

Spis treści

1. Wprowadzenie	3
1.1. Baza danych	3
1.2. Tabele, rekordy, pola, klucze	3
1.3. Import tabeli	4
2. Funkcje bazy danych	7
2.1. Składnia funkcji bazy danych	7
2.2. BD.ILE.REKORDÓW()	8
2.3. BD.POLE()	9
2.4. BD.MAX()	10
2.5. BD.ŚREDNIA()	10
2.6. BD.ILE.REKORDÓW.A()	10
2.7. Pozostałe funkcje	11
2.8. Wyrażenia regularne	12
3. Filtrowanie danych	13
3.1. Autofiltr	13
3.2. Filtr standardowy	14
3.3. Filtr zaawansowany	15
3.4. Stypendia naukowe	16

Rozdział 1

Wprowadzenie

1.1. Baza danych

Baza danych to zbiór danych uporządkowanych w ściśle określony sposób, najczęściej w postaci *tabel* o stałej liczbie kolumn i dowolnej liczbie wierszy. W potocznym ujęciu pojęcie *baza danych* obejmuje także program komputerowy służący do gromadzenia i przetwarzania danych. Obecnie nie będziemy się zajmować tego typu programem, omówimy jedynie dostępne w arkuszu kalkulacyjnym Calc narzędzia służące do obsługi *tabel* z danymi.

1.2. Tabele, rekordy, pola, klucze

Tab. 1.1 przedstawia fragment przykładowej tabeli zawierającej informacje dotyczące studentów pewnej uczelni. Wiersze tabeli odpowiadają konkretnym studentom a kolumny są rodzajami danych, które w danej bazie są przechowywane. Studenci są opisani m.in. przez imię, nazwisko, płeć, numer albumu czy datę urodzenia, przy czym w przypadku każdego studenta są to informacje tego samego typu i w tej samej liczbie (każdy wiersz omawianej tabeli ma dokładnie tyle samo kolumn). Aby w kolejnych przykładach łatwiej było się odwoływać do tej tabeli przyjmijmy, że nosi ona nazwę **STUDENCI**.

Tabela 1.1. Fragment tabeli **STUDENCI**

Imię	Nazwisko	Płeć	Album	RokUrodzenia	...
WANDA	WRÓBEL	K	12194	1983	...
IWONA	ZAJĄC	K	12195	1987	...
ALEKSANDRA	SIKORA	K	12196	1990	...
WIESŁAWA	SZYMAŃSKA	K	12197	1983	...
RENATA	WÓJCIK	K	12198	1985	...
IWONA	MALINOWSKA	K	12199	1990	...
...

Pojedynczy wiersz tabeli zawierający wszystkie informacje dotyczące pojedynczego studenta nazywamy *rekordem*. Kolumny tabeli, odpowiadające rodzajom przechowywanej w bazie danych informacji, noszą nazwę *pól*. Tabela **STUDENCI** zawiera więc m.in. następujące pola: **Imię**, **Nazwisko**, **Płeć**, **Album**, **RokUrodzenia**.

W każdej tabeli jedno z pól powinno zawierać dane unikalne dla każdego rekordu. Warunku tego w przypadku tabeli **STUDENCI** nie spełnia **Imię** ani **Nazwisko** (zarówno imiona jak i nazwiska mogą się powtarzać) czy **RokUrodzenia**. Unikalne są jednak wartości w polu **Album** — tego typu pole może być wykorzystane do jednoznacznego identyfikowania rekordów, pełni ono wówczas rolę **klucza głównego** tabeli. Podczas pracy z arkuszem kalkulacyjnym nie będziemy wykorzystywać kluczy a tabele, którymi będziemy się zajmować, nie będą musiały mieć żadnego pola o unikalnych wartościach.

1.3. Import tabeli

Bardzo często pierwszym krokiem podczas pracy z dużymi tabelami danych jest ich import do arkusza. Jeżeli tabela jest niewielka można ją wprowadzić do arkusza ręcznie, najczęściej nie ma jednak wówczas konieczności wykorzystywania funkcji baz danych do pracy z taką tabelą. W przypadku dużych tabel ręczne wprowadzenie danych jest praktycznie niemożliwe, jedyną możliwością jest importowanie tabel.

Tabele danych przygotowane do importu są najczęściej zapisane w postaci tekstowych plików, w których kolejne wiersze odpowiadają kolejnym rekordom tabeli zaś poszczególne pola są oddzielone *separatorami* (najczęściej stosuje się przecinki, średniki, spacje lub znaki tabulacji), pliki zapisane w tym formacie nazywamy plikami CSV (ang: Comma Separated Values). Fragment przykładowego pliku CSV jest przedstawiony na Rys. 1.1.

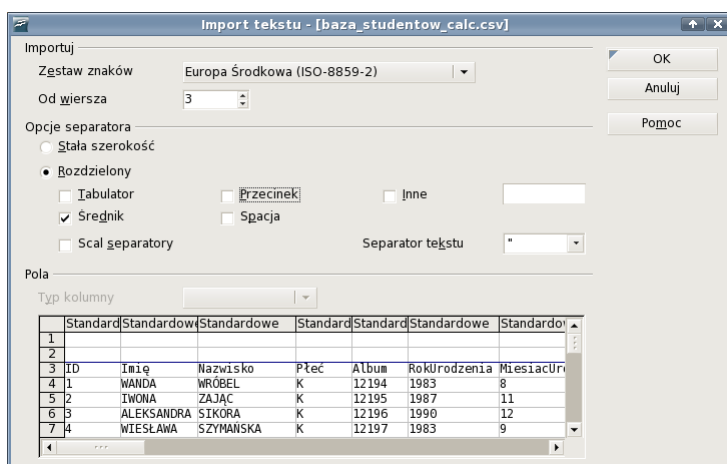
```
ID;Imię;Nazwisko;Płeć;Album;RokUrodzenia;MiesiacUrodzenia;DzienUrodzenia;PESEL;Wydział;Kierunek;RokStudi
1;WANDA;WRÓBEL;K;12194;1983;8;15;83081515967;WEII;elektrotechnika;3;4;0;0;0;3;0;0;0;3;5;0;0;5;0;0;
2;IWONA;ZAJĄC;K;12195;1987;11;16;87111692799;WBIIŚ;inżynieria środowiska;3;3;5;0;0;4;5;0;0;4;0;0;5;0;0;
3;ALEKSANDRA;SIKORA;K;12196;1990;12;5;90120515955;WEII;informatyka;1;4;0;0;0;3;5;0;0;2;0;3;0;2;5;5;0;3;5;
4;WIESŁAWA;SZYMAŃSKA;K;12197;1983;9;17;83091740864;WZIM;zarządzanie;5;4;5;0;0;4;0;0;0;3;0;0;0;4;5;0;0;2;0;
5;RENATA;WÓJCIK;K;12198;1985;12;24;85122451183;WBIIŚ;budownictwo;4;2;0;2;0;3;0;0;0;4;5;0;0;4;5;0;0;3;5;0;
6;IWONA;MALINOWSKA;K;12199;1990;2;15;90021594817;WBIIŚ;budownictwo;4;2;5;2;0;5;0;0;0;3;0;0;0;2;5;2;5;4;0;
7;DOROTA;KRAWCZYK;K;12200;1981;1;13;81011324357;WBIIŚ;budownictwo;5;2;5;4;5;2;0;2;0;3;5;0;0;3;0;0;0;2;0;
8;WIESŁAWA;WOŹNIAK;K;12201;1982;11;15;82111516893;WEII;telekomunikacja;4;2;0;3;0;4;5;0;0;4;5;0;0;3;0;0;0;
9;ANETA;JANKOWSKA;K;12202;1987;10;22;87102242437;WEII;informatyka;3;2;0;4;5;2;0;3;5;4;5;0;0;2;5;3;0;2;0;
10;KAROLINA;KOZŁOWSKA;K;12203;1986;1;13;86011327286;WBMIŁ;lotnictwo i kosmonautyka;3;2;5;2;5;2;0;2;5;2;5;2;5;
```

