

Rozważmy dwuwymiarowy problem wyszukiwania wzorca w tekście. Przyjmijmy, że dostępny alfabet to symbole kodu szesnastkowego czyli $\Sigma = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F\}$, natomiast szukany wzorzec ma postać:

A	B	C
B		
C		

Zatem szukamy wszystkich wystąpień ciągu "ABC" w poziomie i pionie, które mają wspólny symbol "A". Do testowania algorytmu proszę użyć załączonych z instrukcją plików z tekstem: N_pattern.txt (gdzie N oznacza wielkość macierzy symboli $N \times N$, $N = 1000, 2000, 3000, 4000, 5000, 8000$).

Zadanie 1 (0.5 pkt) Zaimplementuj algorytm "naiwny" wyszukiwania wzorca dwuwymiarowego w tekście. Do testowania użyj macierzy o rozmiarze 1000×1000 .

Zadanie 2 (1.0 pkt) Zaimplementuj algorytm Karpa-Rabina wyszukiwania wzorca dwuwymiarowego w tekście. Do testowania użyj macierzy o rozmiarze 1000×1000 .

Zadanie 3 (0.5 pkt) Porównaj czasy wykonywania algorytmów wyszukiwania wzorców z zadań 1 i 2 dla macierzy symboli $N = 1000, 2000, 3000, 4000, 5000$, (8000 dla chętnych). Dla każdego pliku policz liczbę wystąpień wzorca.

Który algorytm zadział lepiej? Przeanalizuj uzyskany wynik porównując ze złożonością obliczeniową algorytmów omówioną na wykładzie.
