5.2. Булеви или логически изрази в реалния свят

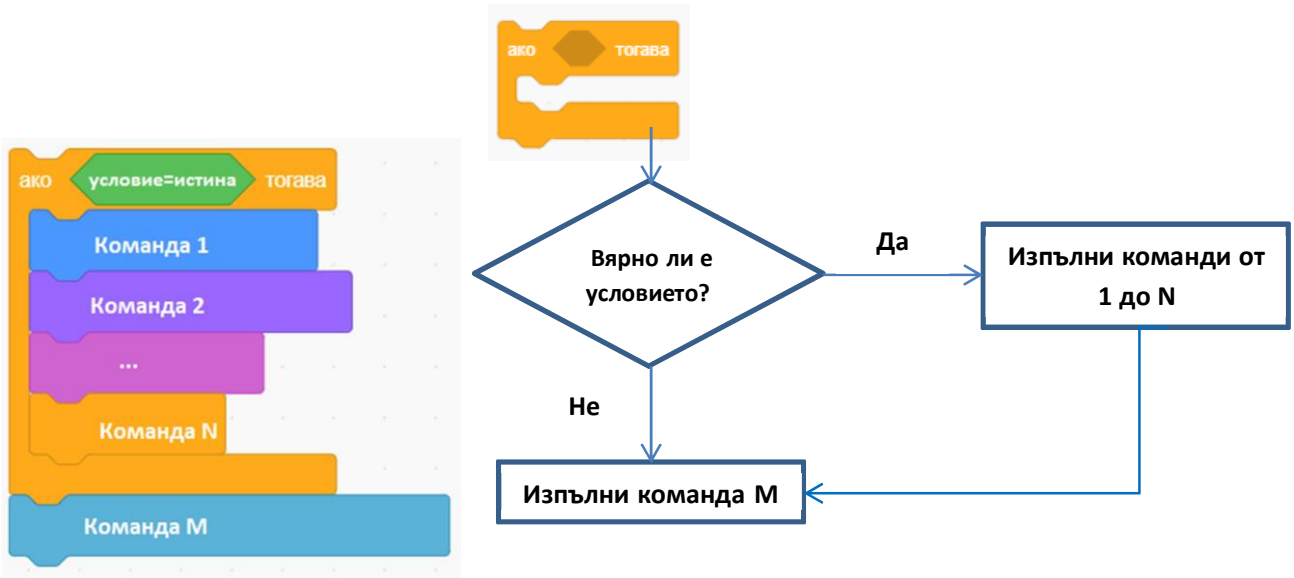
През 19-ти век е живял Джордж Буле, британски математик, който разработва логическа система, основана на две значения: 1 и 0 (вярно и невярно). Булевата алгебра става основа на съвременната компютърна наука.
В ежедневието си ние постоянно използваме булеви изрази за вземане на решения. Компютрите също ги използват, за да определят посоката, която програмата ще следва. В програмирането въпросите, които имат само два варианта на отговор, се наричат логически изрази. За компютъра вярното изпълнение на условието винаги е истина (true), а грешното – лъжа (false).
Роботизираната ръка може да бъде програмирана да проверява движещите се части на конвейер и да ги придвижва в кошница 1, когато доброто качество = истина и тези в кошница 2, за които доброто качество = лъжа.
Системите за домашна сигурност обикновено са програмирани така, че когато се въведе грешен код, алармата се включва (верния код = false), а при правилен код – се изключва (correctCode = true).
Един отдалечен сървър може да предостави или откаже достъп до кредитната Ви карта в магазин, в зависимост от това дали картата е валидна (true) или не (false).
Компютърът в колата автоматично ще задейства въздушните възглавници, ако реши, че е настъпил сблъсък (сблъсък = true).
Мобилният телефон може да показва специална икона, когато батерията е ниска (батерията е разредена = true) и да я скрие, когато зарядът е достатъчен (батерията е изтощена = false).

Това са няколко примера за това как компютрите реализират различни действия, като проверяват резултатите от булевите условия.