Rys. 1.1. Przykładowa zawartość pliku w formacie CSV

Aby zaimportować dane zapisane w pliku CSV należy taki plik otworzyć w arkuszu kalkulacyjnym (**MC Plik>Otwórz**). Automatycznie zostanie uruchomione narzędzie importu tekstu (patrz Rys. 1.2), w którym należy ustawić prawidłowe opcje importu i wcisnąć **OK**. Dla pliku przedstawionego na Rys. 1.1 prawidłowe opcje importu są pokazane na Rys. 1.2:

- strona kodowa (format polskich znaków narodowych): ISO-8859-2,
- import od wiersza trzeciego (dwa pierwsze są puste co widać na podglądzie pliku),
- pola są rozdzielone średnikami (koniecznie należy odznaczyć przecinki, które w przykładowym pliku są separatorami części całkowitej od dziesiętnej w liczbach).

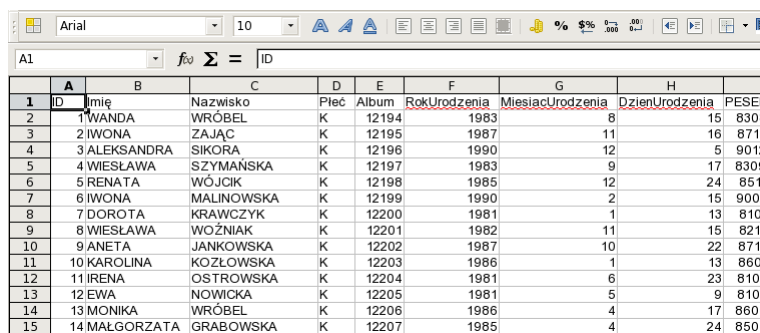
W oknie podglądu w dolnej części okna importu tekstu widoczne są dane w postaci, która będzie dostępna po zakończeniu importu. Przed zaakceptowaniem ustawień i zakończeniem importu należy sprawdzić w oknie podglądu, czy dane będą prawidłowo zaimportowane do arkusza. Rys. 1.3 przedstawia fragment tabeli **STUDENCI** zaimportowanej do arkusza kalkulacyjnego. Plik z tabelą **STUDENCI** jest udostępniony



Rys. 1.2. Import pliku w formacie CSV

pod adresem:

http://www.prz.edu.pl/~bmiller/dydaktyka/ti/tabela_studenci.csv



	A	B	C	D	E	F	G	H	
1	ID	Imię	Nazwisko	Płeć	Album	RokUrodzenia	MiesiacUrodzenia	DzieńUrodzenia	PESEL
2	1	WANDA	WRÓBEL	K	12194	1983	8	15	8301
3	2	IWONA	ZAJĄC	K	12195	1987	11	16	871
4	3	ALEKSANDRA	SIKORA	K	12196	1990	12	5	901
5	4	WIESŁAWA	SZYMAŃSKA	K	12197	1983	9	17	8301
6	5	RENATA	WÓJCIK	K	12198	1985	12	24	851
7	6	IWONA	MALINOWSKA	K	12199	1990	2	15	900
8	7	DOROTA	KRAWCZYK	K	12200	1981	1	13	810
9	8	WIESŁAWA	WOŹNIAK	K	12201	1982	11	15	821
10	9	ANETA	JANKOWSKA	K	12202	1987	10	22	871
11	10	KAROLINA	KOZŁOWSKA	K	12203	1986	1	13	860
12	11	IRENA	OSTROWSKA	K	12204	1981	6	23	810
13	12	EWA	NOWICKA	K	12205	1981	5	9	810
14	13	MONIKA	WRÓBEL	K	12206	1986	4	17	860
15	14	MAŁGORZATA	GRABOWSKA	K	12207	1985	4	24	850

Rys. 1.3. Tabela danych zaimportowana do arkusza kalkulacyjnego

Rozdział 2

Funkcje bazy danych

2.1. Składnia funkcji bazy danych

Wszystkie funkcje bazy danych w arkuszu kalkulacyjnym Calc mają trzy argumenty:

- **baza_danych**: obszar arkusza, w którym znajduje się tabela z danymi, na których będą wykonywane operacje wyszukiwania, filtrowania czy zliczania,
- **pole_bazy_danych**: pole tabeli, na którym będzie działała funkcja po zastosowaniu kryteriów wyszukiwania,
- **kryteria_wyszukiwania**: warunki, które muszą spełnić rekordy brane pod uwagę podczas wyszukiwania, filtrowania czy zliczania.

