informatyka+

Algorytmika i programowanie

Bazy danych

Multimedia, grafika i technologie internetowe

Sieci komputerowe

Tendencje w rozwoju informatyki i jej zastosowań

Człowiek – najlepsza inwestycja

KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

Warszawska Wyższa Szkoła Informatyki

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY

Projekt współfinansowany ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego.

informatyka+

Wszechnica Popołudniowa: Tendencje w rozwoju informatyki i jej zastosowań

Informatyka – klucz do zrozumienia, kariery, dobrobytu

Maciej M. Sysło

Człowiek – najlepsza inwestycja

KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

Warszawska Wyższa Szkoła Informatyki

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY

Projekt współfinansowany ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego.

Informatyka – klucz do zrozumienia, kariery, dobrobytu

Rodzaj zajęć: Wszechnica Popołudniowa
Tytuł: Informatyka – klucz do zrozumienia, kariery, dobrobytu
Autor: prof. dr hab. Maciej M Sysło

Redaktor merytoryczny: prof. dr hab. Maciej M Sysło

Zeszyt dydaktyczny opracowany w ramach projektu edukacyjnego **Informatyka+** — ponadregionalny program rozwijania kompetencji uczniów szkół ponadgimnazjalnych w zakresie technologii informacyjno-komunikacyjnych (ICT).
www.informatykaplus.edu.pl
kontakt@informatykaplus.edu.pl

Wydawca: Warszawska Wyższa Szkoła Informatyki
ul. Lewartowskiego 17, 00-169 Warszawa
www.wwsi.edu.pl
rektorat@wwsi.edu.pl

Projekt graficzny: FRYCZ I WICHA

Warszawa 2011

KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

WARSZAWSKA WYŻSZA SZKOŁA INFORMATYKI

UNIA EUROPEJSKA
EUROPEJSKI FUNDUSZ SPOŁECZNY

Projekt współfinansowany ze środków Unii Europejskiej w ramach Europejskiego Funduszu Społecznego.

Informatyka – klucz do zrozumienia, kariery, dobrobytu

Maciej M. Sysło

Uniwersytet Wrocławski, UMK w Toruniu

syslo@ii.uni.wroc.pl

syslo@mat.uni.torun.pl

http://mmsyslo.pl/

Streszczenie

Wykład składa się z dwóch części – w pierwszej jest krótko omówiona edukacja informatyczna, jej stan, wzloty i upadki, cele oraz dalsze kierunki rozwoju. Opis edukacji informatycznej i informatyki jako dziedziny stanowi tło dla rozważań na temat możliwych karier w informatyce, co ma zachęcić słuchaczy do poważnego zainteresowania się rozwijaniem swoich informatycznych umiejętności włącznie z podjęciem studiów na kierunku informatyka lub pokrewnym. Przedstawione są najpierw kariery w informatyce, te tylko z klasą, jak John Napier, autorzy RSA, Samuel Morse i David Huffman, Claude Shannon oraz anonimowi wynalazcy patentów związanych z maszynami do pisania. Krótko komentujemy również znaczenie dla informatyki i jej zastosowań karier współczesnych, którym poza klasą towarzyszy również „kasa". Ostatnia część wykładu jest poświęcona omówieniu wybranych wyzwań, czyli problemów, które czekają na adeptów informatyki. Wśród nich jest problem komiwojażera, cała gama problemów związanych z liczbami pierwszymi oraz jeden z problemów milenijnych (czy P = NP).

Spis treści

1 WPROWADZENIE

Tytuł tego wykładu nawiązuje do tytułu książki Andrzej Targowski *Informatyka – klucz do dobrobytu*, która ukazała się w 1971 roku. Wtedy było bardzo silne przekonanie wśród osób zajmujących się informatyką, że komputery i cała dziedzina z nimi związana może przyczynić się do znaczącego poprawienia warunków życia nie tylko osób zajmujących się informatyką, ale całego społeczeństwa. Tak się jednak nie stało, chociaż wykorzystanie komputerów przyczynia się do rozwoju dziedzin, w których są używane, również w życiu zwykłych obywateli. Jednym z celów tego wykładu jest przekonanie słuchaczy, że droga do dobrobytu, na której pojawiają się komputery i informatyka, wiedzie w pierwszym rzędzie przez zrozumienie ich istoty, działania, możliwości, kierunków rozwoju, a także ograniczeń. Tak jest rzeczywiście, wbrew powszechnemu mniemaniu istnieją obliczenia, których nie jest w stanie wykonać żaden komputer, co więcej – nawet wszystkie istniejące w świecie komputery razem wzięte nie są w stanie. Cała nadzieja w nowych algorytmach i rozwiązaniach.

Ten wykład jest adresowany do uczniów, których chcemy zainteresować informatyką tak, aby ta dziedzina stała się ich ulubionym zainteresowaniem i w konsekwencji, by podjęli studia na kierunkach informatycznych lub o zbliżonych profilach i w przyszłości związali się z kariera informatyczną.

Informatyka to obecnie ugruntowana dziedzina wiedzy, której zastosowania można znaleźć niemal w każdej innej dziedzinie. Ilustrują to inne wykłady w tym projekcie, dotyczące na przykład: ekonomii, kryptografii, gier, medycyny, mózgu. Wokół informatyki narosło niestety wiele nieporozumień, na ogół związanych z tym, że obecnie łatwo można posiąść podstawowe umiejętności posługiwania się komputerem i jego oprogramowaniem ani nie będąc informatykiem, ani nie kształcąc się w tym kierunku.

Obecnie jednak niemal każdy człowiek, posługując się komputerem powinien w jakimś zakresie znać głębiej jego działanie, a zwłaszcza sposoby jego wykorzystania w różnych sytuacjach i do rozwiązywania różnych problemów. Piszemy o tym w rozdziale 2.3.

W dalszej części, dla kilku wybranych problemów i ilustrujemy ich własności i rozwiązania oraz komentujemy rolę tych problemów w rozwoju metod komputerowych. Niektóre z przedstawionych problemów nadal stwarzają duże wyzwanie dla informatyków, zarówno pracujących nad konstrukcjami nowych komputerów, jak i nad coraz doskonalszym wykorzystaniem tych istniejących.

Z wieloma z opisywanych faktów, problemów i ich rozwiązań są związane znane w świecie informatycznym nazwiska. Za wieloma stoją osoby, które znajdują się na wysokich i najwyższych pozycjach najbogatszych osób w świecie. Chcemy Was przekonać, że jedni i drudzy reprezentują najwyższą klasę informatyków, a Ci najbogatsi, dodatkowo … mają z tego olbrzymią kasę. Źródeł sukcesów jednych i drugich można się doszukiwać głównie w ich osiągnięciach na polu informatyki.

