

# Przewodnik po makrach w aplikacji SUDOKU

## 1. Lista wszystkich wykorzystanych makr i funkcji

### a. Moduł małe2

- Sub wypisz()
- Sub zeruj9()
- Function sw(ByVal w, ByVal c As Integer) As Boolean
- Function sk(ByVal k, ByVal c As Integer) As Boolean
- Function sq(ByVal w, ByVal k, ByVal c As Integer) As Boolean
- Function zrob(ByVal aktualny, ostatni) As Boolean
- Sub licz9()

### b. Moduł małe

- Sub start9()
- Sub wyniki()
- Sub od\_nowa()
- Sub start\_moje()
- Sub podaj()
- Sub przepisz9()
- Function wiersz(ByVal w As Integer, ByVal c As Integer)
- Function kolumna(ByVal k As Integer, ByVal c As Integer)
- Function kwadrat(ByVal w, ByVal k, ByVal c As Integer) As Boolean
- Sub sprawdz\_w\_trakcie9()
- Function sprawdz() As Boolean
- Sub zakoncz9()
- Sub odpowiedz9()
- Sub losuj\_szablon9()
- Sub czyste\_wyniki()
- Sub pokaz\_blad9()

### c. Moduł duże2

- Sub wypisz2()
- Sub zeruj2()
- Function sw2(ByVal w, ByVal c As Integer) As Boolean
- Function sk2(ByVal k, ByVal c As Integer) As Boolean
- Function sq2(ByVal w, ByVal k, ByVal c As Integer) As Boolean
- Function zrob2(ByVal aktualny, ostatni) As Boolean
- Sub licz16()

### d. Moduł duże

- Sub start16()
- Sub start\_moje16()
- Sub przepisz()
- Sub sprawdz\_w\_trakcie()
- Sub zakoncz16()
- Sub odpowiedz()

- Sub losuj\_szablon16()
- Sub pokaz\_blad16()
- Function wiersz(ByVal w As Integer, ByVal c As Integer)
- Function kolumna(ByVal k As Integer, ByVal c As Integer)
- Function kwadrat(ByVal w, ByVal k, ByVal c As Integer) As Boolean
- Function sprawdz() As Boolean

2. W jakiej kolejności wykonują się makra:

2.1 jeżeli decydujemy się wygenerować zadanie wybierając poziom trudności i rozmiar

START

Granatowy guzik START w arkuszu Powitalnym wywołuje makro **start9** makro sprawdza, jaki był zadany rozmiar zagadki. Jeśli był 9x9, to następnie czyści tablicę w zakładce „Zad 9x9”, wywołuje makro **licz9**, które wypełnia pustą tablicę używając funkcji i makr z modułu male2, potem używając procedury **losuj\_szablon9** losuje układ, według którego poda zagadkę, po czym makro **przepisz9** uzupełnia tablicę z zagadką w arkuszu „Sudoku 9x9” i tu użytkownik przejmuje program we władanie.



#### ARKUSZ SUDOKU 9x9

guzik Powrót - makro **od\_nowa**, które odsyła do arkusza Powitalnego,

guzik Nowe SUDOKU – makro **przepisz9**, które po raz kolejny przepisuje zadanie według szablonu. Ta opcja jest użyteczna, gdy popełnimy błąd i chcemy spróbować drugi raz,

guzik Podpowiedź – makro **podpowiedz9**, które dodaje kolejną liczbę z rozwiązania do arkusza użytkownika,

guzik Sprawdź – makro **sprawdz\_w\_trakcie9**, które wywołuje funkcje wiersz, kolumna i kwadrat dla każdego obszaru i każdej liczby, i sprawdza, czy w rozwiązaniu istnieje sprzeczność,

guzik Pokaż błąd – makro **pokaz\_blad9**, które w razie wystąpienia sprzeczności w tablicy koloruje odpowiednie pola na żółto (bug wykryty na zajęciach został poprawiony – problem wynikał z nieprawidłowego wyszukiwania winnych komórek w danym obszarze (wyszukiwał od 1 do 9, a powinien 3 do 11, by adresy zgadzały się z planszą),

guzik Zakończ – makro **zakoncz9**, wywołuje funkcję sprawdz, która decyduje, czy zadanie jest rozwiązane do końca, a następnie na żądanie użytkownika dopisuje kolejny rząd do tabeli wyników w arkuszu Wyniki,





