# ZLOŻONOŚĆ OBLICZENIOWA - WYK. 4

1. Maszyny Turinga z poradą (*advice*)

   - model niejednorodny, ale sekwencyjny
   - Definicja: maszyna $M$ wraz z ciągiem słów $k_1, k_2, \ldots \in \Sigma^*$ rozpoznaje język $L$ wtw.
     $$w \in L \iff k_{|w|}\$w \in L(M)$$
     Czas działania mierzymy względem $|w|$ a nie całego słowa; to znaczy że porada wykładnicza wymusza wykładniczy czas działania (trzeba ją przeczytać).
   - klasa $\mathbf{P/poly}$: języki rozpoznawalne w czasie wielomianowym z poradami o rozmiarze wielomianowym.

2. Twierdzenie: każdy język rozpoznawalny ciągiem obwodów o rozmiarze wielomianowym należy do $\mathbf{P/poly}$. (*Uwaga:* w drugą stronę inkluzja też zachodzi z poprzedniego twierdzenia, bo poradę można zaszyć w obwodzie.)

   *Dowód:* $k_n$ to reprezentacja obwodu $C_n$. Dany obwód, na danym słowie (tj. zestawie wartości bramek wejściowych) można ewaluwoać w czasie wielomianowym.

3. $\mathbf{P/poly}$ to cały czas klasa niejednorodna i zawiera języki nierozstrzygalne. Jest jednak przydatna do modelowania języków, które dają się szybko rozpoznawać po (być może bardzo kosztownym) preprocessingu. W kryptografii zakłada się też czasem, że przeciwnicy mają taką moc obliczeniową.

   Główna hipoteza na temat tej klasy: czy $\mathbf{NP} \subseteq \mathbf{P/poly}$? Wszyscy są zdania że nie. To jest mocniejsze stwierdzenie niż $\mathbf{P} \neq \mathbf{NP}$, bo jest jasne że $\mathbf{P} \subseteq \mathbf{P/poly}$.

4. Obliczanie funkcji w pamięci logarytmicznej:

   - taśma wejściowa tylko do odczytu,
   - taśma robocza długości logarytmicznej,
   - taśma wyjściowa, po której głowica może się poruszać tylko w prawo.

   Uwaga: w pamięci logarytmicznej można obliczyć wyjście dużo dłuższe niż logarytmiczne (ale wielomianowo długie).

5. Twierdzenie: funkcje obliczalne w pamięci logarytmicznej są zamknięte ze względu na składanie. *Dowód*: wiadomo jaki.

6. Jednorodny ciąg obwodów: obliczalny w pamięci logarytmicznej, tj.: istnieje maszyna, która działa w pamięci logarytmicznej i na słowie $0^n$ oblicza reprezentację $n$-tego obwodu.

   Wniosek: te obwody są wielomianowego rozmiaru względem $n$.

7. Fakt: język jest rozpoznawany jednorodnym ciągiem obwodów wtedy i tylko wtedy kiedy jest w $\mathbf{P}$.

   *Dowód:* Z lewa w prawo jest oczywiste: mając dane słowo wejściowe rozmiaru $n$, oblicz $n$-ty obwód i ewaluuj go.

   Z prawa w lewo: wniosek z twierdzenia z zeszłego tygodnia. Trzeba tylko sprawdzić dokładnie, że tamten ciąg obwodów jest obliczalny w pamięci logarytmicznej. Ale to jest prawda: trzeba tylko pamiętać, który poziom obwodu aktualnie wypisujemy i które miejsce w tym rzędzie, a to się mieści w pamięci logarytmicznej.

8. Dodatkowe ograniczenia: głębokość obwodu.

   - klasa $\mathbf{AC}^i$: języki rozpoznawalne jednorodnymi ciągami obwodów o głębokości $\mathcal{O}(\log^i n)$.
   - najciekawsze przypadki: $\mathbf{AC}^0$ (głębokość stała), $\mathbf{AC}^1$.
   - klasa $\mathbf{NC}^i$: to samo, ale dopuszczamy tylko bramki o stopniu wejściowym 2 (bardziej praktyczne). Klasa $\mathbf{NC}^0$ jest nieciekawa.
   - klasy $\mathbf{NC}$, $\mathbf{AC}$.
   - Przykład: mnożenie macierzy binarnych jest w klasie $\mathbf{NC}^1$.

9. *Fakt:* $\mathbf{NC}^i \subseteq \mathbf{AC}^i \subseteq \mathbf{NC}^{i+1}$. Wniosek: $\mathbf{AC} = \mathbf{NC}$.

10. Idea: $\mathbf{NC}$ to te problemy, które się dobrze zrównoleglają (tj. dają się zrobić w czasie polilogarytmicznym na maszynie z odpowiednio dużą liczbą procesorów). Główny problem otwarty: czy $\mathbf{NC} = \mathbf{P}$? Wszyscy myślą że nie.

11. Ciąg inkluzji:

    $$\mathbf{AC}^0 \subseteq \mathbf{NC}^1 \subseteq \mathbf{AC}^1 \subseteq \mathbf{NC}^2 \subseteq \cdots \subseteq \mathbf{AC} = \mathbf{NC} \subseteq \mathbf{P} \subseteq \mathbf{NP} \subseteq \mathbf{PSPACE}$$

    Wszyscy myślą, że wszystkie te inkluzje są ostre, ale na pewno wiemy tylko o dwóch:

    - $\mathbf{AC}^0 \subsetneq \mathbf{NC}^1$,
    - $\mathbf{NC} \subsetneq \mathbf{PSPACE}$.

    Drugi punkt wynika z tego, że $\mathbf{NC}^1 \subseteq L$ (zostawiamy na ćwiczenia), a także $\mathbf{NC} \subseteq \mathbf{polyL}$, i z tw. o hierarchii pamięciowej.

    Pierwszy punkt zrobimy na wykładzie, bo jest to jedno z nielicznych znanych ograniczeń dolnych na złożoność istotnego problemu.

12. Język parzystości PARITY: język tych słów na alfabetem binarnym, w których jest parzysta liczba jedynek.

    *Fakt:* PARITY$\in \mathbf{NC}^1$. **L**atwy obwód zliczający jedynki modulo 2.

    *Twierdzenie:* PARITY$\notin \mathbf{AC}^0$. To zrobimy za tydzień. Na rozgrzewkę ćwiczenie: pokazać, że PARITY nie da się rozpoznać obwodem o rozmiarze wielomianowym i głębokości 2.