2 KSZTAŁCENIE INFORMATYCZNE

W tym rozdziale przedstawiamy „wzloty i upadki” zajęć z informatyki oraz cele powszechnego kształcenia informatycznego.

2.1 PIERWSZE ZAJĘCIA Z INFORMATYKI

Pierwsze regularne zajęcia z informatyki w polskiej szkole miały miejsce w połowie lat 60. XX wieku – był to przedmiot „Programowanie i obsługa maszyn cyfrowych” prowadzony w III LO we Wrocławiu. Uczniowie pisali programy w zeszytach, a później uruchamiali je na komputerze Elliott 803 w Katedrze Metod Numerycznych Uniwersytetu Wrocławskiego. Programy służyły do wykonywania obliczeń matematycznych. Przed erą komputerów osobistych niewiele więcej można było robić z pomocą komputerów.

Dopiero z pojawieniem się IBM PC na początku lat 80. XX wieku stało się możliwe rzeczywiste upowszechnianie nauczania informatyki. Pierwszy program nauczania przedmiotu **elementy informatyki** dla liceów powstał w 1985 roku, a w 1990 roku – dla szkół podstawowych. Na początku lat 90. ukazał się pierwszy podręcznik

do elementów informatyki. Był on bardzo uniwersalny, gdyż mało zależał od konkretnego oprogramowania – miał aż 9 wydań i do dzisiaj można go spotkać na niektórych zajęciach.

Wydzielone zajęcia informatyczne w polskich szkołach były bardzo poważnie traktowane w kolejnych reformach systemu oświaty i nigdy pod żadnym naciskiem przedmiot informatyka nie został usunięty ze szkół, chociaż taki przykład płynął ze Stanów Zjednoczonych, gdzie od lat 90. komputery w szkołach były wykorzystywane głównie do kształcenia umiejętności z zakresu technologii informacyjno-komunikacyjnej. Obecnie w Stanach Zjednoczonych przywraca się kształcenia w zakresie informatyki, co ma powstrzymać spadek zainteresowania uczniów karierami informatycznymi, wynoszący w ostatnich latach aż 50% (podobnie jest w Wielkiej Brytanii).

2.2 REGRES EDUKACJI INFORMATYCZNEJ

Zmniejszone zainteresowanie uczniów kształceniem informatycznym w szkołach jest obserwowane nie tylko w Stanach Zjednoczonych. Powodów tego jest wiele. Z jednej strony, wiele osób, w tym nauczyciele i rodzice, nie uważa informatyki za niezależną dziedzinę nauki, a zatem także za szkolny przedmiot. Powszechnie wiele osób myli i utożsamia informatykę z technologią informacyjno-komunikacyjną i sprowadza edukację informatyczną do udostępniania uczniom i nauczycielom komputerów i Internetu w szkole i w domu. Nie odróżniają oni stosowania komputerów i sieci Internet od studiowania podstaw informatyki. W Polsce, malenie liczby kandydatów na studia informatyczne jest spowodowane również przez niż, na szczęście okresowy, oraz, w poszczególnych uczelniach, powstawaniem kolejnych szkół prywatnych.

Jest też wiele powodów zmniejszonego zainteresowania samych uczniów informatyką, jako dziedziną kształcenia i przyszłą karierą zawodową. Na początku informatyka była kojarzona z programowaniem komputerów, co wywoływało silny sprzeciw, gdyż uważano, że niewielu uczniów zostanie kiedyś programistami. Na przełomie lat 80. i 90. XX wieku tylko nieliczni uczniowie używali komputerów w szkole lub w domu przed wstąpieniem na uczelnię. Na przełomie XX i XXI wieku główny nacisk w szkołach zmienił się diametralnie – kształcono z zakresu korzystania z aplikacji biurowych i Internetu. Obecnie wielu przyszłych studentów zdobywa pierwsze doświadczenia informatyczne przed wstąpieniem na uczelnię, najczęściej poza szkołą. Co więcej, dostępne oprogramowanie umożliwia tworzenie nawet bardzo złożonych aplikacji komputerowych bez wcześniejszego zaznajomienia się z: logiką, metodami programowania, matematyką dyskretną, które należą do kanonu kształcenia informatycznego. W rezultacie, absolwenci szkół średnich nieźle radzą sobie z wykorzystaniem komputerów do zabawy, poszukiwań w sieci i do komunikowania się, ale znikoma jest ich wiedza na temat informatyki jako dyscypliny oraz o tym, jak funkcjonuje komputer i sieć komputerowa. Dorastając, mają oni na tyle dość styczności z technologią informatyczną, że nie interesuje ich rozwijaniem swoich umiejętności w tym zakresie na poziomie uczelni, a w konsekwencji – kreowanie nowej kultury i nowej technologii. Potrzebny jest więc sposób, jak umotywować uczniów, aby zainteresowali się tym, co dzieje się poza ekranem komputera, jak zbudowany jest komputer i sieć oraz jak działa oprogramowanie, a w dalszej perspektywie tworzyli własne rozwiązania informatyczne.

2.3 POTRZEBY ZMIAN

Chcąc zmienić opisaną sytuację, zajęcia informatyczne w szkołach powinny przygotowywać uczniów do dalszego kształcenia się w kierunkach związanych z informatyką, zamiast utwierdzać ich w przekonaniu, że ich wiedza i umiejętności w tym zakresie są wystarczające. Czasem uczniowie są niezadowoleni i zniechęceni sposobem prowadzenia w szkole zajęć informatycznych.

Badania rynku zatrudnienia i potrzeb społecznych potwierdzają jednak, że nadal będą potrzebni eksperci i specjaliści z różnych obszarów informatyki i jej zastosowań. Dlatego duże znaczenie należy przywiązywać do przygotowania uczniów ze szkół, by w przyszłości mogli świadomie wybrać studia informatyczne i karierę zawodową związaną z informatyką.

Warto uwzględnić także, że poszerza się gama zawodów określanych mianem **IT Profession**, czyli zawodów związanych z profesjonalnym wykorzystywaniem zastosowań informatyki i technologii informacyjno-komunikacyjnej. Pracownicy tych zawodów albo są informatykami z wykształcenia, albo najczęściej nie kończyli studiów informatycznych, jednak muszą profesjonalnie posługiwać się narzędziami informatycznymi. Do IT

Profession zalicza się np. specjalistów z zakresu bioinformatyki, informatyki medycznej, telekomunikacji, genetyki itp. – wszyscy oni muszą umieć „programować" swoje narzędzia informatyczne. Informatyk ich w tym nie wyręczy, gdyż nie potrafi. W Stanach Zjednoczonych do IT Profession zalicza się obecnie ponad 40 zawodów, w których profesjonalnie są wykorzystywane zastosowania informatyki, i ta lista stale się powiększa.

