

Maciej Gonet

# ZROZUMIEĆ EXCELA

## FUNKCJE I WYRAŻENIA



Wszelkie prawa zastrzeżone. Nieautoryzowane rozpowszechnianie całości lub fragmentu niniejszej publikacji w jakiegokolwiek postaci jest zabronione. Wykonywanie kopii metodą kserograficzną, fotograficzną, a także kopiowanie książki na nośniku filmowym, magnetycznym lub innym powoduje naruszenie praw autorskich niniejszej publikacji.

Wszystkie znaki występujące w tekście są zastrzeżonymi znakami firmowymi bądź towarowymi ich właścicieli.

Autor oraz Wydawnictwo HELION dołożyli wszelkich starań, by zawarte w tej książce informacje były kompletne i rzetelne. Nie biorą jednak żadnej odpowiedzialności ani za ich wykorzystanie, ani za związane z tym ewentualne naruszenie praw patentowych lub autorskich. Autor oraz Wydawnictwo HELION nie ponoszą również żadnej odpowiedzialności za ewentualne szkody wynikłe z wykorzystania informacji zawartych w książce.

Redaktor prowadzący: Małgorzata Kulik

Projekt okładki: Studio Gravite / Olsztyn  
Obarek, Pokoński, Pazdrijowski, Zaprucki

Grafika na okładce została wykorzystana za zgodą Shutterstock.com

Wydawnictwo HELION  
ul. Kościuszki 1c, 44-100 GLIWICE  
tel. 32 231 22 19, 32 230 98 63  
e-mail: [helion@helion.pl](mailto:helion@helion.pl)  
WWW: <http://helion.pl> (księgarnia internetowa, katalog książek)

Drogi Czytelniku!

Jeżeli chcesz ocenić tę książkę, zajrzyj pod adres

<http://helion.pl/user/opinie/zrefw>

Możesz tam wpisać swoje uwagi, spostrzeżenia, recenzję.

ISBN: 978-83-283-5692-4

Copyright © Helion 2019

Printed in Poland.

- [Kup książkę](#)
- [Poleć książkę](#)
- [Oceń książkę](#)

- [Księgarnia internetowa](#)
- [Lubię to! » Nasza społeczność](#)

# Spis treści

<b>Wstęp .....</b>	<b>15</b>
<b>Rozdział 1. Wprowadzenie .....</b>	<b>23</b>
Czym jest arkusz kalkulacyjny Excel? .....	23
Rozwój arkusza kalkulacyjnego .....	25
<i>Excel w wersji Online</i> .....	27
<i>Kompatybilność z programem Lotus</i> .....	29
<b>Rozdział 2. Podstawy działania arkusza kalkulacyjnego .....</b>	<b>31</b>
Struktura arkusza kalkulacyjnego .....	31
Rodzaje informacji przechowywanych w arkuszu .....	31
<i>Typy danych</i> .....	32
<i>Wartości logiczne i wartości błędów</i> .....	33
<i>Format wprowadzanych danych liczbowych</i> .....	33
Konstrukcja formuł (wyrażeń) w Excelu .....	39
<i>Stałe (literaty)</i> .....	39
<i>Adresy: komórek lub obszarów</i> .....	40
<i>Nazwy: stałych, komórek, obszarów (zakresów) lub formuł</i> .....	51
<i>Operatory działań</i> .....	64
<i>Wywołania funkcji</i> .....	71
<i>Nawiasy okrągłe</i> .....	72
<i>Użycie spacji w celu poprawy czytelności wyrażeń</i> .....	73
<b>Rozdział 3. Kilka użytecznych funkcji .....</b>	<b>75</b>
Popularne funkcje matematyczne .....	75
<i>Odmienność konwencji przy zapisie formuł w matematyce i w Excelu</i> .....	75
<i>Funkcja MODUŁ.LICZBY</i> .....	78
<i>Funkcje PIERWIASTEK i POTĘGA</i> .....	78
<i>Funkcje do obliczania logarytmów</i> .....	78
<i>Funkcje do zaokrąglania liczb</i> .....	78

<i>Funkcja MOD — zasady użycia i ograniczenia</i> .....	79
<i>Wyspecjalizowane funkcje do sumowania</i> .....	81
<i>Funkcja SUMA.ILOCZYNÓW</i> .....	81
Funkcja JEŻELI .....	82
<i>Funkcja JEŻELI zwracająca wartość pustą</i> .....	83
<i>Pierwszeństwo odwołania przed wartością</i> .....	83
<i>Efektywność funkcji JEŻELI</i> .....	84
Funkcja WYBIERZ .....	84
Nowe funkcje logiczne (Excel 2016) .....	85
<i>Funkcja WARUNKI</i> .....	85
<i>Funkcja PRZEŁĄCZ</i> .....	86
Funkcja JEŻELI.BŁĄD .....	87
Funkcje ułatwiające odczytywanie i wyliczanie adresów .....	88
<i>Numery wiersza i kolumny w arkuszu</i> .....	88
<i>Funkcje ILE, WIERSZY i LICZBA.KOLUMN</i> .....	90
<i>Funkcja ADRES</i> .....	90
<i>Funkcja ADR.POŚR</i> .....	92
Funkcja PODAJ.POZYCJĘ .....	97
Funkcje specjalne L (N) i T .....	99
Funkcja KOMÓRKA .....	100
<i>Odczyt nazwy arkusza lub nazwy pliku</i> .....	102
<i>Śledzenie komórki</i> .....	103
Funkcja INFO .....	103
<b>Rozdział 4. Użycie tablic i tabel</b> .....	<b>105</b>
Podstawowe pojęcia dotyczące tablic i macierzy .....	105
Reprezentacja tablic w Excelu .....	106
Wprowadzanie formuł tablicowych .....	108
Operacje tablicowe zaimplementowane w Excelu .....	110
<i>Działania arytmetyczne z użyciem tablic</i> .....	110
<i>Zasada superpozycji</i> .....	112
<i>Łączenie tablic i tworzenie tablic mieszanych</i> .....	114
Funkcje do wykonywania działań na macierzach (tablicach) .....	116
<i>Mnożenie macierzowe</i> .....	116
<i>Transpozycja — funkcja i operacja</i> .....	117
<i>Macierz odwrotna i wyznacznik macierzy</i> .....	121
<i>Macierz jednostkowa</i> .....	121
<i>Macierze trójkątne</i> .....	122
Użycie funkcji JEŻELI i WYBIERZ z argumentami grupowymi .....	123
<i>Użycie funkcji JEŻELI z argumentami grupowymi w wyrażeniu tablicowym</i> .....	123
<i>Użycie funkcji JEŻELI z argumentami grupowymi w wyrażeniu zwykłym</i> .....	125

<i>Funkcja WYBIERZ z argumentami w postaci tablic</i> .....	126
<i>Dodatkowe informacje o funkcji WYBIERZ</i> .....	128
<i>Interpretacja argumentów grupowych przez funkcje JEŻELI i WYBIERZ</i> .....	130
Używanie nazw tablic i formuł tablicowych .....	130
Odwołania strukturalne w tabelach .....	131
<i>Reguły składni odwołań strukturalnych</i> .....	133
<i>Elementy składowe tabeli</i> .....	134
<i>Predefiniowane specyfikatory wierszy</i> .....	134
<i>Operatory odwołania</i> .....	134
<i>Odwołania strukturalne kwalifikowane i niekwalifikowane</i> .....	135
<i>Przykłady użycia odwołań strukturalnych</i> .....	135
<i>Praca z odwołaniami strukturalnymi</i> .....	136
Użycie i interpretacja odwołań wielopoziomowych .....	140
<i>Ogólna charakterystyka odwołań 3-W</i> .....	140
<i>Funkcje, które akceptują odwołania 3-W</i> .....	141
<i>Tworzenie odwołań wielopoziomowych w arkuszu</i> .....	142
<i>Nadawanie nazwy odwołaniu wielopoziomowemu</i> .....	142
<i>Wpływ zmian wprowadzonych w arkuszu na odwołania 3-W</i> .....	143
<i>Przykłady wykorzystania odwołań 3-W</i> .....	143
<b>Rozdział 5. Modyfikacja zawartości arkusza</b> .....	<b>147</b>
Wyświetlanie danych w arkuszu .....	147
<i>Wyświetlanie tekstów formuł zamiast wyników</i> .....	148
Sposoby zaznaczania komórek .....	149
<i>Zaznaczanie niewielkich zakresów komórek</i> .....	149
<i>Zaznaczanie nieprzylegających komórek</i> .....	149
<i>Zaznaczanie całego arkusza, całych wierszy lub kolumn</i> .....	150
<i>Zaznaczanie pierwszych lub ostatnich komórek arkusza, wierszy lub kolumn</i> .....	151
<i>Zaznaczanie zakresów komórek o znanych adresach</i> .....	151
<i>Zaznaczanie dużych zakresów komórek</i> .....	151
Edycja .....	153
<i>Użycie klawisza F9 w czasie edycji</i> .....	153
<i>Znajdowanie i zamiana</i> .....	156
Kopiowanie formuł .....	157
<i>Wklejanie specjalne</i> .....	159
<i>Modyfikacja adresów w formułach przy kopiowaniu</i> .....	160
<i>Tworzenie serii odwołań bezwzględnych</i> .....	161
<i>Kopiowanie odwołań do pełnych wierszy i kolumn</i> .....	162
<i>Kopiowanie formuł zawierających nazwy</i> .....	162
<i>Kopiowanie formuł z odwołaniami względnymi bez modyfikacji odwołań</i> .....	163

Komórki scalone .....	164
<i>Scalanie komórek w sposób standardowy</i> .....	165
<i>Scalanie komórek w sposób niestandardowy</i> .....	167
<i>Kopiowanie formuł w komórkach scalonych</i> .....	167
<i>Formuły tablicowe w komórkach scalonych</i> .....	169
Kopiowanie i łączenie formuł i stałych w komórkach .....	169
Kopiowanie tylko widocznych komórek .....	170
Kopiowanie za pomocą Schowka pakietu Office .....	171
Przenoszenie komórek zawierających formuły .....	173
<i>Przenoszenie powiązanych skoroszytów</i> .....	174
Wstawianie i usuwanie komórek, wierszy lub kolumn .....	174
Edycja formuł tablicowych .....	176
<i>Zmiana zakresu formuł tablicowych</i> .....	176
<i>Zamiana wyników formuł tablicowych na stałe</i> .....	177
<i>Przekształcanie formuł tablicowych w formuły zwykłe</i> .....	178
Kopiowanie formuł tablicowych .....	178
Przeliczanie arkusza i skoroszytu .....	179
Funkcje i działania ulotne .....	181
Zmiana czcionki i początkowego wyglądu arkusza .....	182
<b>Rozdział 6. Odwołania do elementów lub fragmentów tablic .....</b>	<b>185</b>
Użycie funkcji INDEKS .....	185
<i>Argument główny powinien być tablicą lub odwołaniem</i> .....	187
<i>Użycie zera lub pominięcie wartości indeksu</i> .....	187
<i>Użycie funkcji INDEKS do obszaru dwuwymiarowego</i> .....	187
<i>Odwołanie do komórki o wyliczonym adresie</i> .....	188
<i>Łączenie fragmentów tablic</i> .....	189
<i>Zastosowania odwołaniowej formy funkcji INDEKS</i> .....	189
<i>Odwracanie porządku elementów w tablicy</i> .....	190
<i>Uproszczenie zapisu przy wielokrotnym zagnieżdżeniu funkcji JEŻELI</i> .....	190
<i>Listowanie zawartości tablicy jednowymiarowej wierszami lub kolumnami</i> .....	191
Użycie funkcji PRZESUNIĘCIE .....	191
<i>Proste przykłady użycia funkcji PRZESUNIĘCIE</i> .....	194
<i>Odwracanie porządku elementów w zakresie</i> .....	195
Bez użycia funkcji .....	196
<i>Początkowe elementy tablicy</i> .....	196
<i>Część wspólna zakresów</i> .....	196
Dynamiczne generowanie tablic i zakresów o zadanych rozmiarach .....	197
<i>Definicje tablic bazowych</i> .....	197
<i>Wykorzystanie tablic bazowych do definicji innych tablic</i> .....	199
<i>Plan eksperymentu czynnikowego</i> .....	200

<i>Selektywne sumowanie elementów tablic dwuwymiarowych</i> .....	201
<i>Łączenie tablic z wykorzystaniem funkcji MACIERZ.ILOCZYN</i> .....	203
<i>Zakresy dynamiczne</i> .....	204
<i>Formuły tablicowe uwzględniające faktyczne rozmiary obszaru formuły</i> .....	206
<b>Rozdział 7. Klasyfikacja funkcji i interpretacja wyrażeń</b> .....	<b>209</b>
Struktura wyrażeń w Excelu .....	209
<i>Wyrażenia jedno- i wielowartościowe</i> .....	209
<i>Struktura tablicy lub jej brak</i> .....	210
<i>Potrzeba zatwierdzenia tablicowego lub jej brak</i> .....	211
Podział funkcji na grupy .....	213
<i>Rodzaje argumentów funkcji</i> .....	214
<i>Ogólne zasady interpretacji argumentów funkcji</i> .....	215
<i>Przegląd ważniejszych funkcji z podziałem na grupy</i> .....	215
<i>Podział funkcji ze względu na liczbę wartości wynikowych</i> .....	223
<i>Funkcje, które mogą zwracać w wyniku odwołania</i> .....	224
Funkcje bliźniacze odmiennie interpretujące argumenty różnych typów .....	225
Pomijanie argumentów, argumenty opcjonalne, argumenty puste .....	227
<i>Puste argumenty a puste komórki</i> .....	228
<i>Symulowane pominięcie argumentu opcjonalnego</i> .....	229
Konwersja typów argumentów w operacjach arytmetycznych i tekstowych .....	231
<i>Połączone użycie operatorów arytmetycznych i tekstowych</i> .....	231
<i>Użycie operatorów arytmetycznych do danych logicznych</i> .....	231
<i>Użycie plusa jednoargumentowego</i> .....	233
<i>Funkcje ułatwiające konwersję typów danych</i> .....	234
Skutki użycia argumentów innego typu niż przewidziane w opisie funkcji .....	236
<i>Konwersja na tekst</i> .....	236
<i>Konwersja na wartość logiczną</i> .....	236
<i>Konwersja na wartość liczbową</i> .....	237
<i>Użycie liczb z częścią ułamkową zamiast liczb całkowitych</i> .....	239
Użycie odwołań do zamkniętych skoroszytów jako argumentów funkcji .....	240
Powiązania nazw z adresami różnego typu .....	242
<i>Nazwy skojarzone z adresami absolutnymi i względnymi</i> .....	242
<i>Użycie samego wykrzyknika jako kwalifikatora w definicjach nazw</i> .....	245
Zasady interpretacji nazw i adresów w zwykłych wyrażeniach .....	247
<i>Dane grupowe w zwykłych wyrażeniach</i> .....	247
<i>Dane grupowe jako argumenty funkcji</i> .....	248
<i>Nazwy i adresy w zwykłych wyrażeniach — podsumowanie</i> .....	250
Zasady interpretacji nazw i adresów w wyrażeniach tablicowych .....	251
<i>Zestawienie sposobów interpretacji danych grupowych przez funkcje różnych rodzajów</i> ...	254

<b>Rozdział 8. Kontrola poprawności danych i obliczeń .....</b>	<b>257</b>
Kontrola poprawności wprowadzania danych .....	257
<i>Konfiguracja listy rozwijanej .....</i>	<i>261</i>
<i>Lista rozwijana jako odpowiedź .....</i>	<i>263</i>
<i>Lista rozwijana tworzona ad hoc .....</i>	<i>263</i>
<i>Zakreślanie niepoprawnych danych .....</i>	<i>264</i>
Tworzenie dynamicznych list rozwijanych w arkuszu .....	265
<i>Zestawienie pojedynczego wyboru (permutacja) .....</i>	<i>266</i>
<i>Wybór wielokrotny .....</i>	<i>269</i>
<i>Lista wyboru z aktywną odpowiedzią .....</i>	<i>271</i>
Tworzenie list zależnych .....	272
<i>Wynik wyboru w innej komórce niż lista rozwijana .....</i>	<i>274</i>
<i>Lista z wykorzystaniem „Aparatu fotograficznego” .....</i>	<i>275</i>
<i>Lista z wykorzystaniem formatowania warunkowego .....</i>	<i>275</i>
<i>Listy z obsługą zdarzenia Change .....</i>	<i>276</i>
<i>Wykorzystanie formantów pola kombi .....</i>	<i>276</i>
Sygnalizacja i kontrola błędów w obliczeniach .....	278
<i>Kody błędów i ich propagacja .....</i>	<i>278</i>
<i>Funkcje do diagnozowania błędów obliczeń .....</i>	<i>278</i>
<i>Maskowanie wartości błędów .....</i>	<i>281</i>
<i>Ukrywanie wartości zerowych .....</i>	<i>285</i>
Puste teksty i puste komórki .....	287
<i>Identyfikacja .....</i>	<i>287</i>
<i>Puste komórki jako dane .....</i>	<i>288</i>
<i>Usuwanie pustych tekstów .....</i>	<i>289</i>
Wykorzystanie kolumn pomocniczych .....	290
<i>Eliminacja błędnych lub brakujących danych .....</i>	<i>290</i>
<i>Przyspieszenie obliczeń .....</i>	<i>292</i>
Narzędzia wspomagające analizę zawartości arkusza i poprawności formuł .....	292
<i>Poprzedniki i komórki zależne .....</i>	<i>293</i>
<i>Identyfikacja przyczyn błędów .....</i>	<i>294</i>
<i>Czujki .....</i>	<i>295</i>
<i>Szacowanie formuł .....</i>	<i>296</i>
<b>Rozdział 9. Tablicowanie funkcji i wyszukiwanie w tablicach .....</b>	<b>299</b>
Tablicowanie funkcji jednej zmiennej .....	299
<i>Funkcja dana w postaci parametrycznej .....</i>	<i>304</i>
Tablicowanie funkcji dwu zmiennych .....	304
Specyfika użycia polecenia Tabela danych .....	308