Poniżej zostaną szczegółowo omówione następujące funkcje:

- **BD.ILE.REKORDÓW(baza_danych; pole_bazy_danych; kryteria_wyszukiwania)**
funkcja zwraca liczbę rekordów spełniających określone kryteria wyszukiwania.
- **BD.POLE(baza_danych; pole_bazy_danych; kryteria_wyszukiwania)**
funkcja zwraca wartość wskazanego pola w rekordzie spełniającym kryteria wyszukiwania. Jeżeli nie odnaleziono rekordu spełniającego kryteria funkcja wyświetla błąd **#WARTOŚĆ!**, jeżeli funkcja znajdzie więcej niż jedną komórkę wyświetla **Błąd 502**.
- **BD.MAX(baza_danych; pole_bazy_danych; kryteria_wyszukiwania)**
funkcja zwraca największą wartość we wskazanym polu, podczas wyszukiwania brane są pod uwagę wyłącznie rekordy spełniające kryteria wyszukiwania.
- **BD.ŚREDNIA(baza_danych; pole_bazy_danych; kryteria_wyszukiwania)**
funkcja oblicza średnią wartość we wskazanym polu we wszystkich rekordach spełniających kryteria wyszukiwania.

Ponieważ tabela zaimportowana w poprzednim rozdziale zawiera aż 600 rekordów oraz 24 pola podczas szczegółowego omawiania wymienionych funkcji będziemy pracować z mniejszą tabelą przedstawioną na Rys. 2.1. Tabela przedstawia wyniki zawodów szermierczych, odnotowywano wygrane, porażki i remisy kolejnych zawodników, a następnie policzono ich skuteczność zdefiniowaną jako stosunek wygranych do liczby pojedynków. Tabela jest posortowana alfabetycznie według nazwisk zawodników.

Podczas pracy z tą tabelą pierwszy argument każdej z omawianych funkcji bazy danych będzie przyjmował wartość **A1:E10** (należy koniecznie podać zakres komórek obejmujący pierwszy wiersz tabeli z nazwami pól). Jeżeli na tej samej tabeli będzie wykonywanych wiele operacji to zalecane jest nadanie obszarowi komórek **A1:E10** nazwy, tak aby można było się do niego łatwiej odwoływać. Aby nadać nazwę obszarowi **A1:E10** należy go zaznaczyć, wybrać **MG Wstaw>Nazwy>Definiuj...**, wprowadzić nazwę i zatwierdzić wciskając **OK**. Przyjmijmy, że obszarowi **A1:E10** nadano nazwę **WYNIKI**, od tej chwili jako pierwszy argument funkcji bazy danych możemy podawać zarówno zakres **A1:E10** jak i jego nazwę **WYNIKI**.

	A	B	C	D	E
1	Nazwisko	Wygrane	Porażki	Remisy	Skuteczność
2	Babinicz	5	3	0	62,5
3	Bohun	5	2	1	62,5
4	Jurand	8	0	0	100,0
5	Ketling	1	6	1	12,5
6	Kmicic	7	0	1	87,5
7	Kowalski	2	4	2	25,0
8	Maćko	6	2	0	75,0
9	Skrzetuski	6	2	0	75,0
10	Zagłoba	3	4	1	37,5

Rys. 2.1. Tabela WYNIKI w arkuszu

2.2. BD.ILE.REKORDÓW()

Zadanie: ilu zawodników wygrało więcej niż 5 razy?

Najpierw zbudujemy wiersze zawierające kryteria wyszukiwania. Kryteria zawsze zawierają nazwy pól analizowanej tabeli, więc możemy je skopiować pod tabelę: zaznaczamy komórki **A1:E1** i kopiujemy je do wiersza **12**. W komórce **B13** wpisujemy warunek logiczny **>5** (mamy znaleźć liczbę zawodników, którzy wygrali więcej niż 5 pojedynków, więc zakładamy warunek wyłącznie na pole **Wygrane**). Ostatecznie kryteria wyszukiwania są zdefiniowane w komórkach **A12:E13**.

W komórce **A15** wpisujemy tekst **ilu zawodników wygrało więcej niż 5 razy?**, zaś w **D15** formułę dającą odpowiedź na postawione pytanie: **D15 =BD.ILE.REKORDÓW(WYNIKI;0;A12:E13)**. Funkcja zliczy w tabeli **WYNIKI** wszystkie rekordy spełniające kryteria zdefiniowane w **A12:E13**, ponieważ pod uwagę ma być brana cała tabela a nie wybrane pole drugi argument funkcji przyjmie wartość **0**. Formuła może być także zapisana w postaci: **D15 =BD.ILE.REKORDÓW(A1:E10;0;A12:E13)**, tak jak na Rys. 2.2.

	A	B	C	D	E	F
1	Nazwisko	Wygrane	Porażki	Remisy	Skuteczność	
2	Babinicz	5	3	0	62,5	
3	Bohun	5	2	1	62,5	
4	Jurand	8	0	0	100,0	
5	Ketling	1	6	1	12,5	
6	Kmicic	7	0	1	87,5	
7	Kowalski	2	4	2	25,0	
8	Maćko	6	2	0	75,0	
9	Skrzetuski	6	2	0	75,0	
10	Zagłoba	3	4	1	37,5	
11						
12	Nazwisko	Wygrane	Porażki	Remisy	Skuteczność	
13		>5				
14						
15	ilu zawodników wygrało więcej niż 5 razy?	=BD.ILE.REKORDÓW(A1:E10;0;A12:E13)				

Rys. 2.2. Zliczanie zawodników, którzy mieli więcej niż 5 zwycięstw

Zadanie 1: ilu zawodników wygrało wszystkie swoje walki?

Tym razem kryteria wyszukiwania są zdefiniowane w komórkach **A13:E14** (patrz Rys. 2.3), w **C14** wpisujemy **0**, co oznacza, że poszukujemy zawodników z zerową liczbą porażek. Formuła rozwiązująca zadanie jest następująca: **B12 =BD.ILE.REKORDÓW(WYNIKI;0;A13:E14)**.

LICZ.JEŻELI() W arkuszu Calc dostępna jest także funkcja **LICZ.JEŻELI(zakres;kryteria)** pozwalająca na zliczenie rekordów (a dokładniej komórek — funkcja **LICZ.JEŻELI()** nie operuje na tabeli a wyłącznie na zakresie komórek) bez definiowania kryteriów w wybranych komórkach arkusza,

	A	B	C	D	E
11					
12	Zadanie 1.	2			
13	Nazwisko	Wygrane	Porażki	Remisy	Skuteczność
14			0		
15					
16	Zadanie 2.	4			
17	Nazwisko	Wygrane	Porażki	Remisy	Skuteczność
18			>0		>50
19					
20	Zadanie 3.	5			
21	Nazwisko	Wygrane	Porażki	Remisy	Skuteczność
22			0		
23				0	
24					
25	Zadanie 4.	3			
26	Nazwisko	Wygrane	Porażki	Remisy	Skuteczność
27	k.*				
28					
29	Zadanie 5.	7			
30	Nazwisko	Wygrane	Porażki	Remisy	Skuteczność
31	Kmicic				
32					
33	Zadanie 6.	75			
34	Nazwisko	Wygrane	Porażki	Remisy	Skuteczność
35		6			
36					
37	Zadanie 7.	4			
38	Nazwisko	Wygrane	Porażki	Remisy	Skuteczność
39				1	

Rys. 2.3. Przykłady zastosowania funkcji `BD.ILE.REKORDÓW`

mogą one być wprowadzone jako argument funkcji. Dla przykładu aby rozwiązać pierwsze zadanie można posłużyć się formułą `=LICZ.JEŻELI(C2:C10;"=0")`.