2.4 CELE ZAJĘĆ INFORMATYCZNYCH

Kształcenie na wydzielonych przedmiotach informatycznych w szkole było początkowo (lata 80.-90. XX wieku) skupione na prostej **alfabetyzacji komputerowej**, czyli podstawach posługiwania się komputerem i siecią. Na przełomie XX/XXI wieków alfabetyzacja została poszerzona do **biegłości komputerowej**, przygotowującej również na zmiany w technologii, by np. nie uczyć się o kolejnych wersjach pakietu Office. Dużym wyzwaniem jest oparcie kształcenia informatycznego wszystkich uczniów na idei tzw. **myślenia komputacyjnego,** czyli w oparciu o metody rozwiązywania problemów z różnych dziedzin z pomocą komputerów. Podobnie jak maszyny drukarskie przyczyniły się do upowszechnienia kompetencji w zakresie 3R (*reading, writing, arithmethic*), tak dzisiaj komputery i informatyka przyczyniają się do upowszechniania myślenia komputacyjnego, związanego z posługiwaniem się komputerem.

Myślenie komputacyjne, towarzyszące procesom rozwiązywania problemów za pomocą komputerów, można scharakteryzować następującymi cechami:

- problem jest formułowany w postaci umożliwiającej posłużenie się w jego rozwiązaniu komputerem lub innymi urządzeniami;
- problem polega na logicznej organizacji danych i ich analizie, czemu mogą służyć m.in. modele danych i symulacje modeli;
- rozwiązanie problemu można otrzymać w wyniku zastosowania podejścia algorytmicznego, ma więc postać ciągu kroków;
- projektowanie, analiza i komputerowa implementacja (realizacja) możliwych rozwiązań prowadzi do otrzymania najbardziej efektywnego rozwiązania i wykorzystania możliwości i zasobów komputera;
- nabyte doświadczenie przy rozwiązywaniu jednego problemu może zostać wykorzystane przy rozwiązywaniu innych sytuacjach problemowych.

Przestrzeganie tych etapów posługiwania się komputerem w różnych sytuacjach problemowych ma zapewnić, by rozwiązania problemów czy realizacje projektów były:

- w dobrym stylu i czytelne dla wszystkich tych, którzy interesują się dziedziną, do której należy rozwiązywany problem lub wykonywany projekt;
- poprawne, czyli zgodne z przyjętymi w trakcie rozwiązywania założeniami i wymaganiami;
- efektywne, czyli bez potrzeby nie nadużywają zasobów komputera, czasu działania, pamięci, oprogramowania, zasobów informacyjnych.

2.5 POPRAWA SYTUACJI

Świadomość, że maleje zainteresowanie uczniów studiowaniem na kierunkach technicznych, ścisłych, i przyrodniczych, w tym również na informatyce, powoduje podejmowanie różnych działań zaradczych. W Stanach Zjednoczonych powstała specjalna inicjatywa federalna pod nazwą **STEM** (Science, Technology, Engineering, Mathematics), w ramach której jest prowadzonych wiele działań, które mają na celu przynajmniej powrót do sytuacji sprzed 10 lat.

W Polsce inicjatywy tego typu można podzielić na dwie grupy, w obu przypadkach są wspierane przez fundusze UE. Z jednej strony, MNiSzW wspiera kierunki deficytowe, które z kolei oferują studentom dość wysokie stypendia (ok. 1000 zł). z drugiej strony – MEN i organy samorządowe prowadzą projekty, których celem – pod hasłem: Człowiek – najlepsza inwestycja – jest wspieranie rozwoju wiedzy i umiejętności w dziedzinach deficytowych.

Jednym z takich projektów jest **Informatyka +** (http://www.informatykaplus.edu.pl/), w ramach którego jest wygłaszany ten wykład. Weźmie w nim udział w ciągu trzech lat ponad 15 tys. uczniów ze szkół ponadgimnazjalnych z pięciu województw. Celem tego Projektu jest podwyższenie kompetencji uczniów szkół ponadgimnazjalnych z zakresu informatyki i jej zastosowań, niezbędnych do dalszego kształcenia się na kierunkach

informatycznych i technicznych lub podjęcia zatrudnienia, oraz stworzenie uczniom zdolnym innowacyjnych możliwości rozwijania zainteresowań w tym zakresie. Program ten jest formą kształcenia pozalekcyjnego, które ma służyć zarówno zwiększeniu zainteresowania uczniów pogłębionym kształceniem w zakresie współczesnych technologii informacyjno-komunikacyjnych, jak i podniesieniu ich osiągnięć w tym obszarze.

Program Informatyka + jest też przykładem działań określanych mianem ***outreach***, polegających na tym, że uczelnia wyższa wraz ze swoimi pracownikami naukowo-dydaktycznymi stara się przedstawić uczniom ze szkół swoje działania i zachęcić do podjęcia kształcenia w kierunkach reprezentowanych przez uczelnię. Ta prezentacja uczelni i kierunków kształcenia przyjmuje formę zajęć prowadzonych przez pracowników uczelni.

3 CO TO JEST INFORMATYKA

Chociaż źródeł informatyki można się doszukać w różnych dziedzinach nauki i techniki, informatyka jako dziedzina zaczęła rodzić się wraz z pojawianiem się komputerów i dzisiaj jest kojarzona z tymi urządzeniami, chociaż w ostatnich latach przechodzą one głęboką ewolucję. Można powiedzieć, że **informatyka** jest dziedziną, która zajmuje się projektowaniem, realizacją, ocenianiem, zastosowaniami i konserwacją systemów przetwarzania informacji z uwzględnieniem przy tym aspektów sprzętowych, programowych, organizacyjnych i ludzkich wraz z implikacjami przemysłowymi, handlowymi, publicznymi i politycznymi. Wspomniane systemy przetwarzania informacji na ogół bazują na rozwiązaniach komputerowych, a w ogólności – mikroprocesorowych (jak telefony komórkowe). Z kolei informacje mogą mieć najprzeróżniejszą postać. Na początku były to tylko liczby, ale z czasem stało się możliwe przetwarzanie tekstów, a później również grafiki, dźwięków i filmów.