## ARKUSZ WYNIKI

guzik Nowe SUDOKU – makro **od\_nowa**, powrót do arkusza Powitalnego,

guzik Wyczyść wyniki – makro **czyste\_wyniki**, czyszczenie tabelki z wynikami,

guzik wykres – żadne makro. Jest tu dla symetrii ☺,

## 2.2 Jeżeli zdecydujemy się podać własne zadanie z zewnątrz

W arkuszu Powitalnym wybieramy rozmiar i klikamy guzik Własne sudoku, który uruchamia makro **podaj**, które sprawdza wpisany przez nas rozmiar i przenosi nas do odpowiedniego arkusza („Zadanie 9x9” lub „16x16 Zadanie”). Czerwony guzik START uruchamia makro **start\_moje**, które kopiuje wpisane przez nas zadanie do arkuszy „Szablon 9x9” i „Zad 9x9”. Następnie uruchamiane jest ponownie makro **licz9**, które rozwiązuje zadanie, i makro **przepisz9**, które przenosi zadaną zagadkę do arkusza dla użytkownika z odpowiednim formatowaniem. Dalej procedura przebiega analogicznie jak w przypadku wygenerowanego zadania. Przy wyborze rozmiaru 16x16 żółty guzik START w arkuszu „16x16 Zadanie” uruchamia makro **start\_moje16** i po kolei procedury generowania i przepisywania zadania, **licz16** i **przepisz16**.

## 2.3 jeżeli wybieramy generowanie zadania 16x16

Granatowy guzik START w arkuszu powitalnym odwołuje się do makra **start9**, które sprawdza warunek rozmiaru, i przenosi nas do modułu duże, gdzie uruchamiane jest makro **start16**. Tak jak w przypadku mniejszej wersji, czyszczona jest ewentualna zawartość arkusza „16x16 zad”, następnie wywoływane jest makro **licz16**, **losuj\_szablon16** i **przepisz**. Jesteśmy w arkuszu „16x16 Sudoku”, gdzie użytkownik ma możliwość zmierzenia się z dużą zagadką.

### ARKUSZ 16x16 SUDOKU

Guziki działają analogicznie do tych z zadania 9x9:

Powrót – makro **od\_nowa**, przeniesienie do arkusza Powitalnego

Zadanie – makro **przepisz**, wyczyszczenie tablicy i wpisanie zadania ponownie

Sprawdź – makro **sprawdz\_w\_trakcie**, szukanie sprzeczności

Podpowiedz – makro **podpowiedz**

Pokaż błąd – makro **pokaz\_blad16**

Zakończ – makro **zakonc16**, które wywołuje funkcję sprawdz (potwierdzenie poprawnego rozwiązania zagadki), i na żądanie użytkownika zapisuje czas w zakładce Wyniki



## 2.4 Moduły małe2 i duże2

Te moduły omawiane są równocześnie, gdyż ich struktura i działanie jest prawie całkowicie identyczne. Różnią się jedynie „długością” pętli i liczbą małych kwadratów, co jest dobrze widoczne w kodzie.

Zwykły użytkownik nie ma bezpośredniego dostępu do makr z tych modułów. Ich wywoływanie opisane jest wyżej. W domyślnie ukrytych zakładkach „rozwiązywanie 9x9” i „16x16 rozw” są tablice, do których wpisywane jest rozwiązanie wygenerowane przez algorytm rekurencyjny. W tych arkuszach są też opcje omawiane w czasie prezentacji:

- wyświetlanie: 0 dla aktualizacji jedynie komórki „który”, 1 dla aktualizacji całej tabeli postępów w pracy algorytmu
- liczba ruchów: charakterystyczna dla rozmiaru planszy sudoku: 82 dla 9x9 i 257 dla 16x16
- losowo: 0 jeśli kolejne liczby w tablicy mają być dostawiane po kolei (wstaw 1, jeśli nie możesz, wstaw 2, etc.), lub 1, gdy ich kolejność ma być losowa (wektor a wewnątrz funkcji zrob i zrob2)

Oraz tabelka postępów:

- aktualny, czyli obecnie wykonywany ruch
- ostatni, czyli liczba ruchów do wykonania
- w, indeks wiersza aktualnie badanej komórki (wyliczany w funkcji zrob lub zrob2 na podstawie zmiennej aktualny)
- k, indeks kolumny aktualnie badanej komórki (wyliczany w funkcji zrob lub zrob2 na podstawie zmiennej aktualny)
- max, czyli najwyższy osiągnięty ruch. Jak daleko algorytm się posunął w rozwiązywaniu, zanim musiał się wycofać po niemożliwości wykonania kolejnego ruchu.
- który, czyli licznik prób dostawienia kolejnej liczby do tablicy

Guziki LICZ w arkuszach wywołują makra **licz9** lub **licz16**, w zależności od tego, w którym z arkuszy się znajdujemy.

Makro na początku wywołuje procedurę **zeruj** lub **zeruj2**, która czyści tablice roboczą szach przed rozpoczęciem rozwiązywania, następnie wywołuje rekurencyjną funkcję zrob lub zrob2, i w końcu makro **wypisz** lub **wypisz2**, które przepisuje rozwiązanie z tablicy roboczej szach do odpowiedniego excelowego arkusza.

Mamy nadzieję, że ten przewodnik ułatwi Wam lepsze zrozumienie kodu wykorzystanego w naszej aplikacji!