Zadanie 2: ilu zawodników miało porażki, a mimo to ich skuteczność była większa niż 50%?

Kryteria wyszukiwania muszą zawierać dwa warunki: `>0` w polu *Porażka* (zawodnicy, którzy doznali przynajmniej jednej porażki) i `>50` w polu *Skuteczność*. Dopiero połączenie tych dwóch warunków da nam wymagane kryterium wyszukiwania. Rozwiązanie: `B16 =BD.ILE.REKORDÓW(WYNIKI;0;A17:E18)`

Zadanie 3: ilu zawodników nie doznało porażki lub nie miało remisu?

W tym zadaniu będziemy musieli wykorzystać operator logiczny `LUB`. W `C22` wpisujemy `0`, gdyż chcemy wyszukiwać zawodników bez porażki. Gdybyśmy teraz do komórki `D22` wpisali `0`, to wynikiem byłaby liczba zawodników bez porażki i bez remisów (warunki byłyby połączone operatorem `I` ponieważ byłyby zdefiniowane w tym samym wierszu). My poszukujemy zawodników, którzy nie doznali porażki lub nie mieli remisu, więc drugi warunek musimy wpisać w nowym wierszu: `D23 0`. Rozwiązanie: `B20 =BD.ILE.REKORDÓW(WYNIKI;0;A21:E23)`.

Zadanie 4: ilu zawodników z listy ma nazwiska zaczynające się na literę *k*?

W `A27` wpisujemy warunek w postaci *wyrażenia regularnego* `k.*` (krotki opis wyrażeń regularnych znajduje się na stronie 12). Rozwiązanie: `B25 =BD.ILE.REKORDÓW(WYNIKI;0;A26:E27)`.

2.3. BD.POLE()

Zadania 5: ile wygranych miał Kmicic?

Pytanie wydaje się banalne, nasza tabela zawiera bowiem tylko kilka rekordów, odpowiedź nie będzie jednak tak prosta jeżeli tabela będzie zawierać znacznie więcej rekordów. Aby rozwiązać to zadanie korzystając z pomocy arkusza kalkulacyjnego definiujemy kryteria wyszukiwania w obszarze **A30:E31** (patrz Rys. 2.3). Analizowanym polem tabeli jest to, którego nagłówek znajduje się w **B1** (czyli **Wygrane** — interesuje nas liczba wygranych), dane znajdują się w tabeli **WYNIKI**. Rozwiązaniem jest formuła: **B29** =**BD.POLE(WYNIKI;B1;A30:E31)**. Formuła w komórce **B29** poda nam liczbę wygranych tego zawodnika, którego nazwisko jest wprowadzone w **A31**, można więc zmienić podane nazwisko na inne i otrzymamy liczbę zwycięstw innego zawodnika.

Jeżeli w formule **B29** =**BD.POLE(WYNIKI;B1;A30:E31)** zmienimy **B1** na **C1** uzyskamy informację na temat porażek. Podając jako drugi argument funkcji **BD.POLE()** adresy komórek zawierających nagłówki kolejnych pól tabeli (**B1**, **C1**, **D1** lub **E1**) możemy odczytać wszystkie informacje o danym uczestniku.

Ponieważ pola tabeli można także identyfikować przez ich nazwy zamiast pisać **B1** można wpisać **"Wygrane"** (cudzysłów jest obowiązkowy): **B29** =**BD.POLE(WYNIKI;"Wygrane";A30:E31)**.

Proszę w ramach ćwiczenia wpisać w **A31** także inne nazwiska z listy.

WYSZUKAJ.PIONOWO() Podobne zastosowanie ma funkcja **WYSZUKAJ.PIONOWO()**, która w **Macierzy** podanej jako argument odszukuje wartość znajdującą się w kolumnie o numerze **Indeks** i w tym samym wierszu, w którym w pierwszej kolumnie znajduje się wartość opisana przez **Kryteria_wyszukiwania**:

WYSZUKAJ.PIONOWO(Kryteria_wyszukiwania;Macierz;Indeks)

Jeżeli wartości w pierwszej kolumnie nie są posortowane rosnąco to dodatkowo trzeba w wywołaniu funkcji uwzględnić czwarty parametr o wartości **0** (zero). Przykładowa formuła pozwalająca odczytać liczbę wygranych (druga kolumna tablicy) zawodnika o nazwisku **Kmicic** (nazwisko musi być podane w pierwszej kolumnie analizowanej tablicy) jest następująca:

=**WYSZUKAJ.PIONOWO("Kmicic";WYNIKI;2)**

Jeżeli pierwsza kolumna nie jest posortowana a wartość podana jako **Kryteria_wyszukiwania** nie zostanie odnaleziona to funkcja zwróci błąd, jeżeli pierwsza kolumna jest posortowana to funkcja odwoła się do najbliższej wartości z tej kolumny (ta sytuacja jest często przyczyną pomyłek: jeżeli jako **Kryteria_wyszukiwania** zostanie wprowadzona wartość większa od wszystkich wartości z pierwszej kolumny to funkcja nie zwróci błędu a odwoła się do ostatniego wiersza macierzy).

Analogicznie działa funkcja **WYSZUKAJ.POZIOMO()**, z tym, że wyszukiuje wartości w wierszach.

2.4. BD.MAX()

Zadanie 6: jaka była największa skuteczność przy określonej liczbie wygranych?

Rozwiązanie: **B33** =**BD.MAX(WYNIKI;E1;A34:E35)**

Dla przykładowej liczby **6** wpisanej jako kryterium pojawił się wynik **75** (skuteczność **75%**). Tak jak w zadaniu poprzednim możemy adres **E1** zastąpić nazwą pola: **"Skuteczność"**.

2.5. BD.ŚREDNIA()

Zadanie 7: jaka jest średnia wygranych dla zawodników z jednym remisem?