Sam termin informatyka pojawił się w języku polskim jako odpowiednik terminu angielskiego *computer science*. Podobnie brzmią terminy w języku francuskim *informatique* i niemieckim *Infomatik*.

Nieustannie rozszerzające się zastosowania informatyki w społeczeństwie oraz zwiększenie roli komputerów w komunikacji i wymianie informacji miało wpływ na pojawienie się nowej dziedziny, technologii informacyjno-komunikacyjnej (ang. ICT – *Information and Communication Technology*), która znacznie wykracza swoim zakresem poza tradycyjnie rozumianą informatykę. Przyjmuje się, że t**echnologia informacyjno-komunikacyjna (TIK)** to zespół środków (czyli urządzeń, takich jak komputery i ich urządzenia zewnętrzne oraz sieci komputerowe) i narzędzi (czyli oprogramowanie), jak również inne technologie (takie, jak telekomunikacja), które służą wszechstronnemu posługiwaniu się informacją. TIK obejmuje więc swoim zakresem m.in.: informację, komputery, informatykę i komunikację. Technologia informacyjno-komunikacyjna jest więc połączeniem zastosowań informatyki z wieloma innymi technologiami pokrewnymi.

Warto bliżej przyjrzeć się coraz bardziej popularnemu pojęciu technologii informacyjno-komunikacyjnej, które wyłaniało się wraz z rozwojem komputerów, sieci komputerowych i oprogramowania. W języku polskim, ten termin jest wiernym odpowiednikiem określenia w języku angielskim i niesie w sobie to samo znaczenie. Wątpliwości może budzić połączenie słowa technologia (określenie związane z procesem) zwłaszcza ze słowem informacja (w tradycyjnym sensie jest to obiekt o ustalonej formie zapisu). To połączenie ma jednak głębokie uzasadnienie we współczesnej postaci informacji i w sposobach korzystania z niej. **Informacji towarzyszą bowiem nieustannie procesy i działania.** Zarówno sam obiekt – informacja, zwłaszcza umieszczona w sieci Internet w każdej chwili ulega zmianie (poszerzeniu, aktualizacji, dopisaniu powiązań, nowym interpretacjom itd.), jak i korzystanie z niej jest procesem. Nie tylko sięgamy po nią, jak po fragment zapisany w książce (np. w encyklopedii) stojącej na półce, ale – pisząc odpowiednią frazę i wydając polecenie dla komputerowego systemu wyszukiwania informacji, znajdujących się na różnych nośnikach (w tym m.in. w sieci) – uruchamiamy proces jej uformowania w wybranym zakresie i postaci.

Informatyka jest obecnie dziedziną naukową równoprawną z innymi dziedzinami, którą można studiować i w której można prowadzić badania naukowe. Studia informatyczne można podejmować na uczelniach o różnych profilach kształcenia, np. uniwersyteckim, technicznym, ekonomicznym.

W ostatnich latach coraz większą popularnością zwłaszcza w Stanach Zjednoczonych cieszy się termin ***computing***[1], który nie ma ugruntowanego odpowiednika w języku polskim. Przekłada się ten termin na **komputyka**. Informatyka, rozumiana tradycyjnie jako odpowiednik *computer science*, jest jednym z pięciu kierunków studiowania w ramach komputyki (*computing*) według standardów amerykańskich (IEEE, ACM), są to:

- *Computer Engineering* – budowa i konstrukcja sprzęt komputerowego;
- *Information Systems* – tworzenie systemów informacyjnych;
- *Information Technology* – technologia informacyjna, zastosowania informatyki w różnych dziedzinach;
- *Software Engineering* – produkcja oprogramowania;
- *Computer Science* – studia podstawowe, uniwersyteckie studia informatyczne.

Warto jeszcze przytoczyć inne sformułowania związane z określeniem, co to jest informatyka. Na ogół dotykają one wybranych aspektów tej dziedziny.

Uważa się nie bez powodów, że:

informatyka jest dziedziną wiedzy i działalności zajmującą się algorytmami.

Przy okazji warto wspomnieć, że za pierwszy algorytm uważa się algorytm Euklides podany 300 lat p.n.e, gdy jeszcze nie istniało pojęcie algorytm, a o komputerach czy maszynach liczących nikt jeszcze nie myślał. Może się wydawać, że spojrzenie na informatykę przez pryzmat algorytmów jest bardzo ograniczone. Zapewne tak, chociaż komputer jest urządzeniem, które tylko wykonuje programy, a każdy program jest zapisem jakiegoś algorytmu. A więc w tej definicji jest zawarte zarówno programowanie (jako zapisywanie algorytmów), jak i komputer (jako urządzenie do ich wykonywania). W tym określeniu można dopatrzyć się również sieci, która jest medium działającym na bazie odpowiednich algorytmów komunikacyjnych.

Ostatecznie nie byłoby informatyki, gdyby nie było komputerów. Słowo komputer pochodzi od angielskiego słowa *computer*, które w pierwszym rzędzie oznacza osobę, która wykonuje obliczenia, a dopiero na drugim miejscu jest urządzenie służące do obliczeń[2]. Zanim ten termin zadomowił się w języku polskim, komputery nazywano (matematycznymi) maszynami liczącymi. Pierwsze pojawienie się słowa *computer* w odniesieniu do urządzeń liczących to koniec XIX wieku, gdy firma o nazwie *Rapid Computer* z Chicago zaczęła wytwarzać proste urządzenia do dodawania o nazwie *comptometer*.

Nie należy jednak sprowadzać informatyki do dziedziny zajmującej się komputerami. Bardzo zgrabnie to ujął holenderski informatyk Edgar Dijkstra (znane są algorytmy Dijkstry w teorii grafów):

Informatyka jest w takim sensie nauką o komputerach,
jak biologia jest nauką o mikroskopach,
a astronomia – nauką o teleskopach.

Dobrze jest więc pamiętać, że w zajmowaniu się informatyką i korzystaniu z jej osiągnięć, komputer jest głównie narzędziem. Komputer może być jednak bardzo pomocnym narzędziem w pracy intelektualnej, wręcz partnerem w „dyskusji" z nim za pośrednictwem algorytmów. Tutaj z kolei pasują słowa innego znanego informatyka, Amerykanina Donalda Knutha:

Mówi się często, że człowiek dotąd nie zrozumie czegoś,
zanim nie nauczy tego – kogoś innego.
W rzeczywistości,
człowiek nie zrozumie czegoś naprawdę,
zanim nie zdoła nauczyć tego – komputera.

