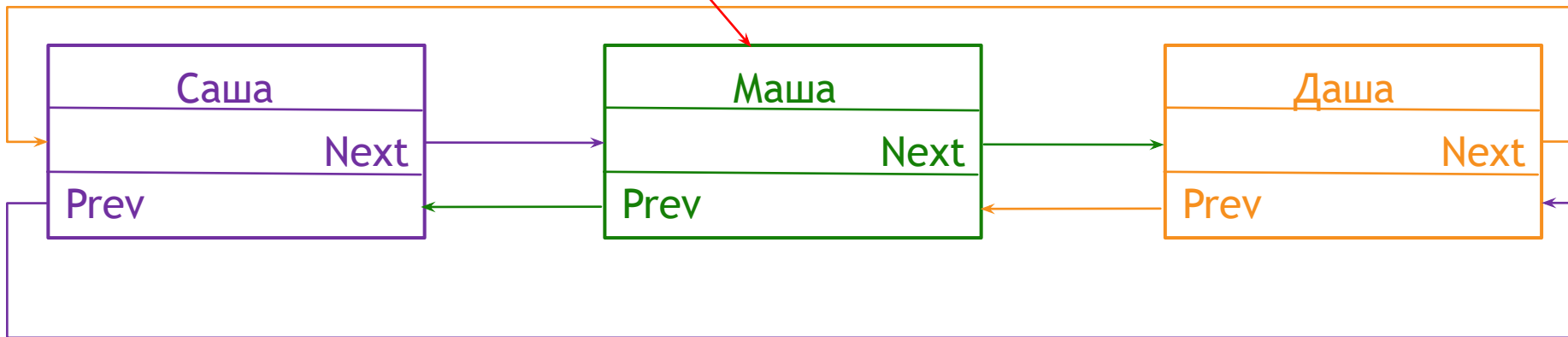


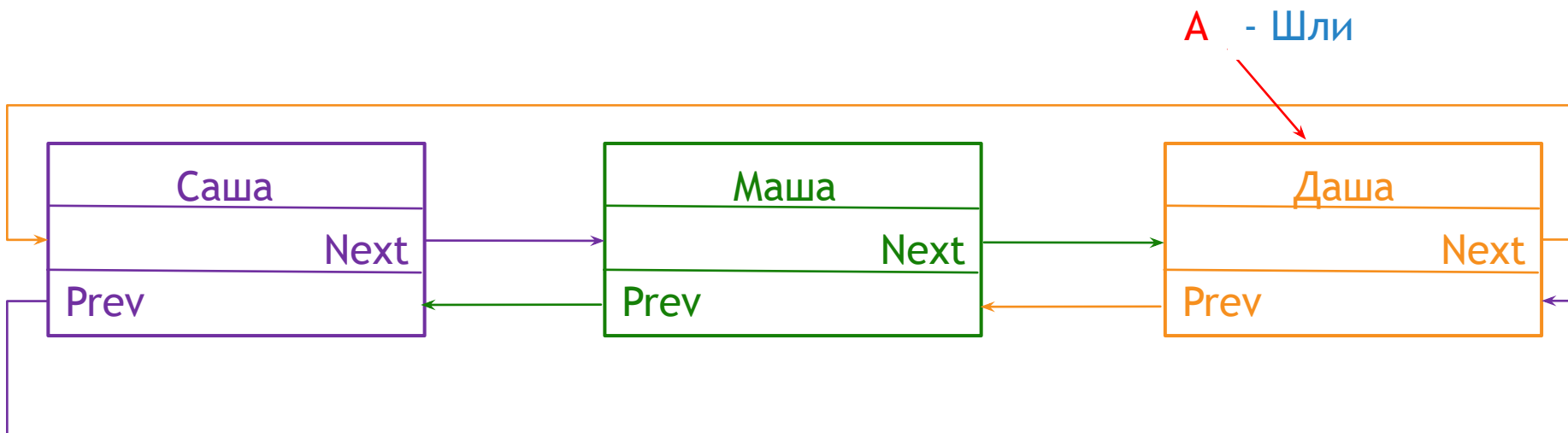
Запускаем считалку

A - Аты