Kryteria wyszukiwania są zdefiniowane w **A38:E39**, analizowanym polem tabeli będzie **"Wygrane"** (mamy obliczyć średnią liczbę zwycięstw). Rozwiązanie: **B37** =**BD.ŚREDNIA(WYNIKI;"Wygrane";A38:E39)**

2.6. BD.ILE.REKORDÓW.A()

Funkcja **BD.ILE.REKORDÓW.A()** ma identyczną składnię jak omówiona już funkcja **BD.ILE.REKORDÓW()** oraz bardzo podobne zastosowanie. Funkcje różnią się tym, że po określeniu konkretnego pola bazy danych do przeszukiwania funkcja **BD.ILE.REKORDÓW()** będzie zliczać wyłącznie niepuste rekordy zawierające w danym polu liczbę zaś funkcja **BD.ILE.REKORDÓW.A()**

będzie zliczać wszystkie niepuste komórki. Działanie tej funkcji zostanie wyjaśnione na przykładzie tabeli `STUDENCI_MALA` (patrz Tab. 2.1).

Tabela 2.1. Tabela `STUDENCI_MALA` z brakującymi danymi w polu `RokUrodzenia`

Imię	Nazwisko	Płeć	RokUrodzenia	RokStudiów	Matematyka
WANDA	WRÓBEL	K	1983	3	4
IWONA	ZAJĄC	K	1987	3	3,5
ALEKSANDRA	SIKORA	K		1	4
WIESŁAWA	SZYMAŃSKA	K	???	5	4,5
RENATA	WÓJCIK	K	1985	4	2
IWONA	MALINOWSKA	K	1990	4	2,5
DOROTA	KRAWCZYK	K	1981	5	2,5
WIESŁAWA	WOŹNIAK	K	1982	5	2,5
ANETA	JANKOWSKA	K	1987	4	2
KAROLINA	KOZŁOWSKA	K	1989	3	2

Tabela 2.2. Kryteria wyszukiwania

	A	B	C
603			
604		Matematyka	
605		≥ 3	
606			

W przypadku tabeli danych przedstawionej w Tabeli 2.1 oraz kryteriów wyszukiwania przedstawionych w Tabeli 2.2 formuły `=BD.ILE.REKORDÓW(STUDENCI_MALA;"RokUrodzenia";B604:B605)` oraz `=BD.ILE.REKORDÓW.A(STUDENCI_MALA;"RokUrodzenia";B604:B605)` zwrócą różne wartości. Pierwsza (wykorzystująca funkcję `BD.ILE.REKORDÓW()`) zwróci liczbę osób z oceną z Matematyki wyższą lub równą 3 (dst), dla których jest znany rok urodzenia — w tabeli są dwie takie osoby (Wanda Wróbel oraz Iwona Zajac), formuła zwróci wartość 2. Druga formuła (z funkcją `BD.ILE.REKORDÓW.A()`) zliczy osoby z oceną 3 (dst) lub wyższą, które mają w polu `RokUrodzenia` dowolną wartość — poza Wandą Wróbel i Iwoną Zajac zostanie do tej grupy zaliczona także Wiesława Szymańska, formuła zwróci wartość 3. Należy zwrócić uwagę, że żadna z formuł nie będzie brać pod uwagę Aleksandy Sikory (puste pole `RokUrodzenia`).

W przypadku, kiedy żadne pole tabeli nie będzie w formule podane wprost, obie funkcje będą działać identycznie. Formuły `=BD.ILE.REKORDÓW(STUDENCI_MALA;0;B604:B605)` oraz `=BD.ILE.REKORDÓW.A(STUDENCI_MALA;0;B604:B605)` zwrócą wartość 4 (zostaną zliczone wszystkie rekordy bez względu na zawartość pola `RokUrodzenia`).

2.7. Pozostałe funkcje

Arkusz kalkulacyjny Calc ma zdefiniowanych 12 funkcji bazy danych, każda funkcja ma identyczną składnię (parametrami są zawsze `baza_danych`, `pole_bazy_danych` oraz `kryteria_wyszukiwania`). Poza już omówionymi w arkuszu są jeszcze dostępne następujące funkcje:

- `BD.MIN()` — najmniejsza wartość we wskazanym polu,
- `BD.ILOCZYN` — iloczyn wartości ze wskazanego pola,
- `BD.ODCH.STANDARD` — odchylenie standardowe wartości ze wskazanego pola,
- `BD.ODCH.STANDARD.POPUL` — odchylenie standardowe wartości ze wskazanego pola,

- **BD.SUMA** — suma wartości ze wskazanego pola,
- **BD.WARIANCJA** — wariancja wartości ze wskazanego pola,
- **BD.WARIANCJA.POPUL** — wariancja wartości ze wskazanego pola.

Opis wszystkich wymienionych funkcji jest dostępny w pomocy programu Calc.

2.8. Wyrażenia regularne

W wielu przypadkach pojawia się konieczność takiego zdefiniowania kryteriów wyszukiwania, aby można było wykonać operacje na rekordach, które nie mają ani jednego pola o takich samych wartościach. Przykładem może być sytuacja, kiedy chcemy policzyć studentów z nazwiskiem rozpoczynającym się na literę **A**. Aby rozwiązać to zadanie nie możemy w opisany powyżej sposób skonstruować kryteriów wyszukiwania, nie ma bowiem żadnej wspólnej wartości w żadnym z pól tabeli. W takiej sytuacji należy wykorzystać *wyrażenia regularne*, służące do zdefiniowania wzorca opisującego całą grupę ciągów znakowych. Wyrażenia regularne są bardzo często wykorzystywane np. podczas obróbki tekstu, jest to tak szerokie zagadnienie, że doczekało się wielu publikacji książkowych w całości mu poświęconych. W tym opracowaniu ograniczymy się wyłącznie do podania kilku przykładów, szczegółowy opis wyrażeń regularnych osoby zainteresowane odnajdą w pomocy pakietu OpenOffice.org. Przykłady:

- **.a** — kropka zastępuje dowolny pojedynczy znak, po którym musi wystąpić litera **a**, pasujące słowa to np. **na**, **za**, **ba**,
- **.a.** — dowolny pojedynczy znak, litera **a**, dowolny pojedynczy znak: **nad**, **zaś**, **bal**,
- **.a*** — dowolny pojedynczy znak, litera **a** występująca dowolną ilość razy (gwiazdka pozwala na zwielokrotnienie znaku, który stoi bezpośrednio przed nią): **na**, **zaa**, **baaaaa** oraz **w** (dowolna liczba wystąpień litery **a** obejmuje także jej brak),
- **a.*** — dowolny ciąg znaków rozpoczynający się literą **a**: **a**, **ala**, **arka**, **akuszerka**,
- **[A-C].*** — dowolny ciąg znaków rozpoczynający się wielką literą **A**, **B** lub **C**,
- **[^A-C].*** — dowolny ciąg znaków rozpoczynający się dowolnym znakiem poza wielką literą **A**, **B** lub **C**.

