

**Uniwersytet Rzeszowski**  
**Wydział Matematyczno-Przyrodniczy**

Przedmiot: **SYSTEMY OPERACYJNE 2** – Laboratorium

Nazwisko i imię <b>Paweł Kolano</b>	Laboratorium nr 5  Temat:  <b>METODY ROZWIĄZYWANIA PROBLEMÓW</b>	Data wykonania: 01.11.2020r.
Grupa laboratoryjna nr 1		Data oddania:
Ocena:		

## Zadanie 1.

Przetestuj zamieszczone wyżej fragmenty kodu i sprawdź ich działanie.

### Wykonanie:

Zadanie to polegało na przepisaniu fragmentów kodu, które znajdowały się w materiale edukacyjnym mającym nas zapoznać z tematem.

## Zadanie 2.

Wygeneruj wszystkie możliwe podzbiory zbioru czteroelementowego. Do rozwiązania tego problemu użyj metody brutalnej siły.

### Wykonanie:

```
void printAllSubset(const char *znaki) {
    cout << "{}" << "\n\n";

    for (int i = 0; i < 4; ++i) {
        cout << '{' << znaki[i] << " ";
    }

    cout << "\n\n";

    for (int i = 0; i < 4; ++i) {
        for (int j = 0; j < 4; ++j) {
            if (i != j) {
                cout << '{' << znaki[i] << " ", " << znaki[j] << " ";
            }
        }
    }

    cout << "\n\n";

    for (int i = 0; i < 4; ++i) {
        for (int j = 0; j < 4; ++j) {
            for (int k = 0; k < 4; ++k) {
                if (i != j && i != k && j != k) {
                    cout << '{' << znaki[i] << " ", " << znaki[j] << " ", " << znaki[k] << " ";
                }
            }
        }
    }

    cout << "\n\n";

    for (int i = 0; i < 4; ++i) {
        for (int j = 0; j < 4; ++j) {
            for (int k = 0; k < 4; ++k) {
                for (int l = 0; l < 4; ++l) {
                    if (i != j && i != k && i != l && j != k && j != l && k != l) {
                        cout << '{' << znaki[i] << " ", " << znaki[j] << " ", " << znaki[k] << " ", " << znaki[l] << " ";
                    }
                }
            }
        }
    }
}
```

[Rys 1] - Kod funkcji w programie

```
Generowanie wszystkich podzbiorów zbioru czteroelementowego:
{}
{a} {b} {c} {d}
{a, b} {a, c} {a, d} {b, a} {b, c} {b, d} {c, a} {c, b} {c, d} {d, a} {d, b} {d, c}
{a, b, c} {a, b, d} {a, c, b} {a, c, d} {a, d, b} {a, d, c} {b, a, c} {b, a, d} {b, c, a} {b, c, d} {b, d, a} {b, d, c} {c, a, b} {c, a, d} {c, b, a} {c, b, d}
{c, d, a} {c, d, b} {d, a, b} {d, a, c} {d, b, a} {d, b, c} {d, c, a} {d, c, b}
{a, b, c, d} {a, b, d, c} {a, c, b, d} {a, c, d, b} {a, d, b, c} {a, d, c, b} {b, a, c, d} {b, a, d, c} {b, c, a, d} {b, c, d, a} {b, d, a, c} {b, d, c, a}
{c, a, b, d} {c, a, d, b} {c, b, a, d} {c, b, d, a} {c, d, a, b} {c, d, b, a} {d, a, b, c} {d, a, c, b} {d, b, a, c} {d, b, c, a} {d, c, a, b} {d, c, b, a}
```

[Rys 2] – Wygenerowanie wszystkich podzbiorów zbioru cztero-elementowego

Pętla wypisuje wszystkie rozwiązania. Najpierw zbiór pusty, następnie podzbiór 1 elementowy, następnie 2 elementowy...

### Zadanie 3.

Napisz program, z wykorzystaniem metody dziel i zwyciężaj obliczający n-ty wyraz ciągu Fibbonacciego.

#### Wykonanie:

```
102 int fibb(int n) {
103     if (n <= 1) return n;
104     return fibb(n-1) + fibb(n-2);
105 }
```

[Rys 3] - Kod programu dla zadania 3

```
*****
***** zadanie 3 *****
*****

Metoda dziel i zwyciezaj

Podaj nr wyrazu ciagu fibbonaciego: 5

Wyraz: 5 ciagu fibbonaciego to: 5

lokyiec@Pawels-MacBook-Pro lab5 %
```

[Rys 4] - Wynik programy obliczający 5-ty wyraz ciągu

Zadanie to polegało na utworzeniu algorytmu rekurencyjnego obliczającego kolejne wyrazy ciągu Fibbonacciego.

### Zadanie 4.

Napisz program, z wykorzystaniem metody programowania dynamicznego obliczający n-ty wyraz ciągu Fibbonacciego.

#### Wykonanie:

```
107 int fibbDyn(int n) {
108     int a, b;
109     if(n == 0) return 0;
110     a = 0;
111     b = 1;
112     for(int i = 0; i < (n-1); i++) {
113         b += a;
114         a = b-a;
115     }
116     return b;
117 }
```

[Rys 5] - Kod funkcji w programie

```
*****
***** zadanie 4 *****
*****

Podaj nr wyrazu ciagu fibbonaciego rekurencyjnie: 5

Wyraz: 5 ciagu fibbonaciego to: 5

lokyiec@Pawels-MacBook-Pro lab3 %
```

[Rys 6] - Wynik programy obliczający 5-ty wyraz ciągu

Zadanie to polegało na utworzeniu algorytmu iteracyjnego obliczającego kolejne wyrazy ciągu Fibbonacciego.