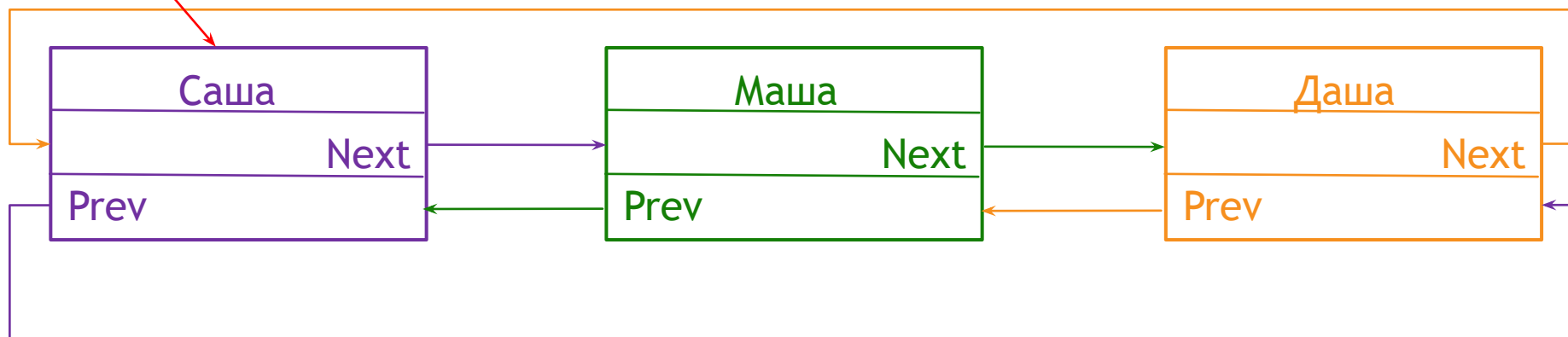


A - Баты

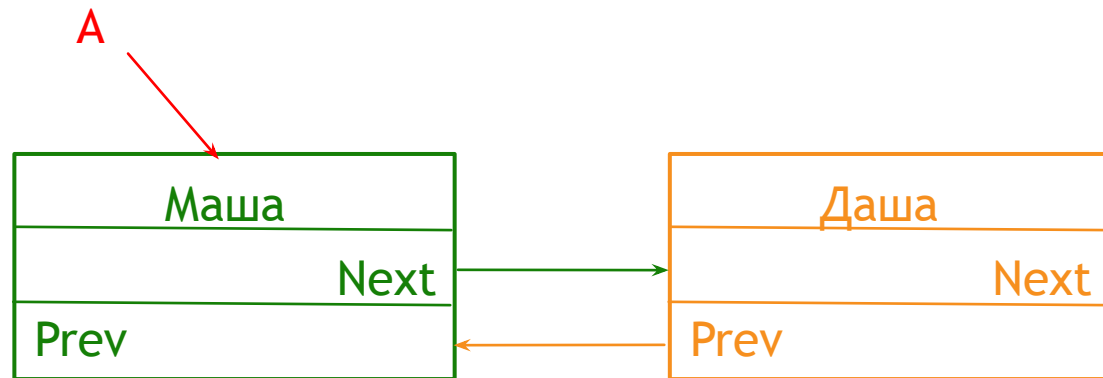
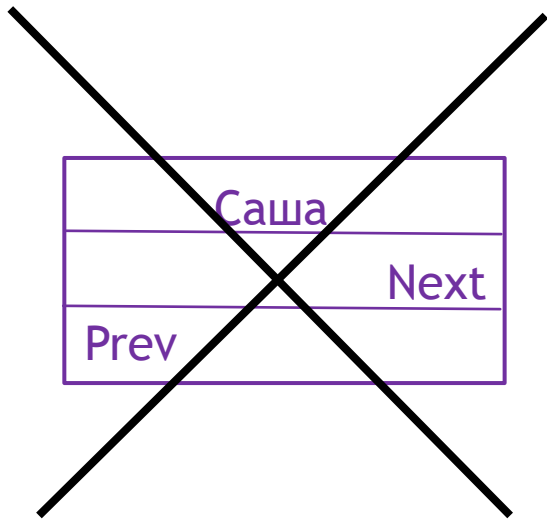