Aby rozwiązać wcześniej postawione zadanie — zliczenie osób o nazwiskach rozpoczynających się na literę **A** — należy zastosować znaną już funkcję **BD.ILE.REKORDÓW()**, dla której kryteria wyszukiwania będą zdefiniowane w sposób przedstawiony w Tab. 2.3.

Tabela 2.3. Kryteria wyszukiwania z zastosowaniem wyrażeń regularnych

	A	B	C
603			
604		Nazwisko	
605		A.*	
606			

Rozdział 3

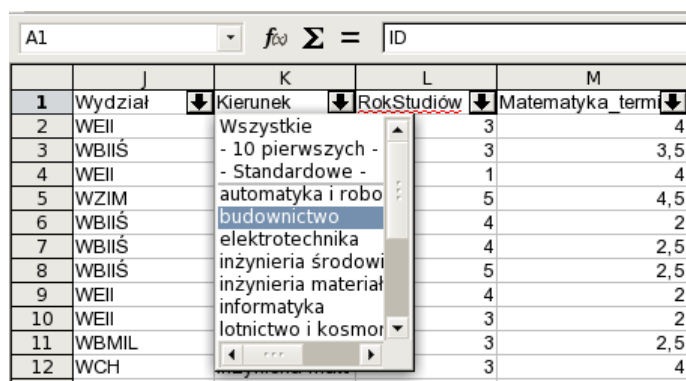
Filtrowanie danych

W wielu przypadkach zachodzi konieczność uzyskania tabeli zawierającej wyłącznie wybrane rekordy z oryginalnej tabeli, przykładem może być lista studentów 1. roku kierunku budownictwo na WBiŚ. Aby przygotować taką listę w arkuszu kalkulacyjnym można wykorzystać jeden z trzech dostępnych *filtrów*: autofiltr, filtr standardowy oraz filtr zaawansowany.

3.1. Autofiltr

Autofiltr to najprostszy z filtrów, który pozwala na bardzo ograniczone filtrowanie danych. Aby go uruchomić należy uaktywnić dowolną komórkę wewnątrz tabeli z danymi i wywołać funkcję **MC Dane>Filtr>Autofiltr** (ponowne wybranie tej samej funkcji wyłącza autofiltr). W nagłówku tabeli, obok nazwy każdego pola, pojawiają się strzałki pozwalające na zdefiniowanie kryteriów filtrowania, po ich zdefiniowaniu w tabeli pozostaną wyświetlane wyłącznie rekordy pasujące do tych kryteriów. Aby wyświetlić listę studentów 1. roku kierunku budownictwo na WBiŚ należy klikając w strzałki w nagłówku tabeli rozwinąć listy możliwych wartości w polach **RokStudiów** i **Kierunek** i wybrać odpowiednie wartości (patrz Rys. 3.1). Należy zwrócić uwagę, że w przypadku tabeli **STUDENCI** nie ma potrzeby zakładania filtra na pole **Wydział**, ponieważ kierunek budownictwo jest wyłącznie na WBiŚ.

Po włączeniu filtrowania po danym polu strzałka w nagłówku zmieni kolor na niebieski a rekordy nie pasujące do zdefiniowanych kryteriów zostaną ukryte (po wyłączeniu filtrowania zostaną ponownie wyświetlone).



	J	K	L	M
	Wydział	Kierunek	RokStudiów	Matematyka_termi
1	WEI	Wszystkie	3	4
2	WEI	- 10 pierwszych -	3	3,5
3	WEI	- Standardowe -	1	4
4	WEI	automatyka i robo	5	4,5
5	WEI	budownictwo	4	2
6	WEI	elektrotechnika	4	2,5
7	WEI	inżynieria środowi	5	2,5
8	WEI	inżynieria materiał	4	2
9	WEI	informatyka	3	2
10	WEI	lotnictwo i kosmor	3	2,5
11	WCH		3	4
12	WCH		3	4

Rys. 3.1. Autofiltr

Autofiltr pozwala zdefiniować kryteria filtrowania wyłącznie przez przyrównanie wybranych pól do wartości, które już w danym polu występują. Jeżeli to nie wystarczy, należy zastosować filtr standardowy lub zaawansowany.

3.2. Filtr standardowy

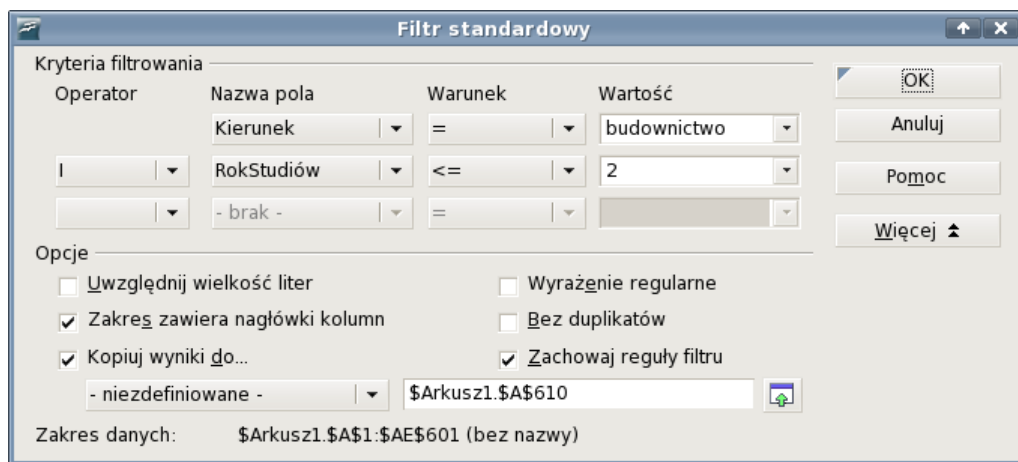
Filtr standardowy można uaktywnić wybierając opcję **-Standardowe-** z listy autofiltru lub, po uaktywnieniu dowolnej komórki wewnątrz tabeli z danymi, przez **MG Dane>Filtr>Filtr standardowy**. W obu przypadkach na ekranie pojawi się okno pozwalające zdefiniować kryteria filtrowania. Na Rys. 3.2 przedstawione są gotowe kryteria pozwalające ponownie uzyskać listę studentów 1. roku budownictwa WBiŚ.




