

Matematyczne podstawy informatyki.

Początkowo pojęcia informatyczne były tworzone, rozwijane i analizowane przez matematyków, dopiero gdzieś około 1960 roku informatyka zaczęła wyraźnie wyodrębniać się z matematyki. Wykład będzie poświęcony tym zagadnieniom informatycznym, które były rozwijane przez matematyków w początkowym okresie tworzenia informatyki teoretycznej. Będzie obejmować trochę historii, ale w większości będzie poświęcony zagadnieniom matematycznym, które legły u podstaw informatyki.

Szkic programu wykładu:

- 1) Formalizacja rachunku kwantyfikatorów.
- 2) Prace Kurta Gödla: funkcje rekurencyjne, tw. o niezupełności arytmetyki.
- 3) Alonzo Church: rachunek lambda, teza Churcha, twierdzenie o nierozstrzygalności arytmetyki i rachunku kwantyfikatorów.
- 4) Alan Turing i maszyna Turinga.
- 5) Funkcje rekurencyjne według Herbranda i Gödla.
- 6) Herbrand i twierdzenie Herbranda.
- 7) Zastosowania funkcji rekurencyjnych: abstrakcyjna złożoność obliczeniowa.
- 8) Ewentualnie: rozstrzygalność teorii dodawania (arytmetyki Presburgera), a może i teorii mnożenia.

Literatura:

- 1) Roman Murawski, Funkcje rekurencyjne i elementy metamatematyki. Problemy zupełności, rozstrzygalności, twierdzenia Gödla, Wydawnictwo Naukowe UAM, Pozna, wyd. czwarte 2010
- 2) M. Machtey, P. Young, An Introduction to The General Theory of Algorithms, 1978
- 3) J. R. Shoenfield, Mathematical Logic, 1967
- 4) S. C. Kleene, Introduction to Metamathematics, 1952
- 5) E. Mendelson, Introduction to Mathematical Logic, 1964
- 6) prace oryginalne.

Kilka uwag:

Wraz z upływem czasu informatyka staje się coraz bardziej praktyczna. Ma to wpływ na programy studiów i odbija się na wykształceniu ogólnym. Treści bardziej abstrakcyjne są coraz częściej pomijane. Odnoszę wrażenie, że studenci nie zapoznają się już z elementarnymi własnościami obliczalności. Np. nie wiedzę, że są takie zadania obliczalne, dla których nie ma algorytmu optymalnego, a dowolny algorytm rozwiązujący takie zadanie można ulepszyć i to w zadanym stopniu. Fakt ten nie ma chyba żadnego znaczenia praktycznego. Za to pozwala lepiej rozumieć obliczalność.

Informatycy bardzo często korzystają też z osiągnięć matematyków. Wiele algorytmów powstało prawdopodobnie dlatego, że ich twórcy mieli bardzo dobre przygotowanie matematyczne. Także samo pojęcie obliczalności zostało po raz pierwszy zdefiniowane przy okazji rozwiązywania problemów dotyczących podstaw matematyki. Idea Prologu tkwiła już chyba w pierwszych wyobrażeniach Goedla o obliczalności. Wyobrażenia te zostały uzupełnione przez Herbranda i następnie sformalizowane (choć niekoniecznie wykorzystane przez twórców Prologu). Idea programowania funkcjonalnego powstała chyba w wyniku bardzo nieudanej próby stworzenia podstawowej dla matematyki teorii, konkurencyjnej w stosunku do teorii mnogości.

Wykład miałby też charakter historyczny. Będzie zawierać trochę faktów z historii pewnego fragmentu matematyki i jednocześnie z historii informatyki. Pierwsza część wykładu będzie poświęcona przedstawieniu sformalizowanego systemu logicznego i pojęciu prawdy. Zostanie przedstawione kilka klasycznych definicji obliczalności. Będą też rozważane różne problemy (zadania) odgrywające rolę w początkowym okresie rozwoju informatyki. Będziemy zajmować się ich rozstrzygalnością i ewentualnie złożonością. Np. twierdzenie Herbranda podaje metodę badania, czy dana formuła jest tautologią rachunku kwantyfikatorów, która jest uogólnieniem znanej metody zero-jedynkowej.