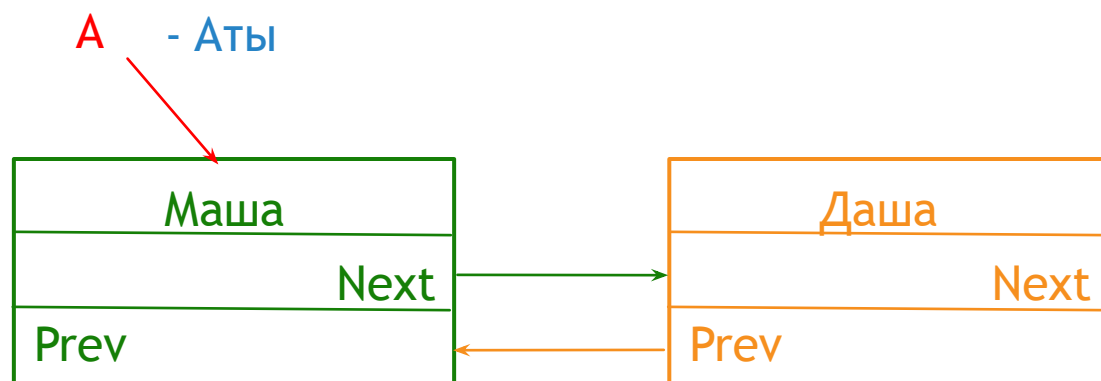
A - Солдаты

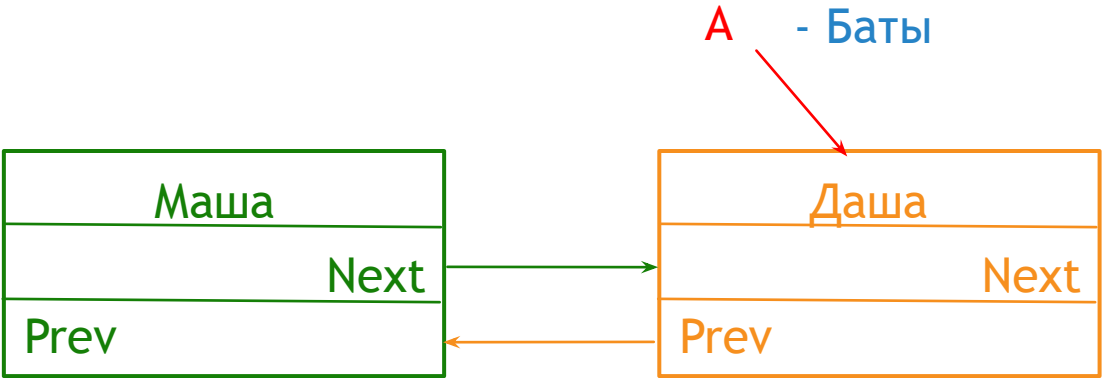


Удаляем элемент, на котором закончилась считалка



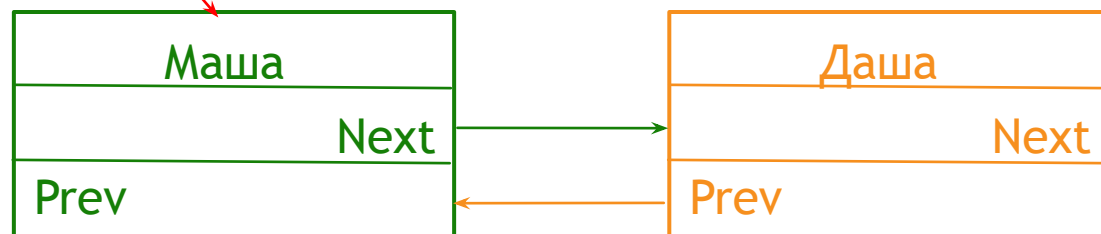
Так как число участников  $>1$ , снова запускаем считалку

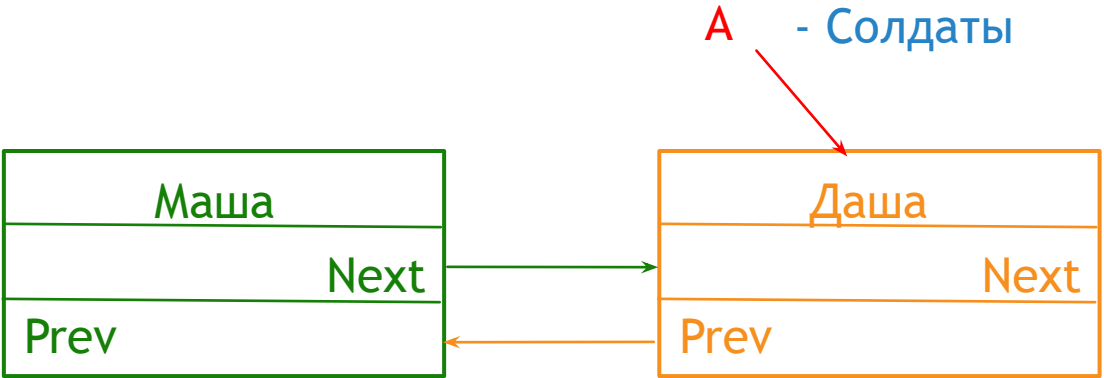




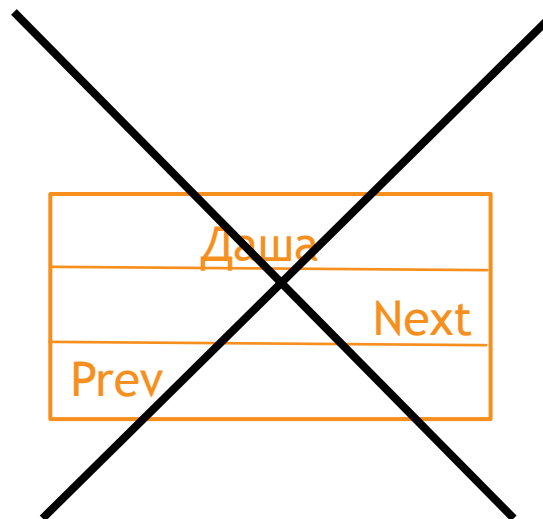
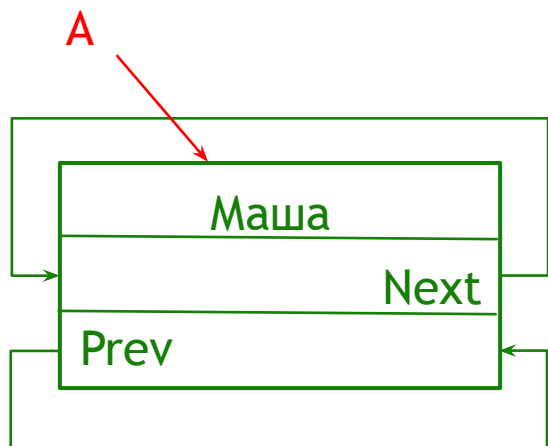


A - Шли

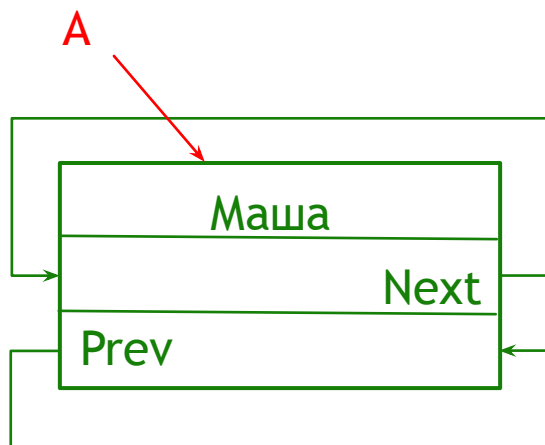




Удаляем элемент, на котором закончилась считалка



Так как остался всего один участник, он объявляется победителем



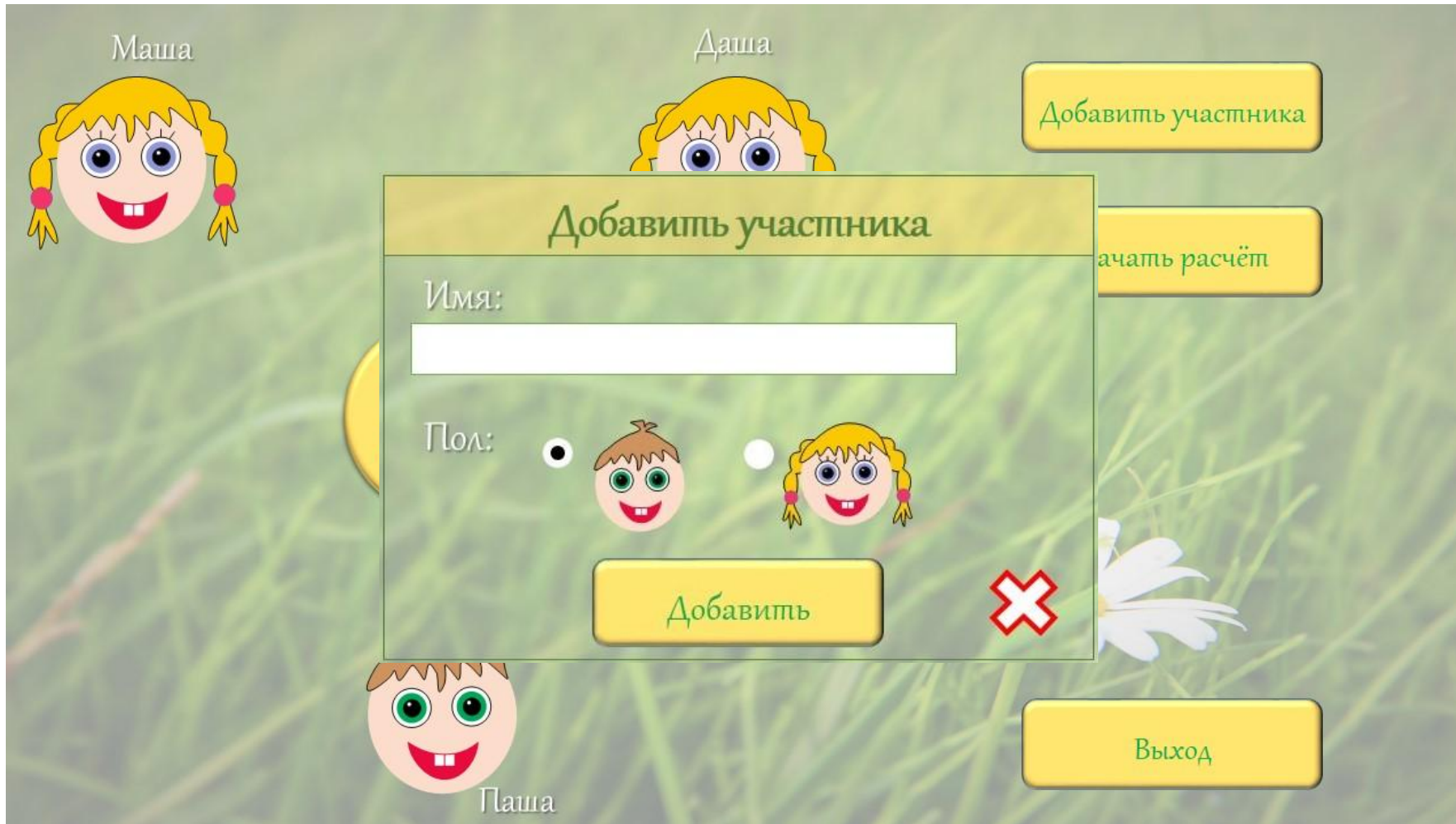
Победитель - Маша

Выбывшие - Саша, Даша

# Интерфейс: выбор параметров расчёта



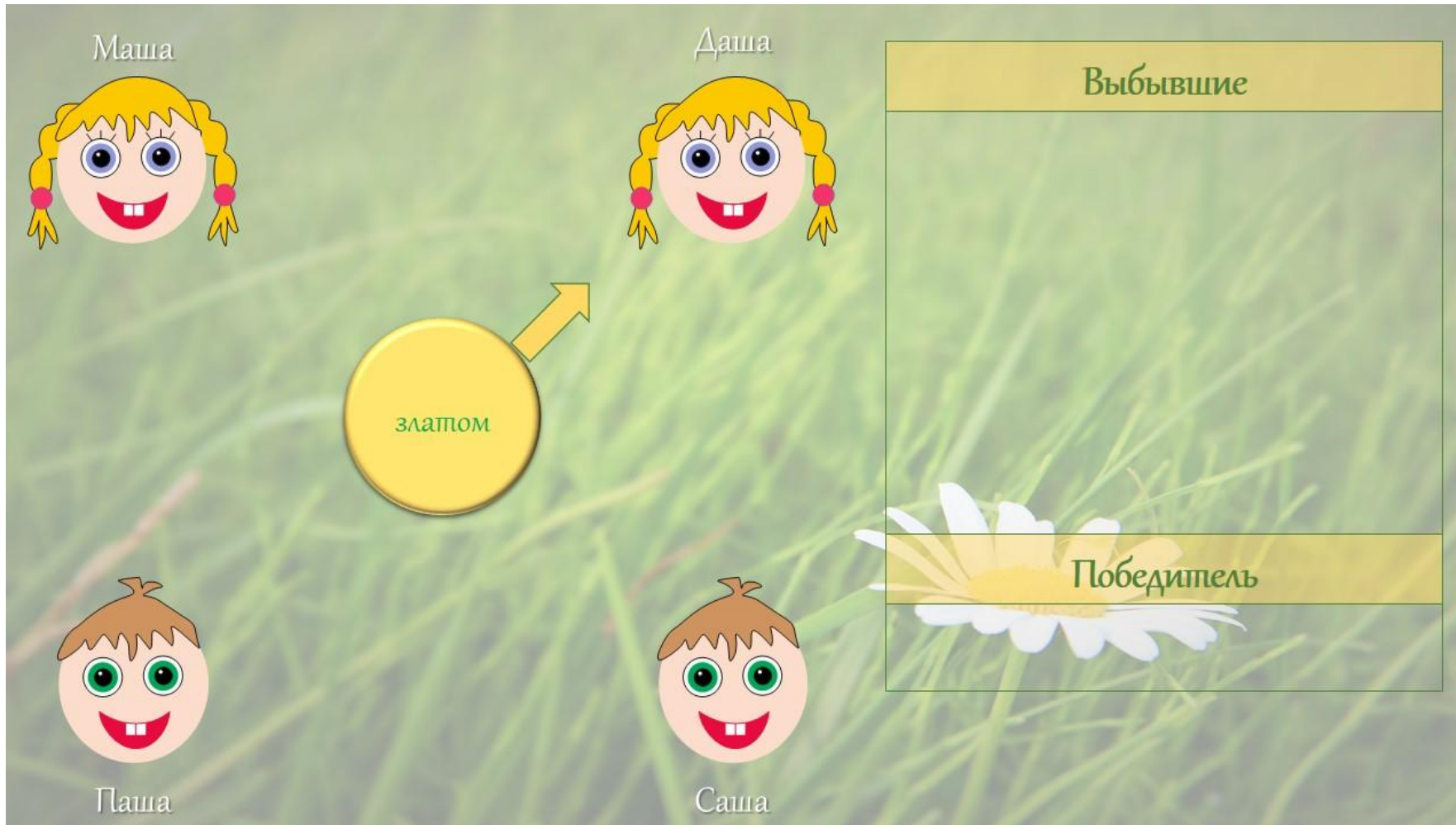
# Интерфейс: Добавление участника



# Интерфейс: Выбор считалки

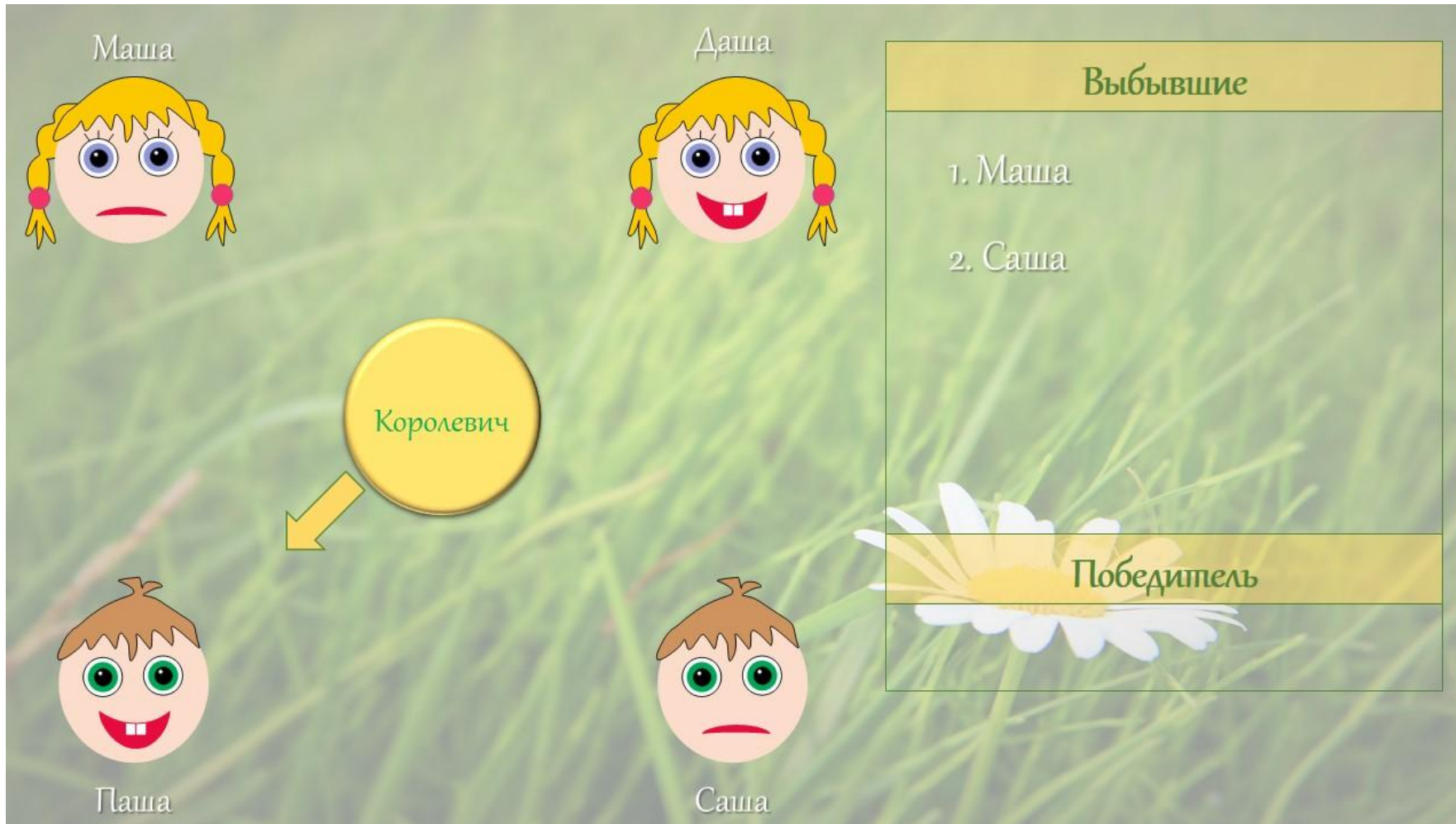


# Интерфейс: Процесс расчёта





# Интерфейс: Процесс расчёта



# Интерфейс: Определение победителя