Przeszukiwanie tablic funkcji jednej zmiennej .....	310
<i>Funkcje WYSZUKAJ.PIONOWO i WYSZUKAJ.POZIOMO</i> .....	311
<i>Funkcja WYSZUKAJ</i> .....	315
<i>Połączenie funkcji PODAJ.POZYCJĘ i INDEKS</i> .....	318
<i>Wyszukiwanie pierwszej lub ostatniej liczby lub tekstu</i> .....	319
<i>Dane liczbowe uporządkowane rosnąco — zaokrąglenie w górę</i> .....	321
<i>Wyszukiwanie wartości najbliższej w stosunku do wzorca</i> .....	322
<i>Wyszukiwanie odpowiedników wartości zduplikowanych</i> .....	322
<i>Funkcja CZĘSTOŚĆ</i> .....	325
<i>Odczyt gęstości roztworu o podanym stężeniu</i> .....	330
Wyszukiwanie znaczących danych w tablicach .....	331
<i>Eliminacja niepoprawnych danych</i> .....	331
<i>Listowanie danych rozproszonych</i> .....	332
Wyszukiwanie danych w układzie dwuwymiarowym .....	333
<i>Wyszukiwanie danych w zakresie</i> .....	334
<i>Wyszukiwanie danych w tablicy</i> .....	336
<b>Rozdział 10. Kody formatu komórek w Excelu .....</b>	<b>339</b>
Bezpośrednie formatowanie komórek w arkuszu .....	340
<i>Typowe rodzaje formatowania zawartości komórek</i> .....	340
<i>Struktura kodów formatu</i> .....	341
<i>Wzorce formatu sekcji</i> .....	344
<i>Wykorzystanie kodów formatu przy konstrukcji list rozwijanych</i> .....	349
<i>Wykorzystanie formatu ułamkowego i jego alternatywa</i> .....	350
<i>Formatowanie daty i czasu</i> .....	352
<i>Odczyt kodu formatu</i> .....	356
<i>Wykorzystanie kodów formatu w funkcjach trygonometrycznych i cyklometrycznych</i> .....	357
Uproszczony system informowania o formacie komórki .....	359
<b>Rozdział 11. Formatowanie warunkowe .....</b>	<b>361</b>
Opcje formatowania warunkowego dostępne we wszystkich wersjach Excela .....	362
Opcje formatowania warunkowego dostępne od wersji 2007 Excela .....	364
<i>Formatowanie komórek zakresu przy użyciu skali barw</i> .....	365
<i>Formatowanie komórek zakresu przy użyciu pasków danych</i> .....	367
<i>Formatowanie komórek zakresu przy użyciu zestawu ikon</i> .....	369
<i>Formatowanie komórek zawierających tekst, liczbę albo wartości daty lub godziny</i> .....	372
<i>Formatowanie wartości sklasyfikowanych jako pierwsze i ostatnie</i> .....	375
<i>Formatowanie wartości powyżej lub poniżej średniej</i> .....	376
<i>Formatowanie wartości unikatowych lub zduplikowanych</i> .....	378
<i>Używanie formuły do określenia komórek, które należy sformatować</i> .....	380

<i>Zasady ustalania zakresu formatowania i interpretacji formuł w formatowaniu warunkowym</i> .....	381
<i>Usuwanie formatów warunkowych</i> .....	384
<i>Zamiana formatowania warunkowego na stałe</i> .....	384
<i>Zarządzanie pierwszeństwem reguł formatowania warunkowego</i> .....	385
<i>Znajdowanie komórek mających formaty warunkowe</i> .....	386
<b>Rozdział 12. Proste bazy danych</b> .....	<b>389</b>
Struktura tabeli bazy danych .....	389
Sortowanie i filtrowanie .....	390
<i>Kierunek i kolejność sortowania</i> .....	390
<i>Sortowanie czy filtrowanie?</i> .....	392
<i>Sortowanie zakresów zawierających formuły</i> .....	396
<i>Odwołania do komórek z sortowanych obszarów</i> .....	397
<i>Użycie autofiltra</i> .....	399
Filtr zaawansowany .....	401
<i>Zakres kryteriów</i> .....	402
<i>Użycie filtra zaawansowanego</i> .....	406
Użycie formuł zamiast filtra .....	407
Użycie funkcji do odzyskiwania danych z tabeli lub listy .....	409
Operacja konsolidacji .....	414
<i>Konsolidacja według położenia</i> .....	414
<i>Konsolidacja według kategorii</i> .....	417
<b>Rozdział 13. Funkcje do obsługi baz danych</b> .....	<b>419</b>
Funkcje podsumowań .....	419
<i>Funkcja SUMY.CZĘŚCIOWE</i> .....	419
<i>Polecenie Suma częściowa</i> .....	422
<i>Funkcja AGREGUJ</i> .....	422
Funkcje zliczania i sumowania warunkowego .....	426
<i>Składnia funkcji</i> .....	426
<i>Postać kryteriów</i> .....	428
<i>Odpowiedniość między zakresem kontrolnym a zakresem sumowania</i> .....	432
<i>Uwzględnianie kilku kryteriów</i> .....	432
<i>Alternatywy funkcji LICZ.JEŻELI</i> .....	433
<i>Formuły równoważne funkcjom LICZ.WARUNKI i SUMA.WARUNKÓW</i> .....	435
<i>Formuły analogiczne do funkcji ŚREDNIA.JEŻELI i ŚREDNIA.WARUNKÓW</i> .....	436
<i>Obsługa długich ciągów cyfr</i> .....	437
<i>Wyrażenia warunkowe tworzone z użyciem funkcji LICZ.JEŻELI</i> .....	437
<i>Odwrotne wykorzystanie funkcji LICZ.JEŻELI</i> .....	438
<i>Inne przykłady użycia funkcji zliczania i sumowania warunkowego</i> .....	438

Funkcje maksimum i minimum warunkowego (Excel 2016) .....	440
Zliczanie pustych i niepustych komórek .....	441
Grupa funkcji baz danych .....	444
<i>Połączenie funkcji baz danych z tabelą danych</i> .....	447
<b>Rozdział 14. Typowe problemy analizy danych .....</b>	<b>449</b>
Numerowanie wierszy i kolumn z danymi .....	449
<i>Licznik statyczny</i> .....	449
<i>Licznik oparty na odwołaniu względnym do poprzedniej komórki</i> .....	449
<i>Licznik z wykorzystaniem funkcji WIERSZ lub NR.KOLUMNY</i> .....	450
<i>Licznik oparty na funkcji MAX</i> .....	451
<i>Licznik generujący określoną liczbę numerów</i> .....	452
<i>Licznik działający z filtrem lub ukrytymi wierszami</i> .....	453
Ranking danych bez sortowania .....	453
<i>Funkcje MIN.K i MAX.K</i> .....	454
<i>Wyznaczanie wskaźników położenia</i> .....	454
<i>Ustalanie rankingu za pomocą formuł</i> .....	456
<i>Uwzględnienie dodatkowych kryteriów sortowania</i> .....	458
Zliczanie wartości unikatowych .....	459
<i>Dane zawarte w zakresie</i> .....	459
<i>Dane zawarte w tablicy</i> .....	461
<i>Wykorzystanie funkcji CZĘSTOŚĆ, dane w zakresie lub tablicy</i> .....	462
Numerowanie wartości unikatowych .....	463
Tworzenie listy unikatów .....	464
<i>Formuły z funkcją LICZ.JEŻELI</i> .....	464
<i>Formuły z funkcją WYSZUKAJ</i> .....	467
<i>Formuły z funkcją CZĘSTOŚĆ</i> .....	468
Porównywanie zbiorów danych .....	469
<i>Wyszukiwanie wartości wspólnych (iloczyn zbiorów)</i> .....	469
<i>Wyszukiwanie wartości, które występują tylko w jednym zbiorze (różnica zbiorów)</i> .....	470
<i>Wyszukiwanie wartości, które występują w obu zbiorach (suma zbiorów)</i> .....	471
Transpozycja danych zduplikowanych .....	472
<i>Z układu pionowego na poziomy</i> .....	472
<i>Z układu poziomego na pionowy</i> .....	473
<b>Rozdział 15. Obsługa baz danych za pomocą programu Microsoft Query .....</b>	<b>475</b>
Nawiązywanie połączenia ze źródłem danych .....	476
Definiowanie kwerendy za pomocą Kreatora kwerend .....	477
Definiowanie kwerendy bezpośrednio w programie Microsoft Query .....	477
Ponowne używanie i udostępnianie kwerend .....	479
Praca z danymi w programie Excel .....	479

Formatowanie pobranych danych .....	481
Odświeżanie danych zewnętrznych .....	482
Określanie rekordów do pobrania przy użyciu kryteriów .....	482
<i>Co to są kryteria?</i> .....	482
<i>Spełnianie pojedynczego warunku</i> .....	483
<i>Spełnianie wielu warunków</i> .....	483
<i>Wyszukiwanie podobnych rekordów</i> .....	484
<i>Określanie zakresu wartości</i> .....	484
<i>Przykłady wyrażen z operatorami porównania</i> .....	485
<i>Korzystanie z obliczania wartości kryteriów</i> .....	485
<i>Przykłady wyrażen wykorzystujących operatory arytmetyczne</i> .....	485
<i>Tworzenie pola obliczeniowego za pomocą wyrażenia</i> .....	486
<i>Tworzenie pola obliczeniowego za pomocą funkcji</i> .....	487
<i>Funkcje dostępne w polu obliczeniowym</i> .....	487
<i>Przykłady wyrażen z operatorami logicznymi</i> .....	488
<i>Przykłady wyrażen z operatorami innego typu</i> .....	488
Tworzenie kwerendy parametrycznej .....	489
Pobieranie rekordów z wielu tabel .....	490
<i>Sprzężanie tabel</i> .....	490
<i>Sprzężenia wewnętrzne</i> .....	491
<i>Sprzężenia zewnętrzne</i> .....	491
<i>Zmienianie typu sprzężenia między tabelami w kwerendzie</i> .....	491
<i>Samosprzężenia</i> .....	491
<i>Pełne sprzężenia zewnętrzne</i> .....	492
<i>Sprzężenia różnicowe</i> .....	493
Pobieranie tabel z różnych skoroszytów do jednej kwerendy .....	494
<b>Rozdział 16. Hiperłącza .....</b>	<b>497</b>
<i>Co to są hiperłącza?</i> .....	497
<i>Tworzenie i usuwanie hiperłączy w sposób bezpośredni</i> .....	498
<i>Tworzenie hiperłącza</i> .....	498
<i>Ustawianie adresu podstawowego hiperłączy w skoroszycie</i> .....	499
<i>Usuwanie hiperłącza</i> .....	500
<i>Zaznaczanie komórki z hiperłączem</i> .....	500
<i>Automatyczne tworzenie hiperłączy</i> .....	500
<i>Wyłączenie opcji automatycznego formatowania na stałe</i> .....	501
<i>Jednorazowe wyłączenie opcji automatycznego formatowania</i> .....	501
<i>Powrót z hiperłącza</i> .....	501
<i>Dodatkowy tekst w komórce z hiperłączem</i> .....	501

---

Tworzenie hiperłączy za pomocą funkcji HIPERŁĄCZE .....	502
<i>Składnia i ogólne zasady użycia funkcji .....</i>	<i>502</i>
<i>Skok do skoroszytu i wyświetlenie określonego tekstu skoku .....</i>	<i>503</i>
<i>Skok do określonej komórki w arkuszu .....</i>	<i>503</i>
<i>Skok do konkretnego zakresu arkusza i wyświetlenie określonego tekstu skoku .....</i>	<i>504</i>
<i>Skok do określonej lokalizacji w dokumencie programu Word .....</i>	<i>504</i>
<i>Skok do skoroszytu na serwerze sieciowym .....</i>	<i>504</i>
<i>Skok do skoroszytu na innym dysku .....</i>	<i>504</i>
<i>Skok do określonego obszaru w skoroszycie zewnętrznym .....</i>	<i>504</i>
<i>Skok do innej komórki w tym samym skoroszycie .....</i>	<i>505</i>
<i>Użycie alternatywnego stylu adresowania do określenia adresu</i>	
<i>miejsca docelowego łącza .....</i>	<i>506</i>
<i>Użycie odwołania jako źródła adresu miejsca docelowego łącza .....</i>	<i>506</i>
<i>Użycie funkcji HIPERŁĄCZE z argumentem tablicowym .....</i>	<i>507</i>
<i>Hiperłącza warunkowe .....</i>	<i>507</i>
<i>Puste hiperłącza .....</i>	<i>507</i>
<i>Łącze do programu pocztowego .....</i>	<i>508</i>
<i>Wykorzystanie funkcji HIPERŁĄCZE w sposób niestandardowy .....</i>	<i>508</i>
<b>Rozdział 17. Wybrane funkcje wbudowane w Excelu .....</b>	<b>509</b>
<b>Rozdział 18. Słowniczek ważniejszych pojęć .....</b>	<b>553</b>
<b>Skorowidz .....</b>	<b>585</b>



## Rozdział 6.

# Odwołania do elementów lub fragmentów tablic

## Użycie funkcji INDEKS

W przypadku, gdy nie potrzeba ujawniać całej tablicy wyników funkcji czy operacji tablicowej, a jedynie wybrane elementy wynikowej tablicy, należy użyć funkcji `INDEKS`. Umożliwia ona wybór pojedynczego elementu lub wskazanych elementów tablicy danych lub wyników. Jeżeli funkcja ma zwrócić pojedynczą wartość, zatwierdza się ją samym klawiszem `Enter`, jeżeli zaś wynikiem ma być tablica lub obszar, zaznacza się najpierw miejsce na wynik, a następnie zatwierdza przez kombinację `Ctrl+Shift+Enter`.

Funkcja `INDEKS` ma formalnie (według dokumentacji) dwa warianty składni: tablicową i adresową. W zależności od tego, co jest pierwszym argumentem, może mieć zastosowanie forma tablicowa:

```
INDEKS(tablica; nr_wiersza; nr_kolumny)
```

lub forma adresowa, zwana też odwołaniową, która różni się tylko obecnością dodatkowego opcjonalnego argumentu:

```
INDEKS(odwołanie; nr_wiersza; nr_kolumny; nr_obszaru)
```

Pierwszy argument funkcji `INDEKS` będzie nazywał argumentem **głównym**, pozostałe — argumentami **indeksowymi**.

**Tablica** jest tablicą albo wyrażeniem dającym w wyniku tablicę, a **odwołanie** jest zakresem komórek, który może obejmować więcej niż jeden obszar. Jeśli **odwołanie** obejmuje adresy kilku obszarów, ich listę należy ująć w nawiasy. W tym miejscu może wystąpić również wyrażenie, dające w wyniku odwołanie.

*Nr\_wiersza* wybiera wiersz tablicy lub obszaru, z którego ma pochodzić wartość wynikowa. Jeśli pominiemy ten argument lub zostanie użyta wartość 0, to w zależności od sposobu użycia funkcji wybrane zostaną albo wszystkie wiersze, albo wiersz bieżący, albo pierwszy.

Argument *nr\_kolumny* określa kolumnę tablicy lub obszaru, z której ma pochodzić wartość wynikowa. Jeśli pominiemy go lub zostanie użyta wartość 0, to w zależności od sposobu użycia funkcji wybrane zostaną albo wszystkie kolumny, albo kolumna bieżąca, albo pierwsza.

*Nr\_wiersza* i *nr\_kolumny* muszą odnosić się do elementu lub komórki **wewnątrz** tablicy lub obszaru, w innym przypadku funkcja INDEKS podaje w wyniku wartość błędu #ADR!, ale tylko w tych wierszach lub kolumnach, które zostały błędnie zaadresowane. Wiersze i kolumny argumentu głównego są numerowane od 1. Jeżeli **odwołanie** obejmuje kilka obszarów, numeracja każdego z nich jest niezależna.

Jeśli zostanie użyty zarówno argument *nr\_wiersza*, jak i *nr\_kolumny*, to funkcja INDEKS daje w wyniku odwołanie do komórki na przecięciu wiersza i kolumny o numerach podanych w *nr\_wiersza* i *nr\_kolumny* lub odpowiednią wartość z tablicy.

Jeśli argument główny zawiera tylko jeden wiersz lub kolumnę, to jeden z argumentów *nr\_wiersza* lub *nr\_kolumny* (wraz z poprzedzającym średnikiem) można pominąć. Można również pozostawić sam średnik lub użyć argumentu o wartości 0 lub 1.

Jeśli argumentem głównym funkcji INDEKS jest **odwołanie** do obszaru dwuwymiarowego lub zakresu nieregularnego, aby wyrażenie było poprawne składniowo, w wywołaniu muszą być co najmniej **dwa średniki**; w innych przypadkach wystarczy jeden.