1 *Computing* określa się jako, *... any goal-oriented activity requiring, benefiting from, or creating computers. Thus, computing includes designing and building hardware and software systems for a wide range of purposes; processing, structuring, and managing various kinds of information; doing scientific studies using computers; making computer systems behave intelligently; creating and using communications and entertainment media; finding and gathering information relevant to any particular purpose, and so on. The list is virtually endless, and the possibilities are vast.* Computing Curricula 2005, ACM, IEEE, 2006. Proponowany przekład tego terminu pojawił się m.in. w publikacjach ks. Józefa Klocha i Andrzeja Walata.

2 **computer** – **1.** a person who computes **2.** a device used for computing, Webstera's New World Dictionary, Simon and Schuster, 1969.

Na zakończenie tych ogólnych rozważań o komputerach i informatyce warto jeszcze skomentować dość częste przekonanie o wszechmocy komputerów. Zapewne, komputery są w stanie szybko wykonywać polecenia, które im wydajemy. Zilustrujemy jednak w dalszej części, że istnieje wiele problemów, przy rozwiazywaniu, których komputery są bezradne. Jest jednak szansa, by usprawnić ich działanie – cała nadzieja w szybkich algorytmach, jak to zgrabnie ujął Ralf Gomory, były szef ośrodka badawczego IBM:

Najlepszym sposobem przyspieszania komputerów
jest obarczanie ich mniejszą liczbą działań.

4 KARIERY W INFORMATYCE

Zachętą do zainteresowania się jakąś dziedziną oraz obrania jej jako obszaru kariery zawodowej mogą być kariery zawodowe wybranych przedstawicieli tej dziedziny. Dzisiaj wymienia się wiele osób, które w informatyce zrobiły karierę, a miernikiem ich kariery są miejsca na liście najbogatszych osób w świecie. Wśród nich, w kolejności na tej liście od najwyżej notowanej pozycji znajdują się następujące osoby związane z różnymi obszarami nowych technologii [dane z września 2011]: Bill Gates (Microsoft), Larry Ellison (Oracle), Jeff Bezos (Amazon), Mark Zuckerberger (Facebook), Sergey Brin (Google), Larry Page (Google), Michael Dell (Dell), Steve Ballmer (Microsoft), Paule Allen (Microsoft), Steve Jobs (Apple). To są kariery z ostatnich lat i można powiedzieć, że to „kariery z kasą". Skomentujemy je jeszcze dalej – okaże się, że ta „kasa" jest pochodną pewnego pomysłu oraz konsekwencji w jego rozwijaniu i wdrażaniu. Okaże się, że droga do dobrobytu w przypadku tych osób wiedzie przez zrozumienie oraz twórcze i innowacyjne działanie w obranym kierunku.

Zanim wrócimy do wymienionych wyżej postaci, chcielibyśmy zwrócić uwagę na kariery w informatyce, które klasyfikujemy jako „kariery z klasą", a są związane z odkryciami epokowymi dla zastosowań komputerów.

4.1 KARIERY Z KLASĄ

Przedstawiamy tutaj wybrane osiągnięcia, które miały przełomowe znaczenie w rozwoju zastosowań informatyki. Chociaż osoby związane z tymi odkryciami na trwałe zapisały się w historii rozwoju myśli, a ich odkrycia należą do kanonu wiedzy informatycznej, ani w swoich czasach, ani tym bardziej teraz nie znajdziemy ich na liście najbogatszych osób czerpiących zyski ze swoich osiągnięć.

4.1.1 LOGARYTM

Zaczniemy od logarytmu, który jest wszechobecny w informatyce.

Komentarz, uwagi historyczne

W przeszłości, uzasadnieniem dla posługiwania się logarytmem, były jego własności, które legły u podstaw jego wprowadzenia do matematyki, a dokładniej – do obliczeń. Logarytm ułatwia wykonywanie złożonych obliczeń, na przykład dzięki zastąpieniu działań multiplikatywnych, takich jak mnożenie i dzielenie, przez dodawanie i odejmowanie. Nie tak dawno jeszcze, w szkołach posługiwano się tablicami logarytmicznymi, a w uczelniach i w pracy przyszli i zawodowi inżynierowie korzystali z suwaków logarytmicznych.

Odkrywcą logarytmu był matematyk szkocki John Napier (1550-1617), a suwak logarytmiczny wynalazł William Oughtred (1575-1660) w 1632 roku. Na Rysunku 1 są pokazane różne typy suwaków: prosty, cylindryczny (model Otis King), na walcu (model Fullera) i okrągły. Różnią się one długością skali, od czego zależy dokładność obliczeń – prosty ma skalę o długości 30 cm, cylindryczny – 1,5 metra, a na walcu – ponad 12 metrów.

Rok 1972 to początek agonii suwaków – zaczęły je wypierać kalkulatory elektroniczne stworzone za pomocą … suwaków. Ponad 40 milionów wcześniej wyprodukowanych suwaków stało się nagle bezużyteczne i obecnie stanowią głównie eksponaty kolekcjonerskie, jak te na ilustracjach (należą one do kolekcji autora).

Dzisiaj jednak nie można wyobrazić sobie zajmowania się informatyką, nawet na najniższym poziomie w szkole, bez przynajmniej „otarcia" się o logarytmy. Logarytm pojawia się, gdy chcemy uzyskać odpowiedź na następujące pytania:

Rysunek 1.
Suwaki logarytmiczne: prosty, cylindryczny, na walcu i okrągły

- ile należy przejrzeć kartek w słowniku, aby znaleźć poszukiwane hasło?
- ile miejsca w komputerze, a dokładniej – ile bitów, zajmuje w komputerze liczba naturalna?
- jak szybko można wykonywać potęgowanie dla dużych wartości wykładników potęg?
- ile trwa obliczanie największego wspólnego dzielnika dwóch liczb za pomocą algorytmu Euklidesa?
- a ogólniej – ile kroków wykonuje algorytm typu dziel i zwyciężaj, zastosowany do danych o n elementach?

O znaczeniu i „potędze" logarytmów i funkcji logarytmicznej w informatyce, a ogólniej – w obliczeniach decyduje szybkość wzrostu jej wartości, **nieporównywalnie mała** względem szybkości wzrostu jej argumentu, co ilustrujemy w Tabeli 1. Zauważmy, że dla liczb, które mają około stu cyfr, wartość logarytmu wynosi **tylko** ok. 333.

