

Adam Cherek s464922

zadanie algorytmiczne nr 5:

algorytm będzie dla danego słowa P w postaci listy i liczby n oznaczającej potęgę do której słowo ma być podniesione, konkatenował słowo P n krotnie z samym sobą przy pomocy CON(P,Q) z poprzedniego zestawu.

funkcja POW(słowoP, liczbaN): zwraca słowo będące n-tą potęgą słowa wykorzystując CON(P,Q)

```
{
    int potęga := 1          % inicjujemy potrzebne zmienne %
    list P:= słowoP
    if liczbaN = 0 then      % jeżeli słowo jest puste to zwracamy słowo
                            puste %
        return []
    else                    % jeżeli nie jest puste to konkatenujemy słowo z
                            samym sobą aż osiągniemy wymaganą potęgę %
        while potęga < liczbaN do
            P:= CON(słowoP, P)
            potęga:= potęga + 1
        return P            % zwracamy P aktualnie będące n-tą potęgą
                            słowa %
}
```

Drugie zadanie na kolejnej kartce gdyż nie chce by zostało rozdzielone w połowie

zadanie algorytmiczne nr 6:

algorytm będzie przesuwiał się po słowie P wykorzystując POS(P,n) oraz LENGTH(P) i sprawdzając czy kolejne symbole słowa Q są kolejnymi symbolami słowa P, aż do ustalenie że słowo Q jest pod słowem lub końca słowa P

funkcja Podslowo(słowoP , słowoQ): zwraca TAK lub NIE w zależności czy słowo Q jest pod słowem słowa P czy nie, wykorzystując procedury POS(P,n) oraz LENGTH(P)

```
{
    int pasujące_symbole := 0           % inicjujemy potrzebne zmienne %
    list P := słowoP
    list Q := słowoQ
    if LENGTH(P) < LENGTH(Q) then      % sprawdzamy czy słowo P nie
                                        jest krótsze od słowa Q %
        return NIE
    else
        for i = 1
        to LENGTH(P) do
            if POS(P,i) = POS(Q,(pasujące_symbole + 1)) then
                pasujące_symbole := pasujące_symbole + 1
                if pasujące_symbole = LENGTH(Q) then
                    return TAK
            else
                pasujące_symbole := 0
        return NIE
}
```

% w pętli for zrobiło się mało miejsca więc opisze tutaj, w pętli przejeżdżamy po kolejnych symbolach słowa P aż do jego końca, sprawdzając czy odpowiadają kolejnym symbolom słowa Q, jeśli tak to rośnie zmienna pasujące_symbole, jeśli nie to zmienna się zeruje. Jeżeli zmienna pasujące symbole osiągnie wartość długości słowa Q zanim pętla się skończy to zwracane jest TAK, jeżeli nie to zwracane jest NIE %