Argument *nr\_obszaru* określa, z którego obszaru ma zostać zwrócony wynik. Ma on znaczenie, gdy **odwołanie** obejmuje kilka obszarów; w razie pominięcia tego argumentu (wraz z poprzedzającym średnikiem) uwzględniony zostanie pierwszy obszar. Argument ten nie może mieć wartości zerowej, nie można też pozostawić w wywołaniu trzeciego średnika bez wartości. Jeśli argumentem głównym jest **odwołanie** do pojedynczego obszaru lub **tablica**, to argument *nr\_obszaru* (jeśli wystąpi) musi mieć wartość 1. Wynika stąd, że w istocie nie ma dwóch wariantów składni, a tylko jeden — wariant odwołaniowy z ograniczeniami dotyczącymi argumentu *nr\_obszaru*.

Jeśli argument główny jest tablicą, funkcja zwraca w wyniku zawsze pojedynczą wartość lub tablicę, jeśli zaś ten argument jest odwołaniem, funkcja zwraca odwołanie, o ile jest to możliwe, a jeśli nie — zwraca wartość lub tablicę wartości.

Argumenty indeksowe mogą mieć również formę tablic lub zakresów komórek zawierających numery wierszy, kolumn lub obszarów. Ta funkcjonalność nie jest jednak opisana w oficjalnej dokumentacji. Należy jej używać z dużą ostrożnością, szczególnie gdy formuły są złożone, bo wynik nie zawsze da się łatwo przewidzieć [B20]. Właśnie w takich przypadkach funkcja INDEKS pomimo użycia formy adresowej zwraca tablicę, a nie odwołania.

Argumentem głównym funkcji INDEKS nie mogą być tzw. odwołania 3-W (piszę o nich w rozdziale 4., w podrozdziale „Użycie i interpretacja odwołań wielopoziomowych”), gdyż w sensie formalnym nie są to ani odwołania, ani tablice.



## Argument główny powinien być tablicą lub odwołaniem

Jeżeli jako argumentu głównego funkcji INDEKS użyto stałej lub wyrażenia zwracającego pojedynczą wartość, wynik jest poprawny tylko wtedy, gdy jest to liczba. W każdym innym przypadku funkcja zwraca wartość błędu (jeżeli wyrażenie zwraca błąd, otrzymamy wartość tego błędu, jeżeli zaś wyrażenie zwraca wartość nieliczbową, wynikiem będzie wartość błędu #ARG!). Przyczyną takiego zachowania funkcji INDEKS jest inna — w porównaniu z pojedynczą wartością — struktura tablicy. Aby uniknąć tego problemu (kiedy *tablica* może czasem być pojedynczą wartością nieliczbową), należy zastosować konstrukcję:

```
= INDEKS(JEŻELI({1};tablica); nr_wiersza; nr_kolumny)
```

Opisana wyżej sytuacja dotyczy również innych funkcji tablicowych.

## Użycie zera lub pominięcie wartości indeksu

Jeżeli wykorzystany zostanie tylko jeden argument: *nr\_wiersza* lub *nr\_kolumny*, a drugi zostanie pominięty (gdy argumentem jest *odwołanie* do zakresu dwuwymiarowego, średnik musi pozostać) lub jako argument indeksowy zostanie podana wartość 0, to INDEKS da w wyniku odwołanie lub tablicę z całym wierszem lub kolumną. Aby wyświetlić te wartości w arkuszu jako sekwencję, należy wprowadzić INDEKS jako funkcję tablicową, naciskając *Ctrl+Shift+Enter*. Przy zatwierdzeniu samym klawiszem *Enter*, jeśli argumentem głównym jest *odwołanie*, funkcja INDEKS zwraca wartość z tego samego wiersza lub tej samej kolumny na zasadzie rzutu prostokątnego, analogicznie jak w przypadku odwołania do zakresu w zwykłej formule, natomiast jeśli argumentem głównym jest *tablica*, INDEKS zwraca jej pierwszy element (por. rozdział 7., punkt „Dane grupowe w zwykłych wyrażeniach”). Opis ten dotyczy typowego użycia funkcji INDEKS, to znaczy takiego, w którym argumenty indeksowe nie mają struktury tablicy, lecz są zwykłymi liczbami.

## Użycie funkcji INDEKS do obszaru dwuwymiarowego

### Przykład 6.1.

Jako źródło danych wykorzystamy zakres (obszar) *TA* o wymiarach 4×4, umieszczony w arkuszu.

1	2	4	6
3	5	-3	7
6	8	10	-2
5	7	4	3

Jego pierwszą kolumnę oznaczymy *TX*, a pierwszy wiersz — *TY*.

Odwołanie do drugiego elementu (wiersza) tablicy *TX* może mieć jedną z postaci:

```
= INDEKS(TX;2)
= INDEKS(TX;2;)
= INDEKS(TX;2;0)
= INDEKS(TX;2;1)
```

Podobnie, odwołanie do trzeciego elementu (kolumny) tablicy **TY** może mieć postać (alternatywnie):

- = INDEKS(**TY**;3)
- = INDEKS(**TY**; ;3)
- = INDEKS(**TY**;0;3)
- = INDEKS(**TY**;1;3)

Jeśli **TA** oznacza tablicę lub zakres dwuwymiarowy, możliwe są na przykład następujące odwołania tablicowe:

- = INDEKS(**TA**;1;)
- = INDEKS(**TA**;1;0) — w obu przypadkach wynikiem jest cały pierwszy wiersz tablicy **TA**.

Jeśli **TA** jest odwołaniem do zakresu, wyrażenie = INDEKS(**TA**;1) byłoby błędne, ale już wyrażenie = INDEKS(+**TA**;1) jest poprawne. W przypadku, gdy **TA** oznacza stałą tablicową, prawidłowe są oba ostatnie wyrażenia.

- = INDEKS(**TA**; ;2)
- = INDEKS(**TA**;0;2) — w obu przypadkach wynikiem jest cała druga kolumna tablicy **TA**.

Jeśli **TA** jest odwołaniem do zakresu, wyrażenie = INDEKS(**TA**;2) byłoby błędne.

Pojedynczy element tablicy lub obszaru **TA** odczytamy za pomocą wyrażenia:

- = INDEKS(**TA**;2;4)

W tym przypadku będzie to element na przecięciu drugiego wiersza i czwartej kolumny (wartość 7 lub odwołanie do komórki zawierającej tę wartość).

## Odwołanie do komórki o wyliczonym adresie

Pierwszy argument funkcji INDEKS może reprezentować cały arkusz, co można zapisać jako zakres wierszy \$1:\$1048576 (od wersji 2007) bądź \$1:\$65536 (we wcześniejszych wersjach). Jeżeli odwołanie ma dotyczyć arkusza innego niż bieżący, należy je poprzedzić kwalifikatorem arkusza. Można oczywiście użyć również mniejszego zakresu wierszy lub odpowiedniego zakresu kolumn, počawszy od kolumny A. Użycie funkcji INDEKS w takiej formie umożliwia odwołanie do komórki lub komórek o wyliczonym adresie. Jeśli nie pamiętamy numeru ostatniego wiersza lub ostatniej kolumny, to wszystko można wyliczyć:

- numer ostatniego wiersza — = ILE.WIERSZY(A:A)
- numer ostatniej kolumny — = LICZBA.KOLUMN(1:1)
- oznaczenie literowe ostatniej kolumny — = PODSTAW(ADRES(1;LICZBA.KOLUMN(1:1);4);1;"")

Aby odczytać wartość z komórki o określonych absolutnych numerach wiersza i kolumny, można użyć wyrażenia:

- = INDEKS(ADR.POŚR("nazwa\_arkusza!1:"&ILE.WIERSZY(\$A:\$A)); numer\_wiersza; numer\_kolumny)

Ten sam wynik można uzyskać za pomocą funkcji `ADR.POŚR` i `ADRES`:

```
= ADR.POŚR(ADRES( numer_wiersza; numer_kolumny;;; nazwa_arkusza))
```

Jeżeli odwołanie dotyczy bieżącego arkusza, w obu wersjach można pominąć *nazwę\_arkusza*.

## Łączenie fragmentów tablic

### Przykład 6.2.

Załóżmy, że mamy dwie tablice (lub obszary danych): *tab11*, obejmującą trzy kolumny, oraz *tab12*, złożoną z dwóch kolumn. Za pomocą odpowiedniej formuły tablicowej w połączeniu z funkcją `INDEKS` możemy utworzyć tablicę złożoną z pięciu kolumn, tak jakby były zapisane obok siebie.

```
= {1;0;0;0;0}*INDEKS( tab11;;1)+ {0;1;0;0;0}*INDEKS( tab11;;2)+
↳ {0;0;1;0;0}*INDEKS( tab11;;3)+ {0;0;0;1;0}*INDEKS( tab12;;1)+
↳ {0;0;0;0;1}*INDEKS( tab12;;2)
```

Ta formuła jest analogiczna do użytej w przykładzie 4.2. Jej koncepcja polega na podziale oryginalnych tablic na kolumny za pomocą funkcji `INDEKS` i połączeniu tak otrzymanych kolumn za pomocą operacji tablicowych. Ze względu na zasadę superpozycji, która umożliwia dopasowanie wymiarów przez kopiowanie tylko pojedynczych wierszy lub kolumn, przy takiej konstrukcji wyrażenia nie ma możliwości uwzględnienia tablicy dwuwymiarowej w całości. Właściwości tej formuły są podobne do tych z przykładu 4.2.: tablice muszą zawierać tylko dane liczbowe, formuła wymaga zatwierdzenia tablicowego, ale po takim zatwierdzeniu może być użyta do dalszych obliczeń. Jest to wyrażenie z odwołaniem lub tablicą.

Wygodniejsze okazuje się w tym przypadku użycie funkcji `WYBIERZ`:

```
= WYBIERZ({1;2;3;4;5}; INDEKS( tab11;;1); INDEKS( tab11;;2); INDEKS( tab11;;3);
↳ INDEKS( tab12;;1); INDEKS( tab12;;2))
```

albo jeszcze prościej, ale w sposób mniej czywisty:

```
= WYBIERZ({1;1;1;2;3}; tab11; INDEKS( tab12;;1); INDEKS( tab12;;2))
```

Czytelnikowi pozostawiam odpowiedź na pytanie, dlaczego tablicy *tab12* nie można przywołać w jednej pozycji. Formuły z funkcją `WYBIERZ` dają wyniki niemal pełnowartościowe (TPA — wyjaśnienie tego skrótu można znaleźć w rozdziale 4, w punkcie „Interpretacja argumentów grupowych przez funkcje JEŻELI i WYBIERZ”), gotowe do dalszego przetwarzania.

Alternatywne sposoby rozwiązania tego problemu korzystają z funkcji `INDEKS` w sposób niestandardowy [B20]. Jeszcze inną metodę, z wykorzystaniem funkcji `MACIERZ`.`ILOCZYN`, pokazałem nieco dalej, w przykładzie 6.5. Problem można rozwiązać również postępując się Visual Basicem [C6].

## Zastosowania odwołań formy funkcji INDEKS

Odwołaniowa forma funkcji `INDEKS` ma szczególne zastosowanie w przypadku zakresów o nieregularnych kształtach, złożonych z kilku obszarów. Jeżeli zakres złożony stanowią pojedyncze komórki, można pominąć *nr\_wiersza* i *nr\_kolumny* i podać tylko *nr\_obszaru*. W ten sposób można zebrać rozproszone dane w jednym miejscu.

Wykorzystując fakt, że ta forma funkcji INDEKS daje w wyniku adres, można jej użyć do utworzenia zakresu dynamicznego o zmiennym kształcie:

```
adr_pocz:INDEKS(kom_kon;;;nr_obszaru)
```

Zakres **kom\_kon** powinien być zakresem nieregularnym i składać się z komórek wyznaczających prawe dolne narożniki konstruowanych obszarów. Początek zakresu jest w tym przykładzie stały, ale można wyobrazić sobie, że w tym miejscu zamiast adresu podanego wprost użyjemy formuły zwracającej adres. Argument *nr\_obszaru* wyznacza bieżący adres końca obszaru.

Analogiczny efekt można uzyskać za pomocą funkcji WYBIERZ z adresami komórek podanymi jako argumenty. Przykład takiego rozwiązania podałem w rozdziale 4., w podrozdziale „Reprezentacja tablic w Excelu”.

## Odwracanie porządku elementów w tablicy

Jeszcze innym przykładem użycia funkcji INDEKS jest odwracanie porządku elementów w tablicy. Twórcy Excela nie przewidzieli specjalnej funkcji, która by rozwiązywała ten problem, na wzór funkcji TRANSPONUJ. Załóżmy, że mamy zakres wierszowy, któremu nadamy nazwę **zakres**. Chcemy wygenerować zakres o identycznych wymiarach, w którym kolejność elementów będzie odwrócona.

Kolejne komórki zawierające wynik opisane będą formułą zwykłą:

```
= INDEKS(zakres; LICZBA.KOLUMN(zakres)-NR.KOLUMNY(A1)+1)
```

LICZBA.KOLUMN(**zakres**) oznacza liczbę komórek w tym zakresie, NR.KOLUMNY(A1) zaś numery kolejnych kolumn, rozpoczynając od pierwszej. W pierwszej komórce zakresu wynikowego w wyrażeniu NR.KOLUMNY(A1) musimy użyć adresu odnoszącego się do kolumny A; po skopiowaniu w prawo adresy będą odnosić się do kolejnych kolumn, zapewniając otrzymanie oczekiwanego wyniku.

Gdybyśmy chcieli osiągnąć podobny efekt za pomocą formuły tablicowej i uzyskać od razu (jedną formułą) całą tablicę wynikową, musielibyśmy użyć funkcji INDEKS w sposób niestandardowy [B20].

## Uproszczenie zapisu przy wielokrotnym zagnieżdżeniu funkcji JEŻELI

Gdy mamy do sprawdzenia kilka warunków i chcemy użyć wartości skojarzonej z pierwszym, który ma wartość PRAWDA, możemy wykorzystać zagnieżdżoną funkcję JEŻELI:

```
= JEŻELI(war1; wartość1; JEŻELI(war2; wartość2; JEŻELI(war3; wartość3; wartość0)))
```

**wartość0** jest wartością domyślną, której chcemy użyć, gdy żaden z warunków nie został spełniony. W Excelu 2016+ podobne zadanie realizuje funkcja WARUNKI

```
= WARUNKI(war1; wartość1; war2; wartość2; war3; wartość3; PRAWDA; wartość0)
```

Każdy z tych zapisów ma jednak tę wadę, że wymaga modyfikacji przy zmianie liczby warunków, a ponadto przy dużej ich liczbie staje się nieczytelny.

Jeżeli konstrukcja warunków i powiązanych wartości jest podobna i uda się je zapisać w postaci ogólnych formuł tablicowych, powyższe zapisy można uprościć, wykorzystując funkcje INDEKS i PODAJ.POZYCJĘ:

= JEŻELI.BŁĄD(INDEKS(*wartości*; PODAJ.POZYCJĘ(PRAWDA; *warunki*; 0)); *wartość0*)

## Listowanie zawartości tablicy jednowymiarowej wierszami lub kolumnami

Często otrzymujemy jako wynik jakichś obliczeń tablicę (najczęściej jednowymiarową) o nieznaną długość. Aby ją wyświetlić w całości, powinniśmy znać jej długość, w celu zarezerwowania odpowiedniej liczby komórek na wynik. Alternatywą jest wyświetlanie elementów tablicy w kolejnych komórkach na podstawie formuły jednokomórkowej, kopiowanej tak długo, jak potrzeba. Jeśli tablica jest jednowymiarowa, jej orientacja jest bez znaczenia. Możemy wyświetlać wyniki w kolumnie, używając w formule funkcji ILE.WIERSZY, lub w wierszu, za pomocą formuły z funkcją LICZBA.KOLUMN. Odpowiednie polecenia mają postać:

= JEŻELI.BŁĄD(INDEKS(tablica; ILE.WIERSZY(H\$3:H3));"")  
 = JEŻELI.BŁĄD(INDEKS(tablica; LICZBA.KOLUMN(\$H3:H3));"")

H3 jest przykładowym adresem pierwszej komórki, w której chcemy ulokować początek listy. Używamy tu adresu absolutnego lub mieszanego jako początkowego i adresu względnego jako końcowego. Są to formuły zwykłe, gdyby jednak tablica była wyrażeniem złożonym, może być wymagane zatwierdzenie tablicowe.

## Użycie funkcji PRZESUNIĘCIE

Druga funkcja, która umożliwi odwoływanie się do fragmentów obszarów, nosi nazwę PRZESUNIĘCIE. Pozwala ona na przykład „wyciąć” z obszaru dowolny fragment o kształcie prostokąta (w tym pojedynczy wiersz lub kolumnę), a także odczytać zawartość obszaru położonego w określonej odległości od znanej komórki. Ścisłej, funkcja zwraca **odwołanie** do komórki lub obszaru, przesuniętych o podaną liczbę wierszy lub kolumn w stosunku do komórki odniesienia. Funkcja PRZESUNIĘCIE spełnia podobne zadania, co INDEKS, ale można wskazać kilka istotnych różnic:

- funkcja PRZESUNIĘCIE przyjmuje jako argument główny **tylko** odwołania do obszarów, podczas gdy INDEKS może przyjmować zarówno odwołania (i to również do zakresów nieregularnych), jak i tablice;
- funkcja PRZESUNIĘCIE może zwracać wynik również **spoza** argumentu głównego, natomiast INDEKS tylko w jego obrębie;
- INDEKS sprawdza się lepiej, gdy chcemy uzyskać w wyniku pojedyncze wartości lub całe wiersze i całe kolumny, PRZESUNIĘCIE dobrze sobie radzi z uzyskaniem pojedynczych wartości oraz dowolnych spójnych obszarów. Żadna z nich nie radzi sobie natomiast dobrze z wybieraniem różnych fragmentów danych i łączeniem ich w nową całość;

- funkcja INDEKS numeruje wiersze i kolumny od 1, a PRZESUNIĘCIE — od 0. Należy o tym pamiętać przy przekształcaniu wzorów wykorzystujących te funkcje;
- PRZESUNIĘCIE jest funkcją ulotną, co wpływa negatywnie na szybkość obliczeń, podczas gdy INDEKS jest nieulotna.