Tabela 1.
Wartości funkcji logarytm dla przykładowych argumentów

n	$\log_2 n$
128	7
1 024	10
65 536	16
1 048 576	20
10^{10}	ok. 34
10^{20}	ok. 67
10^{30}	ok. 100
10^{50}	ok. 167
10^{100}	ok. 333
10^{500}	ok. 1670

Szybkie potęgowanie

Podnoszenie do potęgi jest bardzo prostym, szkolnym zadaniem. Na przykład, aby obliczyć 3^4, wykonujemy trzy mnożenia 3*3*3*3. A zatem w ogólności, aby obliczyć wartość potęgi x^n tym sposobem należy wykonać n – 1 mnożeń, o jedno mniej niż wynosi wykładnik potęgi – ten algorytm będziemy nazywali **algorytmem szkolnym**. Czy ten algorytm jest na tyle szybki, by obliczyć na przykład wartość następującej potęgi:

$$x^{12345678912345678912345678912345}$$

która może pojawić przy szyfrowaniu metodą RSA (patrz p. 4.1.2) informacji przesyłanych w Internecie?

Spróbujmy obliczyć, ile czasu będzie trwało obliczanie powyższej potęgi stosując szkolny algorytm, czyli ile czasu zabierze wykonanie 12345678912345678912345678912344 mnożeń. Przypuśćmy, że dysponujemy superkomputerem, czyli obecnie najszybszym komputerem. Taki komputer w 2011 roku działał z szybkością ok. 1 PFlops, czyli wykonywał 10^{15} operacji na sekundę. A zatem, obliczenie powyższej potęgi będzie trwało:

12345678912345678,912345678912344 sekund;
12345678912345678,912345678912344/60 = 205761315205761,31520576131520567 minut;
205761315205761,31520576131520567/60 = 3429355253429,3552534293552534278 godzin;
3429355253429,3552534293552534278/24 = 142889802226,22313555955646889282 dób;
142889802226,22313555955646889282/365 = 391478910,2088305083823464901 1733 lat,

czyli około $4*10^8$ lat. Jeśli taki algorytm byłby stosowany do szyfrowania naszej poczty w Internecie, to nigdy nie otrzymalibyśmy żadnego listu. Tutaj trzeba dodać, że w praktycznych sytuacjach muszą być obliczane potęgi o wykładnikach, które mają kilkaset cyfr.

Istnieje wiele algorytmów, które służą do szybkiego obliczania wartości potęgi. Większość z nich korzysta, bezpośrednio lub pośrednio, z binarnej reprezentacji wykładnika. Podstawowe algorytmy szybkiego potęgowania przedstawiono w książce [5]. Tutaj krótko opiszemy algorytm wykorzystujący rekurencję.

Zauważmy, że jeśli *n* jest liczbą parzystą, to zamiast obliczać wartość potęgi x^n, wystarczy obliczyć $y = x^{n/2}$ a następnie ponieść *y* do kwadratu. Jeśli *n* jest liczbą nieparzystą, to $n - 1$ jest liczbą parzystą. A zatem mamy następującą zależność:

$$x^n = \begin{cases} 1 & \text{jeśli } n = 0, \\ (x^{n/2})^2 & \text{jeśli } n \text{ jest liczbą parzystą}, \\ (x^{n-1})x & \text{jeśli } n \text{ jest liczbą nieparzystą}, \end{cases}$$

która ma charakter **rekurencyjny** – po prawej stronie równości są odwołania do potęgowania, czyli do tej samej operacji, której wartości liczymy, ale dla mniejszych wykładników. Pierwszy wiersz w powyższej równości to tzw. **warunek początkowy** – służy do zakończenia odwołań rekurencyjnych dla coraz mniejszych wykładników, aby cały proces obliczeń zakończył się. Ta zależność ma prostą realizację w języku programowania:

```
function potega(n:integer):real;
begin
  if n = 0 then potega:=1
  else if n mod 2 = 0 then potega:=potega(n div 2)^2
       else potega:=potega(n-1)*x
```

Aby określić, ile mnożeń jest wykonywanych w tym algorytmie należy zauważyć, że kolejne wywołania rekurencyjne odpowiadają kolejnym od końca bitom w binarnym rozwinięciu wykładnika i podnosimy do kwadratu tyle razy, ile jest pozycji w reprezentacji binarnej oraz dodatkowo mnożymy przez *x*, gdy w rozwinięciu pojawia się bit 1. Liczba bitów w binarnej reprezentacji liczby *n* wynosi około $\log_2 n$ i bit równy 1 może pojawić się na każdej pozycji, a zatem w sumie jest wykonywanych nie więcej niż $2 \log_2 n$ mnożeń. Dla naszej potęgi mamy więc:

$2 \log_2 n = 2 \log_2 12345678912345678912345678912345 = 2*103,28 = 206,56,$

a zatem obliczenie przykładowej wartości potęgi, której wykładnik ma 32 cyfry wymaga wykonania nie więcej niż 206 mnożeń, co nawet na zwykłym komputerze osobistym potrwa niezauważalny ułamek sekundy. Z Tabeli 1 wynika, że obliczenie wartości potęgi dla wykładnika o stu cyfrach wymaga wykonania nie więcej niż 670 mnożeń, a dla wykładnika o 500 cyfrach – nie więcej niż 3400 mnożeń, co także potrwa ułamek sekundy na komputerze osobistym. To zadanie szybkiego potęgowania jest znakomitą ilustracją wcześniej cytowanych słów Ralfa Gomory'ego, że najlepszym sposobem przyspieszania obliczeń komputerowych jest obarczanie komputera mniejszą liczba działań, czyli prawdziwe przyspieszanie obliczeń osiągamy dzięki efektywnym algorytmom, a nie szybszym komputerom.

Przedstawiona w rekurencyjnym algorytmie potęgowania technika algorytmiczna nosi nazwę **diel i zwyciężaj**. Większość algorytmów opartych na tej technice jest bardzo efektywnych i ich złożoność na ogół wyraża się za pomocą funkcji logarytmicznej. Okazuje się, że algorytm Euklidesa, odkryty 1500 lat przed wprowadzeniem logarytmów, także bazuje na tej technice. Euklides był więc bardzo bliski wynalezienia logarytmu.

4.1.2 SZYFROWANIE

Człowiek szyfrował, czyli utajniał treści przesyłanych wiadomości, od kiedy zaczął je przekazywać innym osobom. Największym polem dla szyfrowania były zawsze wiadomości mające związek z obronnością i bezpieczeństwem, a także z prowadzonymi działaniami bojowymi.

