

Zagadnienia Egzaminacyjne

- Języki regularne; należy
 - znać różne ich reprezentacje: wyrażenia regularne, DFA, NFA, epsilon-NFA oraz umieć je przekształcać pomiędzy sobą,
 - umieć konstruować wyrażenia regularne, DFA i NFA, epsilon-NFA dla prostych j. reg.,
 - znać lemat o pompowaniu i umieć go używać
 - wiedzieć na jakie operacje j. reg. są zamknięte a na jakie nie,
 - znać złożoność podstawowych problemów decyzyjnych dotyczących j. reg.,
 - znać tw. Myhill’a-Nerode’a i umieć z niego korzystać, wiedzieć czym jest A-równoważność, L-równoważność oraz umieć konstruować automat minimalny,
 - wiedzieć czym jest słowo resetujące.
- Języki bezkontekstowe; należy
 - znać różne ich reprezentacje: CFG, PDA, PDA akceptujące pustym stosem oraz umieć je przekształcać pomiędzy sobą,
 - umieć konstruować CFG, PDA dla prostych CFL,
 - wiedzieć czym jest wywód, drzewo wyvodu, gramatyki jednoznacznie i niejednoznacznie oraz języki wewnątrznie niejednoznaczne,
 - umieć przekształcić gramatykę do normalnej postaci Chomsky’ego, znać algorytm CYK, lemat Ogdena oraz lemat o pompowaniu dla CFL i umieć ich używać,
 - wiedzieć na jakie operacje CFL są zamknięte a na jakie nie,
 - znać złożoność podstawowych problemów decyzyjnych dotyczących CFL,
 - znać podklasy CFL tj. DCFL, LCFL oraz REG i umieć je pomiędzy sobą rozróżniać.
 - wiedzieć czym są gramatyki liniowe, (silnie) (prawo-) lewoliniowe i jak jest ich moc.
- Języki kontekstowe; należy:
 - znać reprezentacje: CSG oraz LBA
 - umieć konstruować CSG oraz LBA dla prostych CSL,
 - wiedzieć na jakie operacje są zamknięte CSL.
- Języki rekurencyjne i rekurencyjnie przeliczalne: należy

- znać tezę Churcha
- wiedzieć czym jest jej wysokość maszyna Turinga i wiedzieć, że nie można nazywać jej automatem
- znać definicje R i RE oraz konstruować MT dla prostych problemów z R oraz RE,
- znać moc NTM, k-taśmowej TM, 1-taśmowej TM, automatu z k stosami oraz gramatyk typu 0 i umieć dowieść odpowiednich równoważności i zawierań,
- znać i umieć dowieść związku pomiędzy obliczaniem funkcji (częściowych) i rozpoznawaniem języków przez maszyny Turinga,
- wiedzieć na jakie operacje są zamknięte R oraz RE,
- wiedzieć czym są enumeratory i jaki jest ich związek z R oraz RE,
- Nierozstrzygalność; należy
 - wiedzieć czym jest uniwersalna maszyna Turinga
 - znać i umieć dowieść złożoności języka L_d oraz L_u .
 - znać pojęcie redukcji (Turinga) pomiędzy problemami oraz jej własności (redukcji)
 - umieć redukować z L_d oraz L_u i rozumieć czego w ten sposób można dowieść,
 - znać sformułowanie oraz dowód tw. Rice’a,
 - znać nierozstrzygalne problemy tak jak: problem POSTa, kafelkowanie oraz problemy dotyczące gramatyk bezkontekstowych i odpowiednie dowody.
- Złożoność obliczeniowa; należy
 - znać definicje klas złożoności: LOGSPACE, NLOGSPACE, P, NP, coNP, P^{NP} , NP^{NP} , $coNP^{NP}$, PSPACE, NPSPACE, EXPTIME to która klasa się w której zawiera oraz umieć to udowodnić,
 - wiedzieć czym jest redukcja wielomianowa many-to-one (Karpa), redukcja wielomianowa Turinga (Cooke’a) oraz redukcja w pamięci logarytmicznej, znać i własności,
 - wiedzieć czym są problemy zupełne i trudne względem redukcji wielomianowej oraz redukcji w pamięci logarytmicznej dla wymienionych wyżej klas,
 - znać sformułowanie i dowód tw. Cooke’a, tw. Savitcha, tw. Immermana–Szelepcsényi’ego,
 - znać dowód tego, że TAUTOLOGY jest coNP-zupełne,
 - znać dowód tego, że TQBF jest PSPACE-zupełny,
 - umieć przeprowadzać proste redukcje pomiędzy problemami oraz umieć pokazywać zawieranie w odpowiednich klasach złożoności, patrz zadania z list 11–13.
- Poza tym należy zastanawiać się jak mają się do siebie wszystkie klasy złożoności od REG poprzez PSPACE aż do RE.