Oto składnia funkcji PRZESUNIĘCIE:

PRZESUNIĘCIE(*zakres\_odniesienia*; *wiersze*; *kolumny*; *wysokość*; *szerokość*)

**Zakres\_odniesienia** to adres komórki (odwołanie) lub obszaru sąsiednich komórek, od którego wyznacza się przesunięcie. Jeśli nie jest to pojedyncza komórka, przesunięcie liczy się od **komórki odniesienia**, czyli od początku (lewego górnego rogu) tego obszaru. Jeżeli ten argument nie oznacza odwołania, funkcja PRZESUNIĘCIE zwróci wartość błędu #ARG!.

**Wiersze** to liczba wierszy w górę (liczba ujemna) lub w dół (liczba dodatnia) liczonych od początku **zakresu\_odniesienia**, określająca położenie początku zakresu wynikowego. Argument **wiersze** równy 0 oznacza ten sam wiersz. Liczbę 0 można pominąć, ale trzeba pozostawić poprzedzający średnik.

**Kolumny** to liczba kolumn w lewo (liczba ujemna) lub w prawo (liczba dodatnia) liczonych od początku **zakresu\_odniesienia**, określająca położenie początku zakresu wynikowego. Argument **kolumny** równy 0 oznacza tę samą kolumnę. Liczbę 0 można pominąć, ale trzeba pozostawić poprzedzający średnik.

Argumenty **wiersze** i **kolumny** będą określał mianem **argumentów indeksowych**.

Jeśli argumenty indeksowe przesuwają odwołanie poza brzeg arkusza, to funkcja PRZESUNIĘCIE daje w wyniku wartość błędu #ADR! — i to we wszystkich komórkach wyniku, a nie tylko w tych, które powodują błąd (inaczej niż w przypadku funkcji INDEKS). Formalnie jest to jedna wartość błędu, która powiela się przy zatwierdzeniu w kilku komórkach.

Podając wartości opcjonalnych argumentów **wysokość** i **szerokość**, można określić rozmiary zakresu wynikowego, inne niż rozmiary **zakresu\_odniesienia**.

**Wysokość** to liczba wierszy określająca wysokość zakresu wynikowego. Powinna być liczbą dodatnią. Jeśli jest liczbą ujemną, funkcja zwraca zakres położony powyżej nowego **zakresu\_odniesienia**, ale bez zmiany kolejności komórek (jest to właściwość nieudokumentowana).

**Szerokość** to liczba kolumn określająca szerokość zakresu wynikowego. Także ona powinna być liczbą dodatnią. Jeśli jest liczbą ujemną, funkcja zwraca zakres położony na lewo od nowego **zakresu\_odniesienia**, ale bez zmiany kolejności komórek (jest to właściwość nieudokumentowana).

Jeśli argumenty **wysokość** lub **szerokość** zostaną pominięte, to zakres wynikowy będzie mieć taką samą wysokość lub szerokość jak **zakres\_odniesienia**. I to jest jedyna okoliczność, gdy rozmiary **zakresu\_odniesienia** mają znaczenie. Jeżeli **wysokość** i **szerokość** są podane jawnie, znaczenie ma tylko początek **zakresu\_odniesienia** (komórka odniesienia).

Funkcja PRZESUNIĘCIE w rzeczywistości nie przesuwa żadnych komórek ani nie zmienia wyboru zakresu, tylko zwraca **odwołanie** do zakresu wynikowego, który jest przesunięty w stosunku do **zakresu odniesienia**. W związku z tym może być wykorzystywana z innymi funkcjami wymagającymi podania odwołania jako argumentu.

Jeżeli w zakresie działania funkcji PRZESUNIĘCIE wstawiamy lub usuwamy wiersze lub kolumny, wszystkie argumenty będące adresami są stosownie modyfikowane, natomiast **stałe liczbowe nie ulegają zmianie**. Należy o tym pamiętać, gdyż w konsekwencji tego wstawienie lub usunięcie wiersza czy kolumny może zmienić wynik funkcji.

Podsumowując, gdy początek **zakresu odniesienia** (komórkę odniesienia) oznaczymy symbolem **pzo**, zakresy zwracane przez funkcję PRZESUNIĘCIE można obliczyć ze wzorów:

Gdy *wysokość* >0, zakres wierszy obejmuje

od WIERSZ (**pzo**)+*wiersze*

do WIERSZ (**pzo**)+*wiersze*+*wysokość*-1

Gdy *szerokość* >0, zakres kolumn obejmuje

od NR. KOLUMNY (**pzo**)+*kolumny*

do NR. KOLUMNY (**pzo**)+*kolumny*+*szerokość*-1

Gdy *wysokość* <0, zakres wierszy obejmuje

od WIERSZ (**pzo**)+*wiersze*+*wysokość*+1

do WIERSZ (**pzo**)+*wiersze*

Gdy *szerokość* <0, zakres kolumn obejmuje

od NR. KOLUMNY (**pzo**)+*kolumny*+*szerokość*+1

do NR. KOLUMNY (**pzo**)+*kolumny*

Można również zauważyć, że wartości argumentów *wysokość* lub *szerokość* równe -1 mają takie samo znaczenie jak 1, a wartość 0 powoduje błąd #ADR!.

Użycie ujemnych wartości argumentów *wysokość* i *szerokość*, jakkolwiek dopuszczalne, nie poszerza możliwości funkcji, chociaż w pewnych sytuacjach pozwala skrócić zapis wyrażenia. Taki sam wynik uzyskamy zmieniając wartości *wysokości* i *szerokości* na ich wartości bezwzględne i równocześnie korygując odpowiednio argumenty *wiersze* i *kolumny*. Nowe wartości tych argumentów to:

*wiersze*-MODUŁ.LICZBY(*wysokość*)+1 oraz *kolumny*-MODUŁ.LICZBY(*szerokość*)+1

Gdy funkcja PRZESUNIĘCIE jest wywoływana w formule zwykłej, a wynik obejmuje formalnie kilka komórek, efektywny wynik jest wyznaczany według zasady rzutu prostokątnego, ale w odniesieniu do wyniku formuły, a nie do adresu argumentu (por. rozdział 2., podpunkt „Interpretacja adresów obszarów w zwykłych wyrażeniach”). Przykładowo wywołanie

= PRZESUNIĘCIE(\$B\$2:\$B\$4;1;1)

powinno dać w wyniku zakres \$C\$3:\$C\$5. Jeśli to wywołanie umieścimy w dowolnej kolumnie w trzecim wierszu, otrzymamy w wyniku zawartość komórki \$C\$3, a w czwartym — zawartość komórki \$C\$4. W każdym wierszu spoza zakresu 3:5 zwrócona zostanie wartość błędu #ARG!. Jeżeli natomiast podamy *wysokość* równą 1, to w dowolnej komórce powyższe wywołanie będzie odnosić się do komórki \$C\$3.

Należy pamiętać, że funkcja PRZESUNIĘCIE zwraca zakres o wielkości takiej jak **zakres odniesienia**, chyba że podamy inne wartości argumentów *wysokość* lub *szerokość*. Jeżeli wynik działania funkcji obejmuje więcej niż jedną komórkę, należy pamiętać o użyciu klawiszy **Ctrl+Shift+Enter**, nawet jeśli chcemy wykorzystać tylko pierwszą komórkę z wyniku. Ewentualnie można objąć wynik funkcją L (N) lub T.

Gdy argumenty *wiersze* lub *kolumny*, a także *wysokość* lub *szerokość* są tablicami, funkcja PRZESUNIĘCIE zwraca **pseudozakres**, który reprezentuje tablicę zakresów, ale nie może być bezpośrednio użyty w wyrażeniach. Można go użyć jako argumentu funkcji L (N) i T oraz SUMY. ↪ CZĘŚCIOWE i AGREGUJ, funkcji zliczania i sumowania warunkowego, np. SUMA.JEŻELI, a także wielu innych funkcji. Przekształcają one pseudozakres w tablicę, której można użyć w wyrażeniach lub wyświetlić w arkuszu [B21].

Jeżeli chcemy wykorzystać funkcję PRZESUNIĘCIE w arkuszu z dużą ilością danych, należy pamiętać, że jest to funkcja ulotna (por. rozdział 5., podrozdział „Funkcje i działania ulotne”), a więc jest obliczana przy każdym przeliczeniu arkusza. Jeśli zależy nam na szybkości działania arkusza, powinniśmy starać się, aby wywołanie funkcji PRZESUNIĘCIE było jak najprostsze, zaś jej argumenty, jeśli wymagają obliczenia, powinny być obliczane w osobnych komórkach, a tylko wyniki przekazywane do funkcji. Dzięki temu unika się wielokrotnego niepotrzebnego obliczania argumentów.

## Proste przykłady użycia funkcji PRZESUNIĘCIE

### Przykład 6.3.

Wróćmy do przykładu 6.1, w którym wyodrębnialiśmy fragmenty obszaru lub tablicy przy użyciu funkcji INDEKS. Jeśli chcemy w zdefiniowanym wcześniej obszarze **TA** pominąć pierwszy wiersz i pierwszą kolumnę, możemy zastosować funkcję PRZESUNIĘCIE (tablica **TA** musi być zapisana w arkuszu, aby miała adres):

= PRZESUNIĘCIE(**TA**;1;1;3;3)

W wyniku otrzymamy pełnowartościowe odwołanie do obszaru zawierającego takie liczby:

5	-3	7
8	10	-2
7	4	3

### Przykład 6.4.

Funkcję PRZESUNIĘCIE można również wykorzystać do utworzenia odwołania do komórki o wyliczonym adresie. W tym przypadku należy użyć wyrażenia:

= PRZESUNIĘCIE(\$A\$1; numer\_wiersza - 1; numer\_kolumny - 1)

Adres komórki odniesienia można poprzedzić kwalifikatorem arkusza.

### Przykład 6.5.

Przykład 4.3. dotyczył połączenia dwóch obszarów danych: **obsz1**, obejmującego trzy kolumny, oraz **obsz2**, złożonego z dwóch kolumn w jedną ciągłą tablicę pięciokolumnową. Zasygnalizowałem tam możliwość użycia do tego funkcji PRZESUNIĘCIE. Teraz przyszła pora, aby tę zapowiedź zrealizować.



Zadaniem funkcji PRZESUNIĘCIE jest poszerzenie obszaru **obsz2** o trzy kolumny umieszczone z przodu, przed właściwymi danymi. Należy w tym celu wywołać ją w taki sposób:

```
PRZESUNIĘCIE(obsz2;0;-3;;5)
```

Aby wyeliminować błędy, można wykorzystać — jak w przykładzie 4.3. — funkcję JEŻELI.BŁĄD. Doprowadzi to nas do formuły tablicowej:

```
= JEŻELI.BŁĄD({1;1;1;0;0}*obsz1;0)+JEŻELI.BŁĄD({0;0;0;1;1}*PRZESUNIĘCIE(obsz2;0;-3;;5);0)
```

Wzór można zastosować tylko do danych liczbowych (dane nieliczbowe zostaną zastąpione zerami). Funkcja JEŻELI.BŁĄD musi być użyta dwukrotnie, osobno dla każdego obszaru. Formuła po zatwierdzeniu tablicowym może być wykorzystana do dalszych obliczeń.

Do eliminacji błędów można wykorzystać również funkcje JEŻELI lub WYBIERZ. W tym przypadku nie musimy się ograniczać do danych liczbowych; mogą być one dowolnego typu.

```
= JEŻELI({1;1;1;0;0};obsz1;PRZESUNIĘCIE(obsz2;0;-3;;5))
= WYBIERZ({1;1;1;2;2};obsz1;PRZESUNIĘCIE(obsz2;0;-3;;5))
```

Otrzymane formuły są wyrażeniami TPA, więc mogą być wykorzystane do dalszych obliczeń. Warunkiem powodzenia tych metod jest to, by obszar **obsz2** nie był położony zbyt blisko lewej krawędzi arkusza.

Metody tej nie można stosować do tablic, a jedynie do obszarów, które mają swój adres w arkuszu.

## Odwracanie porządku elementów w zakresie

Inny przykład zastosowania funkcji PRZESUNIĘCIE to odwracanie porządku elementów w zakresie. Wcześniej pokazałem, jak można ten problem rozwiązać za pomocą funkcji INDEKS.

Załóżmy, że mamy zakres wierszowy, któremu nadamy nazwę **wiersz**. Chcemy wygenerować zakres o identycznych wymiarach, w którym kolejność elementów będzie odwrócona. Zakres wierszowy można umieścić w dowolnym miejscu arkusza. W pierwszej komórce tego zakresu umieszczamy wywołanie funkcji:

```
= PRZESUNIĘCIE(wiersz;0;LICZBA.KOLUMN(wiersz)-NR.KOLUMNY(A1);1;1)
```

i kopiujemy w całym zakresie wyniku. Jest to zwykła formuła (nie tablicowa).

Analogiczną formułę można zastosować do zakresu kolumnowego o nazwie **kolumna**:

```
= PRZESUNIĘCIE(kolumna;ILE.WIERSZY(kolumna)-WIERSZ(A1);0;1;1)
```

Podobny efekt można osiągnąć za pomocą jednej formuły tablicowej:

```
= T(PRZESUNIĘCIE(kolumna; ILE.WIERSZY(kolumna)- WIERSZ(kolumna)+ KOMÓRKA("wiersz";
↪kolumna)-1;0))
```

Jako funkcji zewnętrznej należy użyć funkcji T, jeśli dane są tekstowe, a L lub N, gdy dane są liczbowe. Formuła w przypadku **wiersza** byłaby analogiczna. Tu wyrażenie z funkcją PRZESUNIĘCIE należy do kategorii tzw. pseudozakresów, stąd potrzeba użycia dodatkowej funkcji, aby udostępnić wartości formuły [B21].

# Bez użycia funkcji

## Początkowe elementy tablicy

Aby odwołać się do początkowych elementów tablicy (początkowych wierszy lub kolumn), nie trzeba używać żadnych funkcji. Wystarczy zaznaczyć miejsce na wynik, wpisać odwołanie do tablicy i zatwierdzić przez *Ctrl+Shift+Enter*. W przypadku obszaru lub tablicy **TA** z przykładu 6.1., po zaznaczeniu obszaru złożonego z dwóch kolumn i trzech wierszy, wprowadzeniu wyrażenia = **TA** + 2 i zatwierdzeniu przez *Ctrl+Shift+Enter*, otrzymamy w wyniku obszar, zawierający następujące dane:

3	4
5	7
8	10

Jeżeli wynik operacji tablicowej umieszczamy w arkuszu, możemy wykorzystać dowolną liczbę początkowych wierszy i kolumn faktycznego wyniku. Wszystko zależy od tego, ile miejsca na wynik przygotujemy. Jednak, jeżeli wynik operacji tablicowej jest argumentem innej funkcji tablicowej lub agregującej, to tablica w takim wyrażeniu zostanie uwzględniona w całości (o ile jest to tablica pełnowartościowa).

## Część wspólna zakresów

Elementy tablicy można również odczytać jako część wspólną wiersza lub zakresu wierszy i kolumn lub zakresu kolumn. Jest to wskazane w przypadku, gdy kolumnom i wierszom tablicy nadano nazwy, jak w przykładzie pokazanym w rozdziale 2., na rysunku 2.4. Sposób ten można wykorzystać do odczytania spójnego fragmentu tablicy, to znaczy fragmentu obejmującego sąsiadujące wiersze i kolumny.

W oparciu o dane widoczne na rysunku 2.4 można na przykład utworzyć odwołanie:

= **T.top:T.wrz Etanol:Metanol**

Obejmuje ono sześć komórek w trzech wierszach i dwu kolumnach. Po zaznaczeniu miejsca na wynik zatwierdzamy formułę przez *Ctrl+Shift+Enter*.

Operator wyznaczania części wspólnej (spacja) może być również użyty do zakresów nieregularnych. Jeśli wynikiem jest pojedyncza komórka, w arkuszu jest wyświetlana jej wartość i do zatwierdzenia wystarczy sam klawisz *Enter*, jeśli zaś obszar, wyświetlana jest zawartość jego komórek po zatwierdzeniu tablicowym w kilku komórkach. W przypadku, gdy wynikiem jest zakres nieregularny, nie można go wyświetlić w arkuszu, możliwe jest natomiast użycie go jako argumentu funkcji akceptującej zakresy nieregularne. Jeśli wynik ma być wyświetlony w jednej komórce, nie jest potrzebne zatwierdzenie tablicowe.

# Dynamiczne generowanie tablic i zakresów o zadanych rozmiarach

## Definicje tablic bazowych

Tablice jednowymiarowe (wierszowe lub kolumnowe) zawierające kolejne liczby naturalne od 1 do  $n$  są bardzo pomocne w obliczeniach dotyczących ciągów i szeregów, jak również w wielu innych, które wykorzystują te tablice w sposób pośredni. Będę je nazywał dalej **tablicami bazowymi** (nie jest to termin powszechnie używany, wprowadziłem go na potrzeby tej publikacji).

Wygenerowanie takich tablic wymaga posłużenia się pewną sztuczką, gdyż nie ma funkcji służącej specjalnie do tego celu. Można mianowicie wykorzystać jedną z funkcji, których pierwotne przeznaczenie było inne. Są to WIERSZ(*przedział*) lub NR.KOLUMNY(*przedział*). Omówiłem je szczegółowo w rozdziale 3., w podrozdziale „Numery wiersza i kolumny w arkuszu”. Funkcje te, użyte w formule tablicowej, zwracają listę numerów wierszy lub kolumn podanych jako argument. Wynik ma formę tablicy jednowymiarowej, niezależnie od wymiarów *przedziału*. W przypadku funkcji WIERSZ jest to wektor kolumnowy, a w przypadku NR.KOLUMNY — wektor wierszowy. Z użyciem tych funkcji wiąże się jednak pewna niedogodność — jeżeli usunie się wiersze bądź kolumny z *przedziału* użytego jako argument lub poprzedzające go (lub doda się w tych miejscach nowe), *przedział* zostanie zmodyfikowany, podobnie jak modyfikowane są w takim przypadku wszystkie adresy w arkuszu. Aby zabezpieczyć się przed tą niedogodnością, a ponadto uzyskać możliwość konstrukcji *przedziału* z segmentów, należy użyć jeszcze funkcji ADR.POŚR(*tekst\_adresu*), podając *tekst\_adresu* jako stałą tekstową w cudzysłowie, wyrażenie tekstowe lub odwołanie do komórki z tekstem adresu. Opis składni i sposób użycia tej funkcji podałem w rozdziale 3., w podrozdziale „Funkcja ADR.POŚR”.

W ten sposób, używając na przykład wyrażenia:

= WIERSZ(ADR.POŚR("1:8")) lub = WIERSZ(ADR.POŚR("A1:A8"))

uzyskamy tablicę  $=\{1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\}$  (od wersji 2010  $=\{1;2;3;4;5;6;7;8\}$ ). Aby „uzmiennić” górną granicę zakresu, należy zamiast stałej 8 wpisać odwołanie do komórki zawierającej tę liczbę. Trzeba pamiętać, że argument jest wyrażeniem tekstowym i trzeba użyć operatora złączenia tekstów &. Wartość w komórce może być wartością numeryczną, Excel sam skonwertuje ją do postaci tekstowej. Jeśli założymy, że wartość 8 umieściliśmy w komórce A5, zmodyfikowane wyrażenie może przyjąć postać:

= WIERSZ(ADR.POŚR("1:"&\$A\$5))

a jeżeli nadaliśmy komórce A5 nazwę  $n$ , jedną z postaci:

= WIERSZ(ADR.POŚR("1:"&n))  
= WIERSZ(ADR.POŚR("A1:A"&n))

Standardowo przy nadawaniu nazwy komórce jej adres jest zapamiętywany jako adres absolutny. Można jednak ręcznie — za pomocą *Menedżera nazw* — zmienić to odwołanie na względne lub mieszane.

Formuła jest dynamiczna w tym sensie, że zmiana wartości  $n$  w arkuszu automatycznie modyfikuje rozmiar tablicy, przy czym jest on ograniczony tylko liczbą wierszy w arkuszu ( $1 \leq n \leq 65\,536$  w Excelu do wersji 2003 oraz  $1 \leq n \leq 1\,048\,576$  od wersji 2007).



Lista numerów wierszy generowana przez funkcję WIERSZ zawsze będzie uporządkowana rosnąco; nie pomoże przestawienie kolejności numerów wierszy typu = WIERSZ(ADR. POŚR("8:1")). Jeśli potrzebna jest odwrócona kolejność, należy użyć wyrażenia w rodzaju = 9 - WIERSZ(ADR. POŚR("1:8")).

Teraz sprawa kluczowa! Tego wyrażenia **nie należy** wpisywać do arkusza (a przynajmniej nie wyłącznie do arkusza), bo wtedy długość tablicy będzie ograniczona wielkością zaznaczonego zakresu. Zamiast tego, trzeba je zapamiętać pod wybraną **nazwą**, na przykład **wn**, posługując się poleceniem *Wstaw/Nazwa/Definiuj* (od wersji 2007 *Formuły/Definiuj nazwę*), a następnie w polu *Nazwy w skoroszybie*: wpisać wybraną nazwę, a w polu *Odwołuje się do*: podane powyżej wyrażenie, po czym zatwierdzić to przyciskiem OK. Jest to formuła tablicowa, jednak przy definiowaniu nazwy **nie trzeba** używać kombinacji *Ctrl+Shift+Enter*. Ponadto wyrażenia z nazwą **wn** są krótsze i bardziej czytelne od tych zawierających konstrukcję WIERSZ(ADR. POŚR("1:"&n)).

Jeśli wyrażenie typu **wn** chcemy wyświetlić w arkuszu, należy pamiętać, że pełną tablicę uzyskamy po jej wprowadzeniu do dokładnie  $n$  komórek. Gdy będzie ich mniej, końcowe liczby nie zostaną wyświetlone; jeśli będzie ich więcej, w „nadmiarowych” pojawią się błędy #N/D!. Aby uniknąć wyświetlania tych błędów, można rozbudować formułę do postaci:

= JEŻELI(WIERSZ()-WIERSZ(**adres\_pocz**)<n; WIERSZ(ADR. POŚR("1:"&n)); "")

**Adres\_pocz** oznacza adres pierwszej komórki, w której wpisujemy formułę. Jeśli chcemy całość później kopiować, powinien to być adres względny. Aby formalnie uniezależnić formułę od konkretnego adresu początkowego, można posłużyć się adresem pośrednim w stylu **WIK1**:

= JEŻELI(WIERSZ()-WIERSZ(ADR. POŚR("WK";))<n; WIERSZ(ADR. POŚR("1:"&n)); "")

Czasem wygodniej jest, gdy ciąg liczb zaczyna się od zera; nie można oczywiście wpisać 0 jako argumentu funkcji ADR. POŚR, bo nie ma zerowego wiersza, ale gdy funkcja WIERSZ już „zrobi swoje”, można odjąć jedynekę:

**wn0** : = WIERSZ(ADR. POŚR("1:"&n+1))-1

Alternatywnymi definicjami **wn** mogą być formuły:

= WIERSZ(\$A\$1:INDEKS(\$A:\$A;n))  
 = WIERSZ(PRZESUNIĘCIE(\$A\$1;;n))

Ich wadą jest podatność na wstawianie lub usuwanie pierwszego wiersza w arkuszu, zaletą — brak funkcji ADR. POŚR, która czasem jest niepożądana.

Jeżeli przy definiowaniu  $n$  wykorzystano adres absolutny, to użycie w dowolnej komórce nazwy **wn** będzie oznaczać taką samą kolumnę (pionową tablicę) liczb od 1 do  $n$ . Jeżeli natomiast przy definiowaniu  $n$  użyto adresu względnego, a komórką aktywną w chwili definiowania była komórka sąsiednia z prawej strony  $n$ , to użycie w dowolnej komórce nazwy **wn** będzie oznaczać

kolumnę (pionową tablicę) liczb od 1 do  $n$ , przy czym  $n$  będzie liczbą pobraną z komórki sąsiadującej z lewej strony z komórką, w której użyto nazwy  $wn$ . Jeżeli wynik będzie wyświetlany tablicowo w kilku komórkach, bierze się pod uwagę pierwszą z nich. Więcej informacji na ten temat podano w rozdziale 7., w podrozdziale „Zasady interpretacji nazw i adresów w wyrażeniach tablicowych”.

Tablica  $wn$  zdefiniowana w powyższy sposób jest więc tablicą kolumnową. W razie potrzeby do zdefiniowania analogicznej tablicy wierszowej można użyć funkcji TRANSPONUJ ( $wn$ ). W zasadzie można też użyć funkcji NR.KOLUMNY, ale jako argument należałoby podać oznaczenia literowe kolumn, co jest dużo mniej wygodne od numerów wierszy. Można też użyć dodatkowo funkcji PRZESUNIĘCIE:

= NR.KOLUMNY (PRZESUNIĘCIE(\$A\$1;0;0;1;n))

albo w wersji uproszczonej:

= NR.KOLUMNY (PRZESUNIĘCIE(\$A\$1;;;n))

Zamiast funkcji PRZESUNIĘCIE można użyć również funkcji INDEKS w taki sposób:

= NR.KOLUMNY (\$A\$1:INDEKS(\$1:\$1;n))

Należy jednak pamiętać, że ostatnie wzory są podatne na modyfikacje przy wstawianiu lub usuwaniu wierszy lub kolumn.

Można również użyć zapisu:

= NR.KOLUMNY (ADR.POŚR("A1:"&ADRES(1;n)))

Ze względu na użycie funkcji ADR.POŚR ten zapis jest odporny na modyfikacje struktury arkusza.

Tablice bazowe można również zdefiniować jako funkcje UDF w VBA [C4].

## Wykorzystanie tablic bazowych do definicji innych tablic

Dysponując tablicą bazową zawierającą liczby od 1 do  $n$ , możemy definiować inne potrzebne tablice jedno- i dwuwymiarowe. W oparciu o definicję wektora  $wn$

$wn$  : = WIERSZ (ADR.POŚR("1:"&n))

można zdefiniować na przykład macierz jednostkową  $E$  o rozmiarze  $n \times n$  za pomocą formuł podanych w rozdziale 4., w punkcie „Macierz jednostkowa”.

Wyrażenia te najlepiej zdefiniować, posługując się formułami nazwanymi za pomocą polecenia *Wstaw/Nazwa/Definiuj* (od wersji 2007 *Formuły/Definiuj nazwę*).  $n$  musi być nazwą stałej zdefiniowanej lub komórki zawierającej rozmiar macierzy. Następnie definiujemy wektor kolumnowy  $wn$  i wreszcie macierz  $E$ .

W Excelu 2013+ zadanie to można wykonać znacznie prościej, dzięki nowej funkcji predefiniowanej. Wystarczy napisać  $E$  : = MACIERZ.JEDNOSTKOWA( $n$ )

Po wykonaniu zwykłego mnożenia dowolnej tablicy wierszowej lub kolumnowej o długości  $n$  przez macierz jednostkową, otrzymamy macierz diagonalną z elementami diagonalnymi równymi elementom użytej tablicy.

Posługując się zdefiniowaną wcześniej tablicą  $wn$ , można łatwo zdefiniować inne często potrzebne tablice, na przykład:

- tablica złożona z  $n$  zer: =  $wn-wn$  albo =  $wn*0$
- tablica złożona z  $n$  jedynek: =  $wn-wn+1$  albo =  $wn*0+1$ , albo =  $wn^0$ , albo =  $1^{wn}$
- tablica złożona z  $n$  jedynek i minus jedynek na przemian: =  $-1^{wn}$  albo =  $-1^{(wn+1)}$ , albo =  $-(-1^{wn})$

Pierwszy wariant ostatniego wzoru zwraca ciąg rozpoczynający się od  $-1$ , dwa następne startują od 1.

W tym przypadku problemem nie jest uzyskanie konkretnej wartości, lecz jej powtórzenie w tablicy ściśle określoną liczbę razy.

Jeśli początkowa liczba w tablicy bazowej  $wn$  ma być inna niż 1, można „uzmiennić” obie granice lub użyć wyrażen arytmetycznych typu  $wn + 5$  bądź  $wn - n/2$ .

Jeżeli w definicji tablicy  $wn$  użyto zmiennej  $n$ , która ma aktualnie wartość 5, to wyrażenie:

$$= a+(wn-1)*(b-a)/(n-1)$$

definiuje kolumnę pięciu wartości rozłożonych równomiernie pomiędzy  $a$  i  $b$ . Analogicznie, posługując się definicją  $wn0$  i wartością  $n = 4$ , można podobny efekt uzyskać po obliczeniu:

$$= a+wn0*(b-a)/n$$

Temu wyrażeniu można nadać nową nazwę, na przykład  $x$ .  $x$  pozostanie wyrażeniem dynamicznym tak długo, jak długo jest używane jako nazwa. Każda zmiana wartości  $n$  spowoduje zmianę rozmiarów i zawartości  $x$ . Jeżeli jednak wprowadzimy  $x$  do arkusza, wpisując do komórki lub zakresu komórek =  $x$  i zatwierdzając przez *Enter* lub *Ctrl+Shift+Enter*, ta komórka czy zakres nie będzie reprezentować już całej tablicy  $x$ , a tylko to, co jest wyświetlone w arkuszu, czyli pierwszą lub kilka pierwszych komórek tablicy  $x$ . Rozmiar tego obszaru pozostaje statyczny, czyli nie zmienia się przy zmianie  $n$ .

Każda formuła tablicowa w danej chwili musi dawać w wyniku tablicę o określonej wielkości i kształcie, zależnie od składowych formuły i zasad wykonywania działań. Z formułą, która reprezentuje tablicę dynamiczną, nie jest jednak związany żaden adres, w związku z tym ani nazwa tablicy dynamicznej, ani nazwa takiej formuły nie może być argumentem funkcji, która wymaga podania adresu, jak choćby funkcja PRZESUNIĘCIE.

## Plan eksperymentu czynnikowego

Przykładem zastosowania tablic dynamicznych może być wygenerowanie planu eksperymentu czynnikowego typu  $2^k$ . Jest to jeden z podstawowych planów stosowanych do badania zależności między różnymi wielkościami. Plan jest tablicą, która ma tyle kolumn, ile jest zmiennych

niezależnych, oraz jedną dodatkową kolumnę wyrazu wolnego. Liczba wierszy odpowiada liczbie doświadczeń, które należy wykonać, równej  $2^k$ . Wartości czynników są kodowane liczbami 1 lub  $-1$ , przy czym chodzi o to, by zrealizować wszystkie możliwe kombinacje (w znaczeniu potocznym, nie kombinatorycznym) tych liczb. Wyjątkiem jest pomocnicza kolumna wyrazu wolnego, umieszczana zwykle jako pierwsza, która zawiera same jedynki.

Liczbę czynników wstawiono do komórki C3, której nadano nazwę *lk*. Następnie zdefiniowano dwa wektory — jeden wierszowy o nazwie *wlk*, zawierający liczby od 0 do *lk*, a drugi kolumnowy o nazwie *nlk*, złożony z liczb od 1 do  $2^{lk}$ . Definicje tych wektorów mają postać:

$wlk := \text{TRANSPONUJ}(\text{WIERSZ}(\text{ADR. POŚR}("1:"\&lk+1)))-1$

albo  $wlk := \text{NR. KOLUMNY}(\text{ADR. POŚR}("A1:"\&\text{ADRES}(1;lk+1)))-1$

$nlk := \text{WIERSZ}(\text{ADR. POŚR}("1:"\&2^{lk}))$

Wreszcie, w oparciu o wektory *wlk* i *nlk*, zdefiniowano tablicę *plan*, obejmującą *wlk* kolumn i *nlk* wierszy, zawierającą niezbędne kombinacje dodatnich i ujemnych jedynek. Tablicę opisano wzorem:

$plan := -1^{\text{ZAOKR.DO.CAŁK}((nlk-1)/2^{(lk-wlk)})}$

Funkcja ZAOKR.DO.CAŁK zwraca część całkowitą wyniku dzielenia  $nlk-1$  przez odpowiednią potęgę dwójki. Jeżeli wynik funkcji jest liczbą parzystą, formuła przyjmuje wartość 1, a jeśli liczbą nieparzystą — zwracana jest wartość  $-1$ . Na rysunku pokazano całą tablicę *plan* oraz wygenerowane na jej podstawie dodatkowe kolumny iloczynów wybranych czynników.

Taki plan wygenerowany dla  $k = 4$  pokazano na rysunku 6.1.

## Selektywne sumowanie elementów tablic dwuwymiarowych

Użycie funkcji SUMA wprost do elementów tablicy dwuwymiarowej prowadzi do obliczenia sumy ogólnej wszystkich elementów tablicy. W obliczeniach często potrzebne są sumy wierszy lub kolumn. W takim przypadku należy wykorzystać dodatkowe funkcje do selekcji danych. Najprostszym rozwiązaniem jest użycie funkcji INDEKS do wyodrębnienia wiersza lub kolumny do sumowania. Jeżeli chcemy obliczyć sumę określonego wiersza lub kolumny, zastosujemy formułę:

$= \text{SUMA}(\text{INDEKS}(tbl; nr\_wiersza;))$

lub

$= \text{SUMA}(\text{INDEKS}(tbl;;nr\_kolumny))$

Są to formuły zwykle, każda zwraca jedną wartość — odpowiednio dla wiersza lub kolumny.

Jeśli chcemy uzyskać wynik w postaci tablicy (kolumny z podsumowaniem wierszy lub wiersza z podsumowaniem kolumn), funkcja INDEKS musiałaby być użyta w sposób niestandardowy (z argumentem indeksowym w postaci tablicy) [B20].