Terminem **kryptografia** określa się utajnianie znaczenia wiadomości. **Szyfr**, to ustalony sposób utajniania (czyli **szyfrowania**) znaczenia wiadomości. Z kolei, łamaniem szyfrów, czyli odczytywaniem utajnionych wiadomości, zajmuje się **kryptoanaliza**. Kryptografia i kryptoanaliza to dwa główne działy **kryptologii**, nauki o utajnionej komunikacji.

Wielokrotnie w historii ludzkości szyfrowanie miało istotny wpływ na bieg wydarzeń. Najbardziej spektakularnym przykładem jest chyba historia rozpracowania niemieckiej maszyny szyfrującej Enigma, dzięki czemu – jak utrzymują historycy – II Wojna Światowa trwała 2-3 lata krócej. Dużą w tym rolę odegrali polscy matematycy: Marian Rejewski, Jerzy Różycki i Henryk Zygalski.

Obecnie, wraz z ekspansją komputerów i Internetu, coraz powszechniej dane i informacje są przechowywane i przekazywane w postaci elektronicznej. By nie miały do nich dostępu nieodpowiednie osoby, szyfrowane są zarówno dane przechowywane w komputerach, jak i tym bardziej dane i informacje przekazywane drogą elektroniczną, np. rozmowy telefoniczne czy operacje bankowe wykonywane z automatów bankowych. Szyfrowanie danych i wiadomości jest więc niezbędnym elementem dobrze zabezpieczonych systemów komputerowych.

Pojawianie się coraz silniejszych komputerów powoduje realne zagrożenie dla przesyłania utajnionych wiadomości. Kryptoanalityk może skorzystać z mocy komputera, by prowadzić analizę kryptogramów metodą prób i błędów. Co więcej, z ekspansją komunikacji najpierw telefonicznej, a obecnie – internetowej wzrosła do olbrzymich rozmiarów liczba przesyłanych wiadomości. Państwa, instytucje, a także pojedynczy obywatele chcieliby mieć gwarancję, że system wymiany wiadomości może zapewnić im bezpieczeństwo i prywatność komunikacji.

Szyfrowanie wiadomości polega na zastosowaniu odpowiedniego algorytmu szyfrowania. Algorytmy szyfrowania są powszechnie znane, a o ukryciu wiadomości decyduje **klucz szyfrowania**, niezbędny do uruchomienia algorytmu szyfrowania i deszyfrowania. Wymienianie się kluczami szyfrowania między nadawcą i odbiorcą wiadomości zawsze stanowiło najsłabszy punkt procedury szyfrowania, klucz może zostać przechwycony. Na przykład Marian Rejewski osiągnął pierwsze sukcesy przy deszyfracji Enigmy, korzystając z faktu, że klucz do zaszyfrowanej wiadomości był powtarzany na początku wiadomości.

W połowie lat siedemdziesiątych pojawiła się sugestia, że wymiana klucza między komunikującymi się stronami być może nie jest konieczna. Tak zrodził się pomysł **szyfru z kluczem publicznym**. Działanie odbiorcy i nadawcy utajnionych wiadomości w przypadku szyfrowania z kluczem publicznym jest następujące:

1. Odbiorca wiadomości tworzy parę kluczy: publiczny i prywatny i ujawnia klucz publiczny, np. zamieszcza go na swojej stronie internetowej.
2. Ktokolwiek chce wysłać zaszyfrowaną wiadomość do odbiorcy szyfruje ją jego kluczem publicznym, zaś tak utworzony kryptogram może odczytać jedynie odbiorca posługując się swoim kluczem prywatnym.

Pierwszy szyfr z kluczem publicznym opracowali w 1977 roku Ronald **R**ivest, Adi **S**hamir i Leonard **A**dleman z MIT i od inicjałów ich nazwisk pochodzi jego nazwa **szyfr RSA**. Ten szyfr został wykorzystany w komputerowej realizacji szyfrowania z kluczem publicznym, zwanej **szyfrem PGP** (ang. *Pretty Good Privacy*), który jest powszechnie stosowany w Internecie. Na przykład uczestnicy zawodów olimpiady informatycznej wysyłają rozwiązania zadań szyfrowane kluczem publicznym dostępnym na stronie olimpiady, a jury olimpiady je rozszyfrowuje stosując sobie tylko znany klucz prywatny.

4.1.3 KOMPRESJA

Możliwość zapisania w pamięci komputera całej książki spowodowała chęć zapisania całych bibliotek. Możliwość pokazania na ekranie monitora dobrej jakości obrazu rozwinęła się do rozmiarów całego filmu. Można odnieść wrażenie, że korzystanie z pamięci rządzi się pewną odmianą prawa Parkinsona odnoszącą się do pamięci komputerów: *Dane zajmują zwykle całą pamięć możliwą do wypełnienia*.

Alternatywą dla powiększenia pamięci jest **kompresja danych**, czyli minimalizowanie ich objętości przez reprezentowanie w zwięzłej postaci. Gwałtowny rozwój metod i form komunikowania się nie byłby możliwy bez ciągłego ulepszania metod kompresji danych. Odnosi się to zarówno do tradycyjnych form wymiany informacji, takich jak: faks, modem czy telefonia komórkowa, jak i do wymiany informacji za pośrednictwem sieci Internet, w tym zwłaszcza do wymiany informacji multimedialnych. Proces kompresji i dekompresji danych jest często automatycznie (bez wiedzy użytkownika) wykonywany przez komputer. Użytkownik zauważa jedynie, że jego dysk może więcej pomieścić lub pewne operacje transferu danych są wykonywane szybciej. Zazwyczaj czas zużyty na kompresję lub dekompresję danych jest rekompensowany zwiększoną szybkością transferu skompresowanych danych.

Kompresja danych jest możliwa dzięki wykorzystaniu pewnych własności danych, na przykład często powtarzające się fragmenty można zastępować umownym symbolem lub im częściej jakiś fragment występuje tym mniejsza (krótsza) powinna być jego reprezentacja. Kompresja polega na zastosowaniu **algorytmu kompresji**, który tworzy reprezentację danych, wymagającą mniejszej liczby bitów. Każdemu algorytmowi kompresji odpowiada **algorytm dekompresji** (**rekonstrukcji**), który służy do zamiany skompresowanej reprezentacji danych na dane oryginalne. Tę parę algorytmów zwykło się nazywać algorytmem kompresji.