Rys. 3.2. Filtr zaawansowany

Filtr standardowy ma możliwość nałożenia dowolnych warunków na maksymalnie trzy pola tabeli, warunki można łączyć operatorami **LUB** oraz **I**. Podobnie jak w przypadku autofiltru rekordy, które nie pasują do kryteriów filtrowania, zostaną ukryte. Po wyłączeniu filtrowania (**MG Dane>Filtr>Usuń filtr**) będą one ponownie wyświetlone.

Zaletą filtra standardowego jest możliwość definiowania kryteriów niezależnie od wartości, które już znajdują się w polach tabeli, oraz dostęp podczas tworzenia tych kryteriów także do warunków takich jak *wiekszy niż* oraz *mniej niż*. Przykładowe ustawienia (wraz z dodatkowymi opcjami omówionymi poniżej) pozwalające wyświetlić wszystkich studentów 1. oraz 2. roku budownictwa są przedstawione na Rys. 3.3.



Rys. 3.3. Filtr standardowy z dodatkowymi opcjami

W oknie filtra standardowego dostępny jest przycisk **Więcej** — wciśnięcie tego przycisku umożliwia ustawienie dodatkowych opcji filtrowania (patrz Rys. 3.3). Można włączyć rozróżnianie małych i wielkich liter podczas filtrowania (**Uwzględnij wielkość liter**), włączyć interpretowanie wyrażeń regularnych (**Wyrażenie regularne**), określić czy tabela z danymi zawiera nagłówki pól (**Zakres zawiera nagłówki kolumn**) oraz włączyć ukrywanie powtarzających się rekordów (**Bez duplikatów**). Możliwe jest także kopiowanie wyników filtrowania w nowe położenie w tym samym lub innym arkuszu, domyślnie filtrowanie jest wykonywane *w miejscu* i polega na ukrywaniu wierszy z rekordami, które nie spełniają kryteriów filtrowania. Aby skopiować wyniki w nowe położenie należy włączyć opcję **Kopiuj wyniki do...** i po wciśnięciu przycisku  (prawy dolny róg okna dialogowego) wskazać lewy górny narożnik obszaru, w

którym mają zostać umieszczone wyniki. Nowa tabela, która zostanie utworzona we wskazanym położeniu, jest niezależna od tabeli wejściowej i może być ponownie filtrowana jeżeli zachodzi taka potrzeba. Pozwala to na ominięcie ograniczenia maksymalnie trzech warunków, które można zdefiniować podczas pracy z filtrem standardowym.

3.3. Filtr zaawansowany

Przed uruchomieniem zaawansowanego filtrowania należy zdefiniować w arkuszu kryteria filtrowania tak, jak podczas pracy z funkcjami bazy danych były tworzone kryteria wyszukiwania. W dowolnym miejscu arkusza należy wstawić w jednym wierszu nagłówki wszystkich pól, na które będą zakładane warunki, a w kolejnych wierszach opisać te warunki. Podobnie jak w przypadku funkcji bazy danych warunki umieszczone w jednym wierszu są połączone logicznym operatorem **I**, warunki umieszczone w osobnych wierszach są połączone operatorem **LUB**.

Zadanie: skonstruuj listę studentów 1. roku budownictwa WBiŚ uprawnionych do otrzymywania stypendium naukowego.

Aby zadanie było łatwiejsze do opisania wprowadzimy dwa uproszczenia:



- w tabeli **STUDENCI** znajdują się wyniki z sesji, podczas której studenci musieli zaliczyć wyłącznie dwa przedmioty (Matematyka i Fizyka), pozwoli to znacznie skrócić formuły, które będą za moment konstruowane,
- jedyne warunki otrzymania stypendium naukowego to zaliczenie wszystkich przedmiotów w pierwszym terminie oraz uzyskanie średniej ocen większej lub równej 4 (bd).

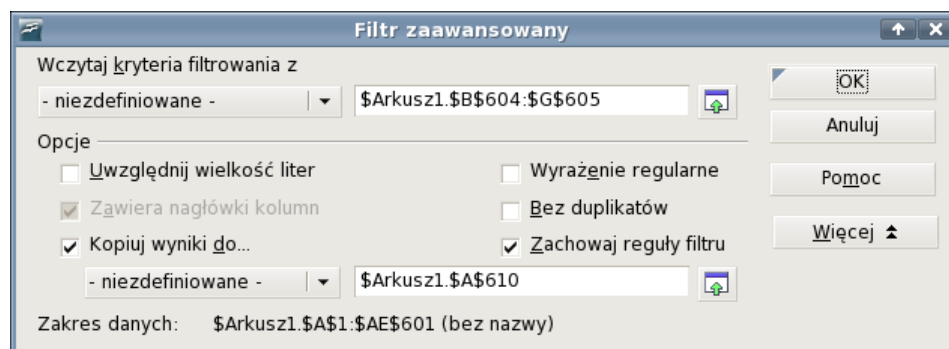
Pierwszym krokiem w obu przypadkach jest obliczenie końcowej oceny z każdego przedmiotu. Ponieważ arkusz ma służyć także do innych zadań, nie tylko do rozdzielania stypendiów naukowych, końcowa ocena musi być obliczona z uwzględnieniem ocen uzyskanych w drugim terminie (podczas sesji poprawkowej). W kolumnie **Y** tabeli dodamy oceny końcowe z Matematyki, w kolumnie **Z** oceny z Fizyki: **Y2 =MAX(M2:N2)** (oceną końcową jest najwyższa ocena uzyskana podczas dwóch terminów), **Z2 =MAX(O2:P2)**. Nowym polom nadajmy nazwy: **Y1 Matematyka** oraz **Z1 Fizyka**. Po obliczeniu oceny końcowej można obliczyć średnią ocen: **AA2 =ŚREDNIA(Y2:Z2)**, nowemu polu tabeli nadajemy nazwę **AA1 Średnia**. Nowe formuły należy skopiować do wszystkich wierszy tabeli.

Tabela 3.1. Kryteria filtrowania — stypendia naukowe

	A	B	C	D	E	F	G
603							
604		Kierunek	RokStudiów	Matematyka1	Fizyka1	Średnia	
605		budownictwo	1	>=3	>=3	>=4	
606							

Odpowiednie kryteria filtrowania są przedstawione w Tab. 3.1. Uzyskanie listy studentów 1. roku budownictwa WBiŚ uprawnionych do pobierania stypendium naukowego byłoby bardzo trudne do uzyskania z zastosowaniem autofiltru (nie można zakładać warunków *wiekszy lub równy*), filtr standardowy wymagałby dwukrotnego uruchomienia (pierwsze filtrowanie w nowe położenie pozwoli uzyskać listę studentów 1. roku budownictwa, drugie spośród tych studentów wybierze uprawnionych do pobierania stypendium). Zastosowanie filtra zaawansowanego pozwala uzyskać tę listę bezpośrednio.

Rys. 3.4 przedstawia okno filtra zaawansowanego z ustawionymi opcjami pozwalającymi uzyskać listę uprawnionych do pobierania stypendium na 1. roku budownictwa. Przed wywołaniem funkcji **MG Dane>Filtr>Filtr zaawansowany** należy uaktywnić dowolną komórkę w tabeli **STUDENCI**, dopiero wówczas można uruchomić filtr zaawansowany. W oknie ustawień należy wcisnąć przycisk  i wskazać kryteria filtrowania a po wciśnięciu przycisku **Więcej** należy określić gdzie będzie wstawiona nowa tabela (po wciśnięciu przycisku  w prawym dolnym okna dialogowego).



Rys. 3.4. Filtr zaawansowany z dodatkowymi opcjami

Kryteria filtrowania, skonstruowane na potrzeby tworzenia listy studentów uprawnionych do pobierania stypendiów, mogą być wykorzystane jako kryteria wyszukiwania dla funkcji `BD.ILE.REKORDÓW()` co pozwala poza skonstruowaniem listy błyskawicznie zliczyć stypendystów.

3.4. Stypendia naukowe

Jako przykład z zastosowaniem filtrowania i funkcji bazy danych obliczmy wysokość stypendiów naukowych i skonstruujemy listy studentów z informacją o wysokości przydzielonego stypendium.

Założenia:

- uczelnia może przeznaczyć na stypendia naukowe 40tys. zł miesięcznie,
- względna wysokość stypendiów w zależności od uzyskanej średniej ocen jest następująca:
 - 25%: $4,00 \leq \text{średnia} < 4,25$,
 - 50%: $4,25 \leq \text{średnia} < 4,50$,
 - 75%: $4,50 \leq \text{średnia} < 4,75$,
 - 100%: $4,75 \leq \text{średnia} \leq 5,00$,

gdzie 100% oznacza pełną wysokość najwyższego stypendium (inaczej mówiąc najniższe stypendium naukowe będzie stanowić jedną czwartą najwyższego),

- wszystkie przedmioty należy zaliczyć w pierwszym terminie.

Naszym zadaniem będzie takie dobranie wysokości pełnego stypendium, aby, zachowując opisane założenia dotyczące wysokości stypendiów, co miesiąc wydawano stypendia w łącznej wysokości 40tys. zł.

Pierwszym zadaniem jest zliczenie studentów, którzy zaliczyli wszystkie przedmioty w pierwszym terminie i uzyskali średnią ocen mieszczącą się w kolejnych, opisanych wyżej przedziałach. Podczas wykonywania tego zadania weźmiemy pod uwagę wszystkie informacje z tabeli i oceny z Matematyki, Fizyki, Chemii, Mechaniki, Języka obcego oraz Ekonomii. Należy, tak jak opisano to w jednym z poprzednich zadań, obliczyć końcową ocenę z każdego przedmiotu oraz średnią ocen. Kryteria wyszukiwania do zastosowania w funkcji `BD.ILE.REKORDÓW()` są przedstawione na Rys. 3.5, odpowiednie formuły zliczające studentów w poszczególnych grupach należy skonstruować samodzielnie.

W następnym kroku obliczymy liczbę stypendiów przeliczeniowych, które uczelnia będzie musiała wydać co miesiąc. Każdy student z najwyższą średnią otrzyma jedno pełne stypendium przeliczeniowe, studenci z niższą średnią otrzymają odpowiednio 3/4, połowę lub 1/4 stypendium przeliczeniowego. Przykładowo liczba stypendiów przeliczeniowych zdobytych przez studentów z najniższą średnią jest równa liczbie tych studentów podzielonej przez 4 (każdy dostaje 25% stypendium przeliczeniowego). Fragment arkusza z obliczeniami liczby stypendiów jest przedstawiony na Rys. 3.6.

Stąd już tylko krok do obliczenia łącznej liczby stypendiów przeliczeniowych (w przykładzie jest ich 19,5) i do obliczenia kwoty przypadającej na jedno stypendium przeliczeniowe. Znając relację stypendium za konkretną średnią do stypendium przeliczeniowego można ostatecznie obliczyć wysokości stypendiów.

Matematyka1 ≥3	Fizyka1 ≥3	Chemia1 ≥3	Mechanika1 ≥3	Język1 ≥3	Ekonomia1 ≥3	Średnia ≥4	Średnia <4,25
Matematyka1 ≥3	Fizyka1 ≥3	Chemia1 ≥3	Mechanika1 ≥3	Język1 ≥3	Ekonomia1 ≥3	Średnia ≥4,25	Średnia <4,5
Matematyka1 ≥3	Fizyka1 ≥3	Chemia1 ≥3	Mechanika1 ≥3	Język1 ≥3	Ekonomia1 ≥3	Średnia ≥4,5	Średnia <4,75
Matematyka1 ≥3	Fizyka1 ≥3	Chemia1 ≥3	Mechanika1 ≥3	Język1 ≥3	Ekonomia1 ≥3	Średnia ≥4,75	

Rys. 3.5. Stypendia: kryteria wyszukiwania

Liczba studentów:	Względna wysokość stypendium	Liczba stypendiów przeliczeniowych
19	25%	4,75
21	50%	10,50
3	75%	2,25
2	100%	2,00
Łącznie liczba stypendiów przeliczeniowych		19,50
Kwota do dyspozycji		40.000 zł
Wysokość pojedynczego stypendium przeliczeniowego		2.051,28 zł
	100%	2.051,28 zł
	75%	1.538,46 zł
	50%	1.025,64 zł
	25%	512,82 zł

Rys. 3.6. Stypendia: liczba stypendiów przeliczeniowych oraz obliczenie wysokości stypendiów

Aby zakończyć zadanie należy jeszcze skonstruować listy osób uprawnionych do pobierania stypendium w określonej wysokości. Kryteria filtrowania są już zdefiniowane (patrz Rys. 3.5), pozostaje wyłącznie zastosować filtr zaawansowany i skonstruować listy studentów.

Osobom zainteresowanym polecamy samodzielne rozwiązanie następującego problemu: ile uczelnia musiałaby wydać co miesiąc na stypendia, aby wysokość najniższego stypendium wynosiła 550,00zł (wzrost o 37,18zł). Podczas rozwiązywania tego zadania należy zastosować funkcję **MG Narzędzia>Szukaj wyniku**.