Plan\_czynnikowy.xlsx - Microsoft Excel użytek niekomercyjny

Narzędzia główne Wstawianie Układ strony Formuły Dane Recenzja Widok Deweloper

D9  $f_x$  [=plan]

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	<b>Plan eksperymentu czynnikowego 2<sup>k</sup></b>												
2													
3		Liczba czynników <i>lk</i> =		4									
4		Zakres liczb od 0 do <i>lk</i> (wiersz):		<i>wlk</i>									
5		Numery doświadczeń od 1 do 2 <sup>k</sup> (kolumna):		<i>nlk</i>									
6													
7		<i>nlk</i>	"x"& <i>wlk</i> =x0	x1	x2	x3	x4		x1*x2	x3*x4	x1*x2*x3		
8													
9		1		1	1	1	1		1	1	1		
10		2		1	1	1	1	-1		1	-1	1	
11		3		1	1	1	-1	1		1	-1	-1	
12		4		1	1	1	-1	-1		1	1	-1	
13		5		1	1	-1	1	1		-1	1	-1	
14		6		1	1	-1	1	-1		-1	-1	-1	
15		7		1	1	-1	-1	1		-1	-1	1	
16		8		1	1	-1	-1	-1		-1	1	1	
17		9		1	-1	1	1	1		-1	1	-1	
18		10		1	-1	1	1	-1		-1	-1	-1	
19		11		1	-1	1	-1	1		-1	-1	1	
20		12		1	-1	1	-1	-1		-1	1	1	
21		13		1	-1	-1	1	1		1	1	1	
22		14		1	-1	-1	1	-1		1	-1	1	
23		15		1	-1	-1	-1	1		1	-1	-1	
24		16		1	-1	-1	-1	-1		1	1	-1	

**RYSUNEK 6.1.** Plan całkowitego eksperymentu czynnikowego dla czterech czynników, wygenerowany za pomocą formuły tablicowej

Na szczęście istnieje też sposób alternatywny — z wykorzystaniem funkcji `MACIERZ.ILOCZYN`. Mieści się on w standardzie użycia tej funkcji, ale jest nieco mniej intuicyjny i nadaje się tylko do sumowania. Sposób jej działania opisałem w rozdziale 4., w punkcie „Mnożenie macierzowe”.

Użycie funkcji `MACIERZ.ILOCZYN` polega na przemnożeniu macierzowo tablicy *tabl* przez kolumnę lub wiersz, złożone z samych jedynek. Wynik otrzymujemy w formie kolumny lub wiersza.

Należy zastosować jedną z formuł tablicowych:

$$= \text{MACIERZ.ILOCZYN}(\text{tabl}; \text{TRANSPONUJ}(\text{NR.KOLUMNY}(\text{tabl}))^0)$$

w odniesieniu do sum wierszy lub

$$= \text{MACIERZ.ILOCZYN}(\text{TRANSPONUJ}(\text{WIERSZ}(\text{tabl})^0); \text{tabl})$$

w odniesieniu do sum w kolumnach.

Formuły z funkcją `MACIERZ.ILOCZYN` są pełnowartościowe, to znaczy mogą stać się elementem większego wyrażenia.



## Łączenie tablic z wykorzystaniem funkcji MACIERZ.ILOCZYN

Już kilkakrotnie zajmowaliśmy się zagadnieniem łączenia obszarów i tablic w jedną spójną tablicę. Wiemy już, że stosunkowo łatwo można połączyć zakresy lub tablice jednowymiarowe (wiersze lub kolumny), a w przypadku obszarów można za pomocą funkcji PRZESUNIĘCIE połączyć również struktury dwuwymiarowe. Najwięcej problemów przysparzają tablice dwuwymiarowe, które nie są umieszczone bezpośrednio w arkuszu, lecz są wynikiem obliczeń. Jak pokazałem w przykładzie 6.2, można je połączyć za pomocą funkcji INDEKS po dekompozycji na pojedyncze wiersze lub kolumny. W przypadku większych tablic sposób ten jednak nie jest zbyt wygodny. Teraz, po zapoznaniu się z możliwościami wykorzystania tablic bazowych, przyszedł czas na omówienie jeszcze jednej metody łączenia tablic z użyciem funkcji MACIERZ.ILOCZYN. Dzięki temu, że jest to funkcja tablicowa, nie podlega ograniczeniom wynikającym z zasady superpozycji i odpowiednio zaprogramowana, może wykonać nasze zadanie. Metodę tę można stosować zarówno do tablic, jak i do obszarów, jest jednak konieczne, aby dane były wyłącznie liczbowe. W zakresie danych nie może być również pustych komórek (funkcja MACIERZ.ILOCZYN jest bardzo wrażliwa na błędne dane).

Omówię teraz pokrótce ideę rozwiązania.

Najpierw ustalamy rozmiary tablicy wynikowej i upewniamy się, czy łączone tablice są dopasowane wymiarami. Tablice łączone w strukturę poziomą (kolumna obok kolumny) muszą mieć taką samą liczbę wierszy, zaś łączone w strukturę pionową (wiersz pod wierszem) muszą mieć taką samą liczbę kolumn.

Przy tworzeniu struktury poziomej obliczamy sumaryczną liczbę kolumn w połączonej tablicy i definiujemy poziomą tablicę bazową z numerami kolumn. Następnie do każdej tablicy składowej definiujemy pionowe tablice bazowe z liczbami od 1 do liczby kolumn w każdej tablicy składowej. Przy tworzeniu struktury pionowej postępujemy analogicznie, tylko z zamianą wierszy na kolumny i odwrotnie.

Generujemy odpowiednie tablice zero-jedynkowe, które pomnożone macierzowo przez tablice składowe dadzą w wyniku tablice o rozmiarach tablicy wynikowej, odwzorowujące tablice składowe uzupełnione zerami.

Sumujemy tablice cząstkowe, uzyskując tablicę wynikową.

### Przykład 6.6.

Rozpatrzmy połączenie poziome trzech tablic:  $TA$ ,  $TB$  i  $TC$ . Tablica  $TA$  ma  $a$  kolumn,  $TB$  —  $b$  kolumn, a  $TC$  —  $c$ . kolumn. Liczba wierszy nie ma znaczenia, musi być jedynie taka sama w każdej tablicy. Łączna liczba kolumn tablicy wynikowej jest równa  $s = a + b + c$ . Definiujemy poziomą tablicę bazową o długości  $s$ :

```
ks = TRANSPONUJ(WIERSZ(ADR.POŚR("1:"&s)))
```

oraz pionowe tablice bazowe o długościach odpowiadających liczbie kolumn każdej z tablic składowych:

```

wa := WIERSZ(ADR.POŚR("1:"&a))
wb := WIERSZ(ADR.POŚR("1:"&b))
wc := WIERSZ(ADR.POŚR("1:"&c.))

```

Tworzymy cząstkowe tablice wynikowe, odpowiadające tablicom składowym uzupełnionym zerami, które następnie dodajemy:

```

= MACIERZ.ILOCZYN(TA; --(ks=wa)) + MACIERZ.ILOCZYN(TB; --(ks=wb+a)) +
↳MACIERZ.ILOCZYN(TC; --(ks=wc+b+a))

```

Wynik jest tablicą **TPW**, może być więc używany do dalszych obliczeń i nie wymaga zatwierdzenia tablicowego, jeśli tylko wynik końcowy jest pojedynczą liczbą. Można przykładowo użyć takiej tablicy jako drugiego argumentu funkcji WYSZUKAJ.PIONOWO. Jeśli rozmiary tablic składowych są stałe i niezbyt wielkie, wygodniejsze może być użycie odpowiednich stałych tablicowych zamiast formuł **wa**, **wb** i **wc**. Procedura łączenia tablic została zilustrowana na rysunku 6.2. Tu użyto do połączenia obszarów, aby przekaz był czytelny, ale obszary mogą być używane na równi z tablicami.

	A	B	C	D	E	F	G	H	I	J
1	Łączenie tablic za pomocą funkcji MACIERZ.ILOCZYN									
2		s =	7							
3	Tablica A	a =	2			--(ks=wa)				
4	3,5	6,7		1	0	0	0	0	0	0
5	4,1	5		0	1	0	0	0	0	0
6										
7	Tablica B	b =	3			--(ks=wb+a)				
8	1	2	2,6	0	0	1	0	0	0	0
9	5,8	3	7,1	0	0	0	1	0	0	0
10				0	0	0	0	1	0	0
11	Tablica C	c. =	2							
12	2,5	8,1				--(ks=wc+b+a)				
13	4,5	7,2		0	0	0	0	0	1	0
14				0	0	0	0	0	0	1
15	Wynik połączenia:									
16	3,5	6,7	1	2	2,6	2,5	8,1			
17	4,1	5	5,8	3	7,1	4,5	7,2			
18										

**RYСУNEK 6.2.** Łączenie tablic z wykorzystaniem funkcji MACIERZ.ILOCZYN

## Zakresy dynamiczne

W poprzednich punktach przedstawiłem sposób definiowania tablic o zmiennych rozmiarach. Taka konstrukcja jest wystarczająca do wielu obliczeń, ale w niektórych przypadkach konieczne jest odwołanie do zakresu komórek w arkuszu. Należy zatem umieć definiować zakres również w sposób dynamiczny. Można sobie wyobrazić kilka możliwych sposobów realizacji tego zagadnienia. Przedstawię je kolejno.

## Różne typy adresu początku i końca

Jak już wcześniej wspomniałem, przy definiowaniu zakresu można na przykład jego początek opisać odwołaniem absolutnym, a koniec — względnym lub mieszanym. W takim przypadku rozmiary zakresu będą zależęć od tego, która komórka była aktywna w momencie definiowania i w której użyto odwołania. Załóżmy na przykład, że komórką aktywną jest C1 i za pomocą *Menedżera nazw* tworzymy definicję:

```
obszar : = $B$1:B$1
```

Tak zdefiniowany **obszar** oznacza zakres komórek od B1 do komórki w pierwszym wierszu arkusza i kolumnie sąsiadującej z lewej strony z kolumną komórki, w której użyto nazwy w formule. Jeżeli w komórce E3 umieścimy formułę

```
= ŚREDNIA(obszar)
```

to obliczona zostanie średnia wartości z zakresu B1:D1. Jeśli skopiuje się formułę w prawo do komórek w kolejnych kolumnach, uśredniany zakres będzie coraz większy.

Ten sposób definiowania zakresów dynamicznych jest najprostszy, ale zarazem najmniej uniwersalny. Przy kopiowaniu danych w inne miejsce na ogół konieczna jest ręczna korekta formuły.

## Stały początek obszaru, długość określona przez położenie formuły

Ogólną koncepcję opisaną w poprzednim podpunkcie można zrealizować również formułą, wykorzystując jako początek zakresu komórkę z charakterystyczną zawartością lub pustą, a koniec uzależniając od położenia komórki z formułą. Wiele formuł dynamicznych konstruowanych jest w taki sposób, że odnoszą się do zakresu komórek położonych nad lub przed komórką (po jej lewej stronie), w której znajduje się formuła. Przykładowo, wyobraźmy sobie, że umieszczamy formuły w kolumnie E, przy czym pozostawiamy nad nimi pustą komórkę. Zakres dynamiczny ma obejmować obszar od pustej komórki do znajdującej się bezpośrednio nad naszą formułą. Załóżmy, że pusta komórka ma adres E5, a pierwszą formułę wprowadzamy do E6. W pierwszej komórce zakres będzie obejmował tylko komórkę E5, a w miarę kopiowania w dół będzie się wydłużał. Początek obszaru określimy jako ostatnią pustą komórkę w kolumnie E przed komórką bieżącą. Zrealizujemy to za pomocą formuły, której można nadać nazwę i wykorzystywać jako element złożonych obliczeń.

```
zw : = INDEKS(E$1:E5; MAX(JEŻELI(CZY.PUSTA(E$1:E5);WIERSZ(E$1:E5)))):E5
```

Przy wprowadzaniu tej definicji komórką aktywną jest E6. Formuła jest tablicowa, ale jeśli nadamy jej nazwę, nie będzie to miało praktycznego znaczenia. Zakładam, że użytkownik nie będzie wstawiał ani usuwał pierwszego wiersza. Dzięki użyciu względnego adresu kolumny, po zapisaniu definicja dopasowuje się do komórki, w której jest używana.

## Stały początek obszaru, długość określona przez liczbę komórek

Wyobraźmy sobie, że tablicę bazową **wn**, zdefiniowaną jako:

```
wn : = WIERSZ(ADR.POŚR("1:"&n))
```

wprowadzimy do wybranej kolumny arkusza, zaczynając na przykład od komórki C6, której możemy nadać nazwę **pocz. Zakres**, do którego wprowadzono formułę, powinien być zaplanowany z zapasem, gdyż określa on maksymalną wartość  $n$ , jakiej można w praktyce użyć. Na ogół nie ma sensu odwoływać się do takiego zakresu w całości. Zawiera on tylko  $n$  wartości, pozostałe komórki wypełnione są kodem braku danych #N/D!. Chcemy tak zdefiniować nasz zakres, aby obejmował tylko  $n$  początkowych komórek, zawierających liczby od 1 do  $n$ .

Mamy w Excelu trzy funkcje, które można wykorzystać w tym celu. Są to INDEKS, ADR.POŚR i PRZESUNIĘCIE. Zgodnie ze składnią tych funkcji możemy zapisać formuły:

- = pocz:INDEKS(**zakres**;  $n$ )
- = pocz:ADR.POŚR(ADRES(WIERSZ(**pocz**)+ $n$ -1; NR.KOLUMNY(**pocz**))
- = pocz:ADR.POŚR("W" & WIERSZ(**pocz**)+ $n$ -1 & "K" & NR.KOLUMNY(**pocz**);)
- = PRZESUNIĘCIE(**pocz**; 0; 0;  $n$ )

Drugą formułę można uprościć, jeśli uwzględni się konkretny adres początku zakresu:

- = ADR.POŚR("C6:C" &  $n$ +5)

Jeśli chcemy, aby zakres rozpoczynał się nie od konkretnej komórki, ale na przykład bezpośrednio pod komórką, w której wpisujemy odwołanie do niego, można posłużyć się formułą z adresem pośrednim zapisanym w stylu **WIK1**:

- = ADR.POŚR("W[1]K:W[" &  $n$  & "]K");)

Każda z powyższych formuł zwraca zakres obejmujący  $n$  komórek. Formule takiej można nadać nazwę i używać jej wszędzie tam, gdzie wymagane jest użycie odwołania, czyli jako argument funkcji odwołaniowej lub przy definiowaniu źródła listy rozwijanej. Istotne przy tym jest, żeby **zakres** w arkuszu był faktycznie wypełniony odpowiednimi danymi. Gdyby jednak taka kolumna psuła estetykę arkusza, można ją ukryć.

## Formuły tablicowe uwzględniające faktyczne rozmiary obszaru formuły

Zazwyczaj, gdy zapisujemy formułę tablicową, z góry przewidujemy rozmiary obszaru wynikowego i rezerwujemy w arkuszu tyle miejsca, ile potrzeba do wyświetlenia wszystkich wyników. Gdy zaznaczymy zbyt duży obszar, w tych dodatkowych komórkach pojawią się kody błędów (najczęściej #N/D!). Zwykle staramy się unikać takich sytuacji ze względów estetycznych. Możemy wykorzystać różne sposoby maskowania błędów, opisane w rozdziale 8., w punkcie „Maskowanie wartości błędów”. Problemy stwarzają szczególnie formuły dynamiczne, bo w ich przypadku rozmiary obszaru wynikowego są zmienne, a jeśli chcemy pokazać wyniki w arkuszu, to musimy użyć do tego obszaru statycznego.

Jeśli weźmiemy pod uwagę typową formułę tablicy bazowej:

- wn** := WIERSZ(ADR.POŚR("1:"& $n$ ))

w której  $n$  reprezentuje wartość liczbową, to w formułach złożonych formuła ta oznacza tablicę dokładnie  $n$ -elementową i jeśli będzie argumentem funkcji zwykłej, w arkuszu wymagać będzie dokładnie  $n$  komórek. Jak zmodyfikować tę formułę, aby w przypadku zaznaczenia większej

liczby komórek, w tych nadmiarowych wyświetlane były puste teksty zamiast błędów? Ta zmodyfikowana formuła musi umieć odczytać nie tylko wartość  $n$ , ale również rozmiary obszaru, do którego została wprowadzona. Takich informacji dostarczyć mogą funkcje WIERSZ() lub NR.KOLUMNY(), użyte bez argumentów. Wyrażenie:

$$l_{nr} := \text{WIERSZ}() - \text{WIERSZ}(ap) + 1$$

gdzie  $ap$  jest adresem komórki, w której rozpoczyna się zakres formuły, zwraca listę kolejnych liczb całkowitych od 1 do wypełnienia całego zakresu. Zamiast podawać adres  $ap$  bezpośrednio, można posłużyć się formułą funkcji ADR.POŚR z argumentem w stylu WIKI:

$$l_{nr} := \text{WIERSZ}() - \text{WIERSZ}(\text{ADR.POŚR}("WK";)) + 1$$

Formuły tego rodzaju mogą być wygodnie używane do wyświetlania w arkuszu tyłu wyników, ile chcemy zaznaczyć wierszy. Ale tego typu formuły nie nadają się do wykorzystania jako argumenty funkcji agregujących. Gdybyśmy chcieli obliczyć sumę numerów zaznaczonych wierszy od 1 do 7 (przykładowo), to formułę

$$\text{SUMA}(l_{nr})$$

musielibyśmy wprowadzić do siedmiu komórek i otrzymalibyśmy siedem identycznych wyników.

Jeśli chcielibyśmy ograniczyć długość ciągu liczbowego do  $n$  wartości, powinniśmy wykorzystać funkcję JEŻELI:

$$= \text{JEŻELI}(l_{nr} \leq n; l_{nr}; "")$$

Aby uzyskać formułę w miarę uniwersalną, którą można by wykorzystywać zarówno w funkcjach zwykłych, jak i agregujących, należy połączyć oba podejścia w jednej formule:

$$= \text{JEŻELI}(l_{nr} \leq n; wn; "")$$

Gdy chcemy użyć tej konstrukcji z funkcją zwykłą, musimy ją wprowadzić do wnętrza funkcji JEŻELI, na przykład:

$$= \text{JEŻELI}(l_{nr} \leq n; \text{LN}(wn); "")$$

Funkcję agregującą można umieścić podobnie lub na zewnątrz — jeśli wyrażenie będzie zatwierdzane w jednej komórce, forma:

$$= \text{JEŻELI}(l_{nr} \leq n; \text{SUMA}(wn); "")$$

nie wymaga zatwierdzenia tablicowego (o ile na  $wn$  nie są wykonywane żadne operacje), natomiast forma:

$$= \text{SUMA}(\text{JEŻELI}(l_{nr} \leq n; wn; ""))$$

wymaga takiego zatwierdzenia.

Poniżej przytaczam inne warianty formuł wykorzystujących funkcje WIERSZ() lub NR.KOLUMNY(). Przykładowo, obliczając wartości wyrażień:

$$l_w := \text{MAX}(\text{WIERSZ}()) - \text{MIN}(\text{WIERSZ}()) + 1$$

lub

$$\text{SUMA(WIERSZ())}/\text{ŚREDNIA(WIERSZ())}$$

możemy określić liczbę wierszy, które zajmuje formuła tablicowa.

Analogicznie można obliczyć liczbę kolumn:

$$\mathbf{lk} : = \text{MAX}(\text{NR. KOLUMNY()}) - \text{MIN}(\text{NR. KOLUMNY()}) + 1$$

Proszę zauważyć, że nie można tu wykorzystać funkcji ILE.WIERSZY ani LICZBA.KOLUMN, gdyż one wymagają, aby adres zakresu był jawnie określony.

Można teraz wykorzystać formuły nazwane **lw** i **lk**, na przykład do utworzenia kopii zakresu komórek, począwszy od znanej komórki początkowej (na przykład A3), obejmującej tyle komórek, ile zaznaczymy. Można to zrobić kilkoma sposobami, stosując jedną z funkcji PRZESUNIĘCIE, INDEKS lub ADR.POŚR.

```
= PRZESUNIĘCIE($A$3;; lw; lk)
= $A$3:PRZESUNIĘCIE($A$3; lw-1; lk-1)
= $A$3:INDEKS($A$3:$AA$100; lw; lk)
= ADR.POŚR("A3:"&INDEKS(ADRES(WIERSZ($A$3)+lw-1; NR.KOLUMNY($A$3)+lk-1);1))
```

W przypadku formuły z funkcją INDEKS należy podać w argumencie zakres dostatecznie duży, aby objął on planowaną kopię. Występująca w przypadku formuły z funkcją ADR.POŚR potrzeba użycia dodatkowo funkcji INDEKS została wyjaśniona w rozdziale 7., w przykładzie 7.21.

Podane cztery formuły są pełnowartościowe i mogą być wykorzystane jako argumenty innych funkcji. Jeśli jednak użyjemy ich jako argumentu funkcji agregującej, będziemy musieli zatwierdzić formułę w całym obszarze określającym wielkość kopii, otrzymamy zatem wynik powtórzony wielokrotnie. Można tego uniknąć, dopisując do formuły warunek wyświetlenia wyniku tylko w pierwszej komórce:

$$= \text{JEŻELI}((\text{WIERSZ()}=\text{MIN}(\text{WIERSZ()})) * (\text{NR. KOLUMNY()}=\text{MIN}(\text{NR. KOLUMNY()})); \text{SUMA}(\mathbf{kopia}); "")$$

W powyższym wzorze **kopia** oznacza jedną z czterech formuł podanych powyżej, a SUMA została użyta jako przykład funkcji agregującej.

Inne typy formuł uwzględniających rozmiary obszaru, w którym zostały zatwierdzone, wykorzystują struktury pseudozakresów [B21].

# Skorowidz

## A

adresowanie, 40  
adresy, 31, 41, 553  
  absolutne, 160  
  bezwzględne, 57  
  komórek, 40  
  komórek scalonych, 44  
  kwalifikowane, 46  
  lokalne, 43  
  obszaru, 40, 43  
  odczytywanie, 88  
  różnego typu, 242  
  URL, 497  
  w stylu A1, 160  
  w stylu W1K1, 161  
  wprowadzane do formuł, 42  
  wyliczanie, 88  
  względne, 160  
aktywny, 554  
alternatywa kryteriów, 403  
analiza  
  ciągów, 328  
  danych, 449  
  zawartości arkusza, 292  
Analysis ToolPak, 141  
aparatury fotograficznej, 275  
apostrof, 91  
aproksymacja funkcji, 555  
argumenty  
  funkcji, 77, 214, 555  
  grupowe, 123, 125, 130  
  niedopasowane, 111  
  opcjonalne, 227  
  puste, 227, 228  
  w postaci tablic, 126

arkusz kalkulacyjny, 23, 556  
  analiza zawartości, 292  
  odwołania wielopoziomowe, 142  
  przeliczanie, 179  
  rodzaje informacji, 31  
  struktura, 31  
  wyświetlanie danych, 147  
  zmiana wyglądu, 182  
ASCII, 52, 75  
autofiltr, 399  
automatyczne  
  formatowanie, 501  
  tworzenie hiperłączy, 500  
autouzupełnianie formuł, 72

## B

baza danych, 389, 475  
  właściwości pierwiastków chemicznych, 393  
  z tabelą danych, 448  
bieżący, 554  
błąd  
  #ADR! , 97, 280  
  #ARG! , 108, 250, 280  
  #DZIEL/0! , 280  
  #LICZBA! , 80, 280  
  #N/D! , 111, 280  
  #NAZWA? , 280  
  #ZERO! , 280  
błędy  
  identyfikacja przyczyn, 294  
  kody, 278  
  kontrola, 278  
  obliczeń, 280  
  sygnalizacja, 278

**C**

ciąg, 556  
 cyfr, 437  
 cieniowanie wierszy, 382  
 czas, 36, 352, 354  
 systemowy, 356  
 czcionki, 182  
 część wspólna zakresów, 55  
 czujki, 295

**D**

dane  
 grupowe, 105, 247, 248  
 liczbowe uporządkowane, 316, 321  
 nieliczbowe, 327  
 rozproszone, 332  
 w tablicy, 461  
 w zakresie, 459  
 zewnętrzne, 46, 48  
 data, 36, 352, 355  
 definiowanie  
 formatów warunkowych, 362  
 kwerendy, 477  
 tablic bazowych, 197  
 dereferencja, 234, 557  
 diagnozowanie błędów obliczeń, 278  
 dodatek Analysis ToolPak, 141  
 dwukropek, 106  
 dynamiczne generowanie tablic, 197  
 dysk sieciowy OneDrive, 27  
 działania arytmetyczne, 110

**E**

edycja, 153  
 formuł tablicowych, 176  
 elementy  
 składowe tabeli, 134  
 tablicy, 105

**F**

filtr  
 nazw, 62  
 zaawansowany, 401  
 alternatywa kryteriów, 403  
 koniunkcja kryteriów, 403  
 kryteria z nagłówkami kolumn, 402

filtrowanie, 390, 392  
 formanty pola kombi, 276  
 format, 559  
 Czas, 36  
 Data, 36  
 Liczbowe, 33, 344, 348  
 Naukowe, 34  
 Ogólne, 33, 36, 344  
 Procentowe, 35  
 Tekstowe, 37, 348  
 Ułamkowe, 34, 344, 350  
 formatowanie, 559  
 daty i czasu, 352  
 komórki, 340  
 paski danych, 367  
 pustych, 373  
 skala barw, 365  
 zakresu, 365, 367, 369  
 zestaw ikon, 369  
 zawierających tekst, 372  
 zawierających wartości błędu, 373  
 liczb dziesiętnych, 374  
 liczby, daty lub godziny, 373  
 pobranych danych, 481  
 sekcji, 344  
 wartości  
 powyżej średniej, 376  
 sklasyfikowanych jako pierwsze, 375  
 unikatowych, 378  
 zduplikowanych, 378  
 warunkowe, 275, 361  
 opcje, 362, 364  
 reguły, 385  
 ustalanie zakresu, 381  
 usuwanie, 384  
 zamiana na stałe, 384  
 znajdowanie komórek, 386  
 według daty, 373  
 według tekstu, 373  
 formularz, 389  
 formuła, 32, 39, 560  
 „zatrzaškowa”, 560  
 dynamiczna, 580  
 tablicowa, 39, 108, 130, 169, 206  
 edycja, 176  
 kopiowanie, 178  
 przekształcanie w formułę zwykłą, 178  
 zamiana wyników, 177  
 zmiana zakresu, 176  
 zwykła, 39, 178



- formuły
- autouzupełnianie, 72
  - dla komórek do sformatowania, 380
  - kopiowanie, 157, 162
  - łączenie, 169
  - maskowanie błędów, 282
  - modyfikacja adresów, 160
  - przenoszenie komórek, 173
  - sprawdzanie poprawności, 292
  - szacowanie, 296
  - ustalanie rankingu, 456
  - w komórkach scalonych, 167
  - wprowadzanie adresów, 42
  - wprowadzanie funkcji, 71
  - z odwołaniami względnymi, 163
  - zapis w Excelu, 75
  - zapis w matematyce, 75
  - zawierające nazwy, 162
- FTP, File Transfer Protocol, 497
- funkcja, 561
- ADR.POŚR, 45, 92, 96, 138, 181
  - ADRES, 90
  - AGREGUJ, 133, 422
  - CZĘSTOŚĆ, 144, 325, 328, 462, 468
  - CZY.BŁ, 278
  - CZY.BŁĄD, 278
  - CZY.BRAK, 278
  - DZIŚ, 181
  - HIPERŁĄCZE, 502, 506, 508
    - z argumentem tablicowym, 507
  - ILE.LICZB, 227, 336
  - ILE.NIEPUSTYCH, 227
  - ILE.WIERSZY, 90, 144
  - INDEKS, 138, 185, 187, 189, 208, 252, 318
    - forma odwołaniowa, 189
  - INFO, 103, 181
  - JEŻELI, 82, 84, 123, 125, 129, 130
    - wielokrotne zagnieżdżenie, 190
  - JEŻELI zwracająca wartość pustą, 83
  - JEŻELI.BŁĄD, 87, 112, 144, 282, 290, 337
  - JEŻELI.ND, 112
  - KOMÓRKA, 57, 100–103, 181, 359
  - L, 99
  - LICZ.JEŻELI, 143, 281, 429, 433, 437, 464
  - LICZ.WARUNKI, 240, 435
  - LICZBA.CAŁK, 79
  - LICZBA.KOLUMNY, 90
  - LOS, 181
  - LOS.ZAKR, 181
  - MACIERZ.ILOCZYN, 117, 120, 122, 202
  - MAKS.WARUNKÓW, 440
  - MAX, 208, 451
  - MAX.A, 225
  - MAX.K, 454
  - MIN.K, 454
  - MOD, 79, 382
  - MODUŁ.LICZBY, 78
  - NR.BŁĘDU, 280
  - NR.KOLUMNY, 88, 450
  - OBLICZ, 303
  - PIERWIASTEK, 78
  - PODAJ.POZYCJĘ, 97, 98, 318, 321, 327, 472
  - POTĘGA, 78
  - POZYCJA, 145
  - PRZEŁĄCZ, 86
  - PRZESUNIĘCIE, 57, 138, 181, 191, 194, 208
  - RADIANY, 77
  - RANDBETWEEN, 181
  - SILNIA, 239
  - STOPNIE, 77
  - SUMA.ILOCZYNÓW, 81, 252
  - SUMA.JEŻELI, 143
  - SUMA.WARUNKÓW, 240, 435
  - SUMA.X2.M.Y2, 81
  - SUMA.X2.P.Y2, 81
  - SUMA.XMY.2, 81
  - SUMY.CZĘŚCIOWE, 133, 419–421
  - SUMY.KWADRATÓW, 81
  - ŚREDNIA.JEŻELI, 240, 436
  - ŚREDNIA.WARUNKÓW, 240, 436
  - T, 99
  - TERAZ, 181
  - TRANSPONUJ, 117–120
  - TYP, 279
  - WARUNKI, 85
  - WIERSZ, 88, 197, 336, 382, 450
  - WYBIERZ, 83, 84, 123, 126–130
  - WYST.NAJCZĘŚCIEJ, 141
  - WYSZUKAJ, 315, 318, 467
    - składnia tablicowa, 318
    - składnia wektorowa, 315
  - WYSZUKAJ.PIONOWO, 311, 314
  - WYSZUKAJ.POZIOMO, 311
  - XOR, 233
  - ZAKR, 78
  - ZAKR.DO.CAŁK, 79

## funkcje

- agregujące, 215, 249
- nietypowe, 217
- aproksymacja, 555
- argumenty, 77
  - opcjonalne, 227, 229
  - puste, 227
- autouzupełniania formuł, 136
- baz danych, 444, 447
- bliźniacze, 225
- cyklometryczne, 357, 358
- do diagnozowania błędów obliczeń, 278
- do obliczania logarytmów, 78
- do odzyskiwania danych, 409
- do sumowania, 81
- do wykonywania działań na macierzach, 116
- do zaokrąglania liczb, 78
- dwu zmiennych, 304
- ekspansywne, 113, 223
- grupowanie, 213
- interpretacja
  - argumentów, 215
  - danych grupowych, 254
  - wyrażeń złożonych, 220
- inżynierskie, 223, 248
- jednej zmiennej, 299, 310
- kolejność danych, 219
- liczba zwracanych wyników, 218, 223
- logiczne, 85
- matematyczne, 75
- minimalna liczba argumentów, 216
- odwołania 3-W, 141, 221, 240
- parametryczne, 304
- podsumowań, 419
- pomijanie argumentów, 227
- redukujące, 223
- rodzaje argumentów, 214
- specjalne, 99, 222
- statystyczne, 227
- tablicowe, 218, 249, 299, 304
- translacyjne, 223
- trygonometryczne, 76, 357
- ulotne, 181
- ułatwiające konwersję typów, 234
- użycie argumentów innego typu, 236
- w polu obliczeniowym, 487
- wbudowane, 509–52
- wywołania, 71

- zagnieżdżanie, 71
- zliczania i sumowania warunkowego, 426
- zwracające odwołania, 224
- zwykle, 222, 248

**H**

- hiperłącza, 497
  - automatyczne tworzenie, 500
  - puste, 507
  - tworzenie, 498
  - usuwanie, 500
  - warunkowe, 507
- hiperłącze
  - do komórki, 503, 505
  - do określonego obszaru, 504
  - do określonej lokalizacji, 504
  - do programu pocztowego, 508
  - do skoroszytu, 503
  - do skoroszytu na innym dysku, 504
  - do skoroszytu na serwerze, 504
  - do zakresu arkusza, 504
- HTTP, Hypertext Transfer Protocol, 497

**I**

- iloczyn zbiorów, 469
- importowanie danych, 479
- informacje o formacie, 359
- inspekcja formuł, 264
- interpolacja, 310
- interpretacja
  - adresów, 247, 251
  - danych grupowych, 254
  - nazw, 247, 251

**J**

- język VBA, 24

**K**

- klawisz funkcyjny F9, 153
- kody
  - błędów, 278
  - formatu, 341, 356
  - daty i czasu, 355
  - komórek, 339
  - w funkcjach, 357