Historia kompresji sięga wiele lat przed erą komputerów. Ideę oszczędnego reprezentowania informacji odnajdujemy w połowie XIX wieku, gdy **Samuel Morse** (1791-1872) wynalazł **telegraf**, mechaniczne urządzenie do przesyłania wiadomości i posłużył się przy tym specjalnym alfabetem, znanym jako **alfabet Morse'a**, który umożliwia kodowanie znaków w tekście za pomocą dwóch symboli – kropki i kreski. Najważniejszą cechą tego alfabetu jest kodowanie najczęściej występujących znaków w tekście za pomocą możliwie najkrótszych kodów, np. kodem litery E jest kropka, a kodem litery T jest kreska, gdyż są to dwie najczęściej występujące litery w tekstach w języku angielskim. Ponieważ w telegrafie wysyłanie tekstu polega na przekazaniu kluczem kodów kolejnych znaków z tekstu, alfabet Morse'a znacznie zmniejszał liczbę znaków (kropek i kresek) potrzebnych do wysłania wiadomości.

Wadą alfabetu Morse'a jest to, że kody niektórych liter są częścią kodów innych liter, np. każdy kod zaczynający się od kropki zawiera na początku kod litery E. To powoduje, że w tekstach w kodzie Morse'a potrzebny jest dodatkowy znak oddzielający kody kolejnych liter. Tej wady nie ma **kod Huffmana**, zaproponowany w 1952 roku przez **Davida Huffmana** w jego pracy magisterskiej. W tym kodzie również często występujące znaki mają krótkie kody, ale żadan kod nie jest początkiem innego kodu. Kodowanie w tym kodzie nie wymaga więc dodatkowego znaku oddzielającego litery.

Na przykład słowo abrakadabra ma w kodzie Huffmana postać: 00101011001011001000010101100, czyli zamiast 88 bitów w kodzie ASCII wystarczy 29 bitów w kodzie Huffmana.

Algorytm Huffmana jest wykorzystywany w wielu profesjonalnych metodach kompresji tekstu, obrazów i dźwięków, również w połączeniu z innymi metodami. Redukcja wielkości danych przy stosowaniu tego algorytmu wynosi około 50% (w przypadku obrazów i dźwięków kodowane są nie same znaki, ale różnice między kolejnymi znakami).

4.1.4 POCZĄTKI KOMPUTERÓW ELEKTRONICZNYCH

Za jednego z ojców komputerów elektronicznych uważa się **Claude'a Shannona** (1916-2001), który w swojej pracy magisterskiej zaproponował realizację operacji algebry Boole'a w postaci układów przełączających. Jego praca jest uznawana za najważniejszą pracę dyplomową XX w.

Shannon był iście renesansowym człowiekiem. Twórca teorii informacji i propagator systemu binarnego do zapisywania obrazów i dźwięku, zafascynowany sztuczną inteligencją zaprojektował pianino do odtwarzania

zaprogramowanych utworów muzycznych oraz samouczącą się mysz, by znajdowała drogę w labiryncie, twórca komputera szachowego MANIAC.

4.1.5 HISTORYCZNE PROCESORY TEKSTU

Niewiele osób jest świadomych, że dzisiejsze edytory tekstu i klawiatury do wprowadzania tekstów mają wiele elementów wspólnych z wcześniejszymi, mechanicznymi procesorami tekstu, czyli z maszynami do pisania. Faktycznie, popularna klawiatura komputerów, zwana często QWERTY, została po raz pierwszy użyta pod koniec XIX w maszynach do pisania i od ponad pół wieku jest standardową klawiaturą do komunikacji z komputerem, a obecnie również z innymi urządzeniami elektronicznymi, takimi jak smartfony czy tablety. Z klawiaturą QWERTY nie jest kojarzony z nazwiska żaden wynalazca, można jedynie mówić o pierwszych firmach produkujących maszyny do pisania, które używały tych klawiatur, takich jak: Sholes & Glidden, Remington, Underwood i inne.

Pomysł pisania tekstów różnymi czcionkami pojawił się również na początku rozwoju maszyn do pisania. Realizowano go w postaci wymiennych głowic z różnymi czcionkami, które łatwo wymieniało się podczas pisania. Ten wynalazek również nie ma swojego autora.

Na Rysunku 2 jest pokazana maszyna Blickensderfer wyprodukowana w latach 90. XIX wieku w Stanach Zjednoczonych. Widoczne są wymienne głowice, przechowywane w drewnianych pudełkach. W tych maszynach stosowano odmienny typ klawiatury, którą można nazwać klawiaturą Morse'a, gdyż klawisze z najczęściej występującymi literami w tekście znajdują się blisko klawisza spacji (odstępu).

Rysunek 2.
Maszyna do pisania Blickensderfer, z wymiennymi głowicami z czcionkami oraz klawiaturą Morse'a (zdjęcie eksponatu z kolekcji maszyn autora)

4.2 KARIERY Z KLASĄ I KASĄ

W tym rozdziale krótko wspominamy o karierach informatycznych, kojarzonych obecnie przede wszystkim z olbrzymimi dochodami finansowymi. Chcielibyśmy jednak podkreślić, że w każdym przypadku te kariery są związane z wynalazkami i innowacjami w informatyce, które przyczyniły się do znaczącego rozwoju zastosowań informatyki na szeroką (globalną) skalę. Wymieniamy ponownie osoby z listy najbogatszych informatyków i przypisujemy im zasługi w rozwoju narzędzi i zastosowań informatyki.

1. Bill Gates, Steve Ballmer i Paule Allen, wszyscy z Microsoft. Microsoft jest obecnie dostawcą najpopularniejszych rozwiązań w zakresie systemów operacyjnych i sieciowych oraz pakietów użytkowych (biurowych) przeznaczonych dla komputerów osobistych typu IBM PC.
2. Larry Ellison, Oracle. Firma dostarczająca najpopularniejszy system zarządzania bazami danych.
3. Jeff Bezos stworzył firmę internetową Amazon, która na początku zajmowała się sprzedażą książek, nowych i używanych, a później także innych towarów. Była pierwszą księgarnią internetową, która oferowała e-książki dla e-czytników Kindle, które także sprzedawała. Okazało się to strzałem w dziesiątkę, gdyż ludzie zaczęli kupować więcej e-książek niż książek papierowych.

4. Mark Zuckerberger to twórca najpopularniejszego serwisu społecznościowego Facebook, ten serwis dostarcza coraz to nowych usług służących do komunikacji w różnych grupach jego użytkowników.
5. Sergey Brin i Larry Page stworzyli wyszukiwarkę Google a ich celem jest skatalogowanie wszystkich zasobów informacji (również tych papierowych po digitalizacji) i udostępnienie ich wszystkim użytkownikom sieci. W wyszukiwarce Google zrealizowano nowatorskie algorytmy indeksowania i wyszukiwania informacji, które są przykładem implementacji algorytmów kombinatorycznