



[tu.koszalin.pl/bpi](http://tu.koszalin.pl/bpi)

## ZADANIA KONKURSOWE

### I etap

### XXVI edycja konkursu „Bieg po Indeks”

Zestaw 1

#### Zadania z informatyki

Treści zadań:

##### Zadanie 1.

Opracować w pseudokodzie lub dowolnym języku programowania funkcję, która dla każdej naturalnej wartości  $n$  większej od 1, umożliwi wygenerowanie struktury danych jak na rysunku 1.

a) b) c)

1	0	0	0	1
0	2	0	2	0
0	0	3	0	0
0	2	0	2	0
1	0	0	0	1

1	0	0	0	0	0	1
0	2	0	0	0	2	0
0	0	3	0	3	0	0
0	0	0	4	0	0	0
0	0	3	0	3	0	0
0	2	0	0	0	2	0
1	0	0	0	0	0	1

1	0	0	0	0	0	0	0	1
0	2	0	0	0	0	0	2	0
0	0	3	0	0	0	3	0	0
0	0	0	4	0	4	0	0	0
0	0	0	0	5	0	0	0	0
0	0	0	4	0	4	0	0	0
0	0	3	0	0	0	3	0	0
0	2	0	0	0	0	0	2	0
1	0	0	0	0	0	0	0	1

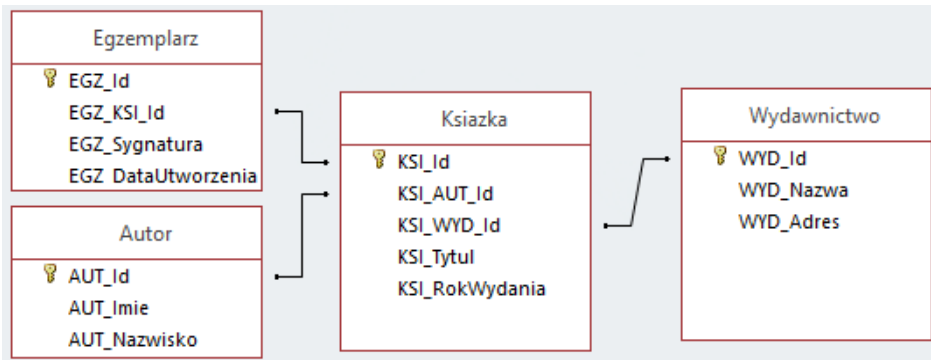
Rysunek 1. Schemat do analizy struktury danych do utworzenia. Przykład opracowany dla: a)  $n = 3$ ; b)  $n = 4$ ; c)  $n = 5$

##### Zadanie 2.

Prosta baza danych biblioteki może składać się zaledwie z kilku tabel, przykładowa realizacja takiej bazy danych w środowisku MS Access znajduje się na rysunku 2. Wypisz nazwy kolumn jakie zostaną wyświetlone po wykonaniu następujących instrukcji SQL:

1. `select k.KSI_Tytul as "Tytuł", count(*) as "Liczba" from egzemplarz e inner join ksiazka k on k.KSI_Id = e.EGZ_KSI_Id group by k.KSI_Tytul`
2. `select [k.KSI_Tytul], [k.KSI_RokWydania], [w.WYD_Nazwa] from Ksiazka k inner join Wydawnictwo w on k.KSI_WYD_Id = w.WYD_Id`
3. `SELECT [k.KSI_Tytul] AS Tytuł, Left(a.AUT_Imie,1) & '.' & ' ' & a.AUT_Nazwisko AS Autor FROM Ksiazka AS k INNER JOIN Autor AS a ON k.KSI_AUT_Id = a.AUT_Id;`

Pamiętaj, aby zapisać nazwy kolumn w odpowiedniej kolejności oraz z dodatkowymi znakami (jeśli występują).

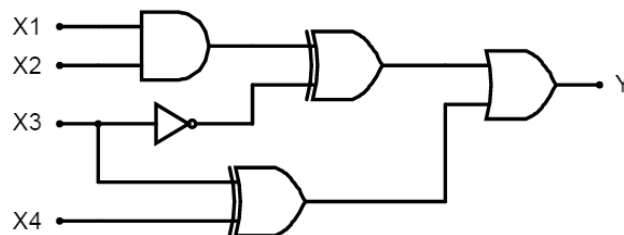


Rys. 2 Schemat bazy danych oraz widok projektowania tabel w środowisku MS Access

### Zadanie 3.

Na rysunku 3 przedstawiono układ kombinacyjny realizujący pewną funkcję logiczną czterech zmiennych. Przeanalizuj podany układ i wykonaj następujące zadania.

- Rozpisz tablicę prawdy uwzględniającą wszystkie możliwe kombinacje wejść logicznych X1, X2, X3, X4 i odpowiadającą im reakcję wyjścia logicznego Y.
- Rozpisz funkcję logiczną układu. Zadbaj o to, by funkcja logiczna miała jak najprostszą postać. Możesz w tym celu wykorzystać na przykład metodę Karnaugh'a.



Rys. 3 Układ kombinacyjny

### Rozwiązanie 3.

Rozwiązanie do punktu a przedstawia tabela 1. Przykładowe rozwiązania do punktu b przedstawia rysunek 4.

### Zadanie 4.

Na listingu 1 przedstawiono kod programu, który „odwiedza” wybrane komórki dwuwymiarowej tablicy i kasuje wartość tych komórek. Wypisz kolejne litery, które zostaną skasowane w tablicy. Litery wypisz posortowane od najmniejszej sumy indeksów.

```
#include "stdafx.h"
#include <iostream>
using namespace std;
int main() {
    char tab[5][10] = {
        { 'a', 'b', 'c', 'd', 'f', 'g', 'i', 'k', 'j', 'l' },
        { 'c', 'f', 'C', 'f', 'i', 'e', 'i', 'k', 'Z', 'M' },
        { 'd', 'f', 'o', 'd', 'g', 'W', 'I', 'p', 'W', 'N' },
        { 'g', 't', 'O', 'R', 'p', 'W', 'l', 'D', 'S', 'E' },
        { 'h', 'r', 'm', 'Z', 'A', 'W', 'L', 'P', 'A', 'E' } };
    for (int i = 0; i < 5; i++) {
        for (int j = 8; j > 0; j--) {
            if ((i%j >= j/2) && (j%(i+1) == i))
                tab[i][j] = 0;
        }
    }
    system("pause");
}
```

```
return 0; }
```

Listing 1

### Zadanie 5.

Na listing 2 przedstawiono kod pewnego programu w języku C++.

Przeanalizuj ten kod i odpowiedz na następujące pytania:

- Jak nazywa się ciąg generowany przez powyższą implementację funkcji f?
- Jakie niebezpieczeństwo niesie ze sobą sposób implementacji funkcji f?

Następnie napisz funkcję f pozbawioną wady opisanej w podpunkcie b).

```
#include <stdio.h>
#include <stdlib.h>

int f(int n) {
    if (n < 3) {
        return 1;
    }
    return f(n - 2) + f(n - 1);
}

int main(int argc, char *argv[])
{
    printf("%i\n", f(6));
    system("pause");
    return 0;
}
```

Listing 2