- kolejność
    - działań, 70
    - sortowania, 390
  - kolor tekstu, 342
  - kolumny, 31
    - dodawanie, 137
    - liczby w nagłówkach, 139
    - obliczeniowe, 132
    - pomocnicze, 290
    - przestawianie, 314
    - ukrywanie nagłówków, 137
    - usuwanie, 137, 174
    - wstawianie, 174
    - zmienianie nazwy, 137
  - komentarz, 562
  - komórki, 31, 562
    - informowanie o formacie, 359
    - o wylczonym adresie, 188
    - puste, 32, 287
      - zliczanie, 441
    - rodzaje formatowania, 340
    - scalone, 44, 58, 164
      - formuły, 167
      - formuły tablicowe, 169
    - z hiperłączem, 501
    - zależne, 293
  - konfiguracja listy rozwijanej, 261
  - koniunkcja kryteriów, 403
  - konsolidacja, 414
    - według kategorii, 417
    - według położenia, 414
  - kontrola
    - błędów, 278
    - poprawności wprowadzania danych, 257
  - konwersja
    - na tekst, 236
    - na wartość liczbową, 237
    - na wartość logiczną, 236
    - typów argumentów, 231
    - typów danych, 234
    - zakresu na tabelę, 140
  - kopiowanie
    - formuł, 157, 162, 167, 169
    - formuł tablicowych, 178
    - komórek widocznych, 170
    - odwołań, 162
    - za pomocą schowka, 171
  - Kreator kwerend, 477
  - kryteria, 428, 432, 482, 563
    - alternatywa, 403
    - koniunkcja, 403
    - określanie rekordów, 482
    - sortowania, 458
    - wyrażenia logiczne, 404
    - z wartościami błędów, 429
  - kwalifikator, 47
    - nazwy arkusza, 245
  - kwerendy, 477
    - miejsce docelowe wyników, 480
    - parametryczne, 481, 489
    - pobieranie tabel, 494
    - ponowne używanie, 479
    - udostępnianie, 479
- L**
- liczba, 37
    - całkowita, 239
  - liczbotekst, 563
  - licznik
    - generujący liczbę numerów, 452
    - oparty
      - na funkcji MAX, 451
      - na funkcji WIERSZ, 450
      - na odwołaniu względnym, 449
    - statyczny, 449
    - z filtrem, 453
  - lista
    - niestandardowa, 564
    - rozwijana, 564
      - aparatury fotograficznej, 275
      - dynamiczna, 265, 270
      - jako odpowiedź, 263
      - kody formatu, 349
      - konfiguracja, 261
      - tworzona ad hoc, 263
      - wynik wyboru, 274
      - z formatowaniem warunkowym, 275
      - z obsługą zdarzenia Change, 276
    - unikatów, 464–466
    - wyboru
      - z aktywną odpowiedzią, 271
    - zwracająca dane
      - liczbowe, 349
      - tekstowe, 350
    - zależna, 272

literał, 39

tablicowy, 106

logarytm, 78

Lotus, 29

## Ł

łącze, link, 46, 120, 564

łączenie

formuł i stałych, 169

fragmentów tablic, 189

tablic, 114, 203

## M

macierz, 105, 574

diagonalna, 106

jednostkowa, 121

kwadratowa, 106

odwrotna, 121

transponowana, 106

trójkątna, 122

macierze

wyznacznik, 121

maksimum warunkowe, 440

maskowanie wartości błędów, 281, 283

metody, 284

Menedżer

nazw, 57, 63

reguł formatowania warunkowego, 365, 372, 376

metoda Evaluate, 302

Microsoft Query, 475

definiowanie kwerendy, 477

działanie programu, 475

okienka kryteriów, 483

określanie zakresu wartości, 484

wyszukiwanie rekordów, 483

minimum warunkowe, 440

mnożenie macierzowe, 116

## N

nadawanie nazw, 54

nagłówki kolumn, 139

nawiasy

klamrowe, 107, 575

kwadratowe, 46

okrągłe, 72

nazwa, 564

a odwołania cykliczne, 60

automatyczna wymiana adresów, 62

formuł, 51, 59

komórek, 51

komórek scalonych, 58

kwalifikowane, 58

lokalne, 58

metody nadawania, 54

modyfikacja, 61

obszarów, 51

odczyt, 102

odwołań wielopoziomowych, 142

stałych, 51, 59

tabel, 133

tablic, 130

wyświetlanie listy, 63

z zaznaczenia, 55

zakres obowiązywania, 53

zakresów

a identyfikatory kolumn, 57

komórek, 53

wierszowych i kolumnowych, 55

zalety i wady, 63

nieprawidłowe dane, 264

numerowanie kolumn z danymi, 449

## O

obliczanie

logarytmów, 78

reszty z dzielenia, 79

wartości kryteriów, 485

obsługa

baz danych, 475

długich ciągów cyfr, 437

obszar, 41, 105, 565

przyległy, 566

roboczy, 566

odczyt

adresów, 88

gęstości roztworu, 330

kodu formatu, 356

nazwy, 102

odświeżanie danych zewnętrznych, 482

odwołania, 41, 564, 566

cykliczne, 49, 60, 566

do danych zewnętrznych, 46

do elementów tabeli, 138

- do elementów tablic, 185
- do fragmentów tablic, 185
- do komórki, 188
- do zamkniętych skoroszytów, 240
- mieszane, 95
- strukturalne, 131, 567
  - kopiowanie, 137
  - kwalifikowane, 135
  - niekwalifikowane, 135
  - przenoszenie, 137
  - używanie nazw, 139
  - wypełnianie, 137
  - zewnętrzne, 136
- wielopoziomowe, 3-W, 49, 140, 567
  - a zmiany w arkuszu, 143
  - nadawanie nazwy, 142
  - zastosowania, 143
- względne, 95, 449
  - zewnętrzne, 96
- odwracanie porządku elementów
  - w tablicy, 190
  - w zakresie, 195
- odzyskiwanie danych, 409
- okno inspekcji kwerendy, 480
- określanie zakresu wartości, 484
- OneDrive, 27
- opcje
  - autokorekty, 501
  - automatycznego formatowania, 501
  - formatowania warunkowego, 362, 364
- operacje
  - konsolidacji, 414
  - tablicowe, 110
  - transpozycji, 119
- operator, 567
  - Between, 489
  - In, 489
  - Is Nie Null, 489
  - Is Null, 488
  - Like, 488
- operatory
  - adresowe, 65
  - arytmetyczne, 231, 485
  - działań, 64
  - jednoargumentowe, 233
  - logiczne, 488
  - odwołania, 134
  - porównania, 485
  - relacji, 67

**P**

- parametr, 568
- paski danych, 367
- permutacja, 266
- pierwszeństwo wykonywania działań, 64
- plan eksperymentu czynnikowego, 200
- pobieranie
  - rekordów, 490
  - tabel, 494
- podzbiór, 582
- pole, 389, 573
  - kombi, 276
  - listy, 564
  - nazwy, 53
  - obliczeniowe, 486
    - funkcje, 487
- połączenie ze źródłem danych, 476
- poprawność
  - danych, 264
  - formuł, 292
- porównywanie
  - danych różnych typów, 69
  - tekstów, 68
  - zbiorów danych, 469
- priorytet operatorów, 64
- program
  - Lotus, 29
  - Microsoft Query, 475
- protokół
  - FTP, 497
  - HTTP, 497
- przeliczanie
  - arkusza, 179
  - automatyczne, 180
  - automatyczne z wyjątkiem tabel danych, 180
  - ręczne, 180
  - skoroszytu, 179
- przenoszenie
  - komórek, 173
  - powiązanych skoroszytów, 174
- przesstawianie wierszy, 314
- przeszukiwanie tablic, 310
- przyczyny błędów, 294
- przyspieszanie obliczeń, 292
- pseudofunkcja, 569
- pseudotekst, 569
- pseudozakres, 41, 569

## puste

- argumenty, 227, 228
- hiperłącza, 507
- komórki, 32, 228, 287
- komórki jako dane, 288
- teksty, 287
- usuwanie, 289

**R**

- ranking danych, 453
- regresja, 570
- reguły formatowania warunkowego, 385
- rekord, 389, 573
- rekurencja, 570
- relacje, 67
- reprezentacja tablic, 106
- rozmiary obszaru formuły, 206
- różnica zbiorów, 470

**S**

- samosprzężenia, 491
- scalanie komórek, 165
  - niestandardowe, 167
  - standardowe, 165
- schówek, 171
- sekwencje
  - językowe, 356
  - warunkowe, 343
- seria odwołań bezwzględnych, 161
- skala barw, 365
- skalar, 106
- składnia
  - tablicowa funkcji WYSZUKAJ, 318
  - wektorowa funkcji WYSZUKAJ, 315
- skoroszyt, 31, 494
  - wstawianie hiperłącza, 499
- skrót klawiaturowy
  - Ctrl+Alt+F9, 180
  - Ctrl+Alt+V, 120, 159
  - Ctrl+C, 170
  - Ctrl+D, 159
  - Ctrl+Enter, 159
  - Ctrl+F, 156
  - Ctrl+H, 156
  - Ctrl+Shift+Alt+F9, 180
  - Ctrl+Shift+Enter, 39, 108, 211
  - Shift+F9, 180

- sortowanie, 390, 392, 458
  - bazy danych, 390
  - zakresów
    - odwołania do komórek, 397
    - zawierających formuły, 396
- spacja, 73, 139, 346
- specjalne wyrażenie indeksowe, SWI, 572
- specyfikator
  - tabeli, 133
  - wierszy, 134
- sprzęganie tabel, 490
- sprzężenia
  - różnicowe, 493
    - wewnętrzne, 490
    - zewnętrzne, 490, 491, 494
    - pełne, 492
- stała, 572
  - liczbowa, 32, 36, 39
  - tablicowa, 106, 248, 575
  - tekstowa, 33, 39
  - logiczna, 33, 39
- stężenie procentowe, 330
- struktura
  - adresu URL, 497
  - arkusza kalkulacyjnego, 31
  - kodów formatu, 341
  - tabeli bazy danych, 389
  - tablicy, 210
  - wyrażeń, 209
- styl adresowania, 40, 506, 573
  - A1, 95, 506
  - W1K1, 91, 506
- suma
  - częściowa, 422
  - zbiorów, 471
- sumowanie, 81
  - warunkowe, 426, 438
- superpozycja, 112
- sygnalizacja błędów, 278
- symbole funkcji matematycznych, 76
- szacowanie formuł, 296

**Ś**

- śledzenie komórki, 103
- średnik, 114, 341, 561

**T**

tabele, 389, 573  
danych, 308, 448  
elementy składowe, 134  
formuły tablicowe, 139  
nazwa, 133  
odwołania strukturalne, 131  
pełne sprzężenia zewnętrzne, 492  
przestawne, 574  
samospzężenia, 491  
specyfikator, 133  
sprzężenia  
    różnicowe, 493  
    wewnętrzne, 491  
    zewnętrzne, 491, 494  
strukturalne, 131  
zmienianie nazwy, 137

tablica pierwiastków chemicznych, 393

tablice, 105, 574  
bazowe, 197, 576  
    definiowanie innych tablic, 199  
do obliczeń, TDO, 575  
do wyświetlania, TDW, 142, 213, 218, 575  
dynamiczne generowanie, 197  
dwuwymiarowe, 201  
jednowymiarowe, 197  
listowanie zawartości, 191  
łączenie, 203  
łączenie fragmentów, 189  
mieszane, 114  
nazwane, TN, 213, 576  
nazwy formuł, 248  
nieuporządkowane, 317  
odwracanie porządku elementów, 190  
pełnowartościowe, TPW, 204, 212, 576  
quasi-pełnowartościowe, TQP, 215, 576  
selektywne sumowanie elementów, 201  
struktura, 210  
wyszukiwanie danych, 336  
    liczbowych, 329  
    znaczących, 331  
z pamięcią adresu, TPA, 124, 576

tablicowanie funkcji  
    dwu zmiennych, 304  
    jednej zmiennej, 299

tekst, 33, 37  
    określenie koloru, 342

tekst adresu, 41

transpozycja, 117, 118, 120  
    danych zduplikowanych, 472

tworzenie  
    hiperłącza, 498  
    kwerendy parametrycznej, 489  
    list dynamicznych, 269  
    list zależnych, 272  
    listy unikatów, 464  
    nazw z zaznaczenia, 55  
    odwołań wielopoziomowych, 142  
    poła obliczeniowego, 486  
    serii odwołań, 161  
    tablic mieszanych, 114

typy  
    danych, 32, 577  
    informacji, 101, 104

**U**

uchwyt wypełniania, 158

układ dwuwymiarowy, 333

ukośnik wsteczny, 107

ukrywanie  
    nagłówków kolumn, 137  
    wartości zerowych, 285

ulotny, 577

ułamek, 239  
    złożony, 76

unikaty, 464, 465, 466

URL, Uniform Resource Locator, 497

usuwanie  
    błędnych danych, 290  
    formatów warunkowych, 384  
    hiperłącza, 500  
    kolumn, 174  
    komórek, 174  
    pustych tekstów, 289  
    wierszy, 174

użycie  
    adresów względnych i mieszanych, 242  
    autofiltera, 399  
    dynamicznej listy rozwijanej, 266, 268, 270  
    filtra zaawansowanego, 406  
    formuł tablicowych, 130, 139  
    formuł zamiast filtra, 407  
    klawisza F9, 153  
    listy dynamicznej, 266

## użycie

- nazw tablic, 130
- odwołań strukturalnych, 135, 136
- pasków danych, 367
- plusa jednoargumentowego, 233
- sekwencji językowych, 356
- skali barw, 365
- znaku spacji, 139

**V**

VBA, Visual Basic for Applications, 24, 578

**W**

## wartości

- błędów, 33
- domyślne, 227
- logiczne, 33, 231
- unikatowe, 459, 463
- zerowe, 285

## wektor, 105

## wersja online

- ograniczenia funkcjonalności, 27

## wiersz, 31

## wklejanie

- łącza, 120, 499
- specjalne, 159, 160

## wprowadzanie

- adresów do formuł, 42
- danych, 257
- funkcji do formuł, 71

## wskaźnik położenia, 454

## wstawianie

- kolumn, 174
- komórek, 174
- wierszy, 174

## wykres, 578

- przetawny, 579

## wykrzyknik, 46, 245, 246

## wyliczanie adresów, 88

## wyniki kwerendy, 480, 483

wyrażenia, 579, *Patrz także* formuła

- adresy, 250
- bazujące na wartościach odwołań, 247
- dynamiczne, 580
- interpretacja nazw, 247
- jednowartościowe, 209

## logiczne, 404

- błędów, 283

## nazwy, 250

## proste, 212, 580

## tablicowe, 123

- interpretacja nazw, 251

## wielowartościowe, 209

## z odwołaniami, 213

## z tablicami nazwanymi, 213

## złożone, 213, 580

## wyróżnianie wierszy, 383

## wyszukiwanie

## cyfr, 313

## danych, 312

- liczbowych, 329

## w tablicy, 336

## w układzie dwuwymiarowym, 333

## w zakresie, 334

## zduplikowanych, 326

## liczby, 319

## odpowiedników wartości, 322

## rekordów, 484

## wartości, 322

## wartości wspólnych, 469

## wszystkich wystąpień, 323

## znaczących danych, 331

## wyświetlanie

## danych, 147

## tekstów formuł, 148

## wywołania funkcji, 71

## wyznacznik macierzy, 106, 121

## wzorce formatu sekcji, 344

**Z**

## zagnieżdżanie funkcji, 71

## zakres, 41, 105, 580

- do wyświetlania, ZDW, 218, 575

## dynamiczny, 204, 580

- adres początku i końca, 205

## długość, 205

## początek obszaru, 205

## stały początek obszaru, 205

## formatowania, 381

- komórek, 365, 367, 369

## kontrolny, 432

## nieregularny, 219

## odwracanie porządku elementów, 195



- sumowania, 432
- złożony, 66
- znajdowanie wartości, 317, 334
- zakresy
  - część wspólna, 55, 196
  - wierszowe i kolumnowe, 55
- zamiana wyników formuł, 177
- zaokrąglanie liczb, 78
- zapis naukowy, 34
- zasada
  - rzutu prostokątnego, 44, 247, 581
  - sekwencji, 581
  - superpozycji, 112, 581
- zasieg, 581
- zatwierdzenie tablicowe, 211
- zaznaczanie
  - całego arkusza, 150
  - kolumn, 150
  - komórek, 149
  - nieprzylegających komórek, 149
  - ostatnich komórek arkusza, 151
  - wierszy, 150
  - zakresów komórek, 151
- zbiory, 582
- iloczyn, 469
- różnica, 470
- suma, 471
- zdarzenie Change, 276
- zliczanie
  - danych, 326
  - pustych komórek, 441
  - wartości unikatowych, 459
  - warunkowe, 426, 438
- zmienianie
  - czcionki, 182
  - nazwy tabeli, 137
  - zakresu formuły, 176
- zmienna, 582
- znajdowanie i zamiana, 156
- znak
  - #, 345
  - %, 346
  - \*, 346
  - „, 345, 346, 359
  - ., 345
  - /, 346
  - ?, 345
  - !, 46, 245, 246
  - @, 341, 347
  - ;, 114, 341, 561
  - 0, 345
  - C, 359
  - D, 359
  - E-, 346
  - F, 359
  - G, 359
  - H, 359
  - P, 359
  - S, 359
  - spacji, 73, 139, 346
- znaki wieloznaczne, 313

**Ż**

źródło danych, 476



# PROGRAM PARTNERSKI

— GRUPY HELION —

1. ZAREJESTRUJ SIĘ
2. PREZENTUJ KSIĄŻKI
3. ZBIERAJ PROWIZJĘ

Zmień swoją stronę WWW w działający bankomat!

**Dowiedz się więcej i dołącz już dzisiaj!**

<http://program-partnerski.helion.pl>

GRUPA  
**Helion** 

## Funkcje i wyrażenia Excela? Nic prostszego!

- Naucz się podstaw
- Poznaj funkcje
- Zrozum Excela

Na rynku nie brakuje książek opisujących obsługę i zastosowania arkusza kalkulacyjnego MS Excel, żadna jednak nie wprowadzi Cię w tę tematykę tak skutecznie, jak ta! Omiń rafa i białe plamy dokumentacji, skorzystaj z doświadczenia autora i śmiało wkrocz w świat niesamowitych możliwości Excela.

Dzięki tej książce poznasz podstawy działania arkusza kalkulacyjnego i najbardziej przydatne funkcje oferowane przez program. Nauczysz się obsługiwać tablicami i tabelami, sprawdzać poprawność danych i obliczeń oraz stosować różne sposoby formatowania. Dowiesz się także, jak wykorzystać Excela w roli systemu bazodanowego.

- Podstawowe możliwości arkusza
- Przydatne funkcje Excela
- Zastosowanie tablic i tabel
- Interpretacja wyrażen
- Kontrola poprawności danych i obliczeń
- Przeszukiwanie tablic
- Formatowanie komórek
- Funkcje bazodanowe
- Analiza danych w Excelu
- Zastosowanie hipertęczy

Dowiedz się, jak działa Excel i jak wykorzystać go w swojej pracy!

 <b>Helion</b>	Sprawdź nasze szkolenia!	<b>KOD KORZYŚCI</b> Sięgnij po więcej! ▶ 
 <b>helion.pl</b>	 <b>SZKOLENIA</b> AKADEMIA IT & BUSINESS	ISBN 978-83-283-5692-4  9 788328 356924
 <b>HELION SA</b> ul. Kościuszki 1c 44-100 Gliwice tel.: 32 230 98 63 helion@helion.pl	<a href="http://WWW.SZKOLENIA.HELION.PL">WWW.SZKOLENIA.HELION.PL</a>	Cena: 99,00 zł
<b>INFORMATYKA W NAJLEPSZYM WYDANIU</b>		