Структури за вземане на решения

Кодовете, които съдържат условие и действие за верен или грешен отговор се наричат **разклонен алгоритъм**.
Разделът „Контрол“ в Scratch съдържа два такива кода, които Ви позволяват да вземате решения и да управлявате действията в програмите си: ако и ако/иначе. Използвайки тези блокове, можете да зададете въпрос и да предприемете действие въз основа на отговора.

Код „Ако“
Този код е кратък блок за разклонение, който съдържа условие (ако) и действие, което се изпълнява само при верен отговор (тогава).



Фиг. 3 Структура и схема на код „Ако“

Код „Ако/иначе“

Това е пълен блок за разклонение, който съдържа условие (ако), действие при верен отговор (тогава) и действие при грешен отговор (иначе).

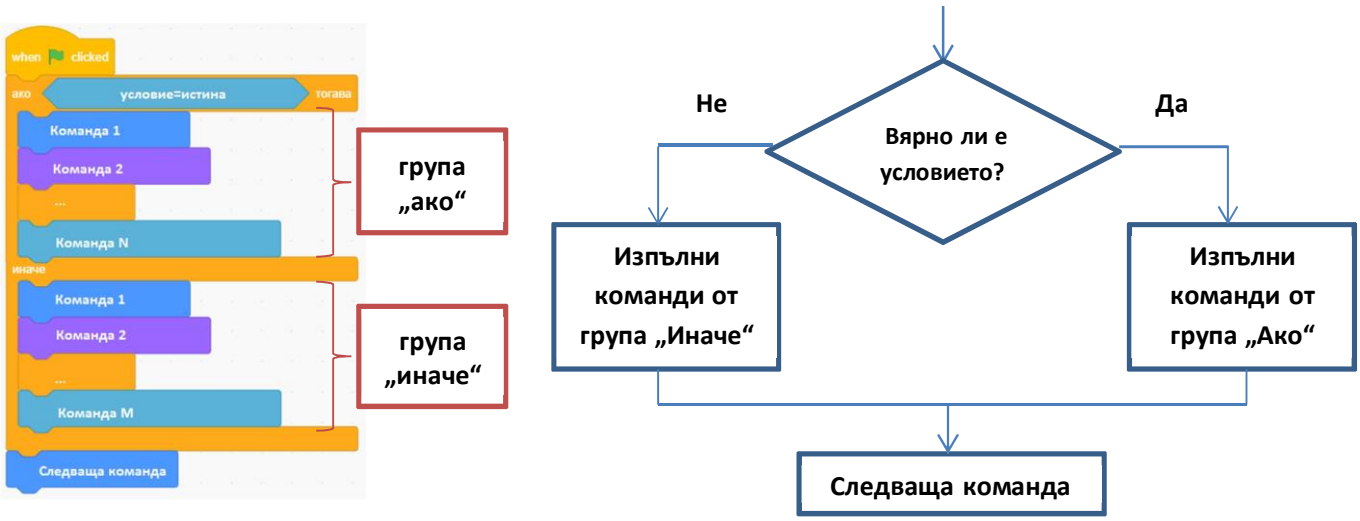
Представете си, че трябва да създадете игра, която да обучи учениците от началното училище на основите на математиката. Играта предлага задачата за добавяне, след което иска да въведете отговор. Ученикът трябва да получи точка за правилен отговор или да загуби точка за грешен отговор. Можете да изпълните тази задача с два блока „ако“:

Ако отговорът е правилен, добавете една точка към резултата.
Ако отговорът е неправилен, извадете една точка от резултата.

Можете също така да опростите тази логика и да направите кода по-ефективен, като комбинирате двата блока в един и той е „ако/иначе:

Ако отговорът е правилен, добавете една точка към резултата.
В противен случай извадете една точка от резултата.

Избраното условие се тества. Ако е вярно, се изпълняват командите от блока „ ако“. Ако не е, тогава се изпълняват командите блок „иначе“. Програмата ще изпълни само една група блокови команди. Блок структурата и схемата на „ако/иначе“ е показана на фиг. **4.**



Фиг. 4 Структура и схема на код „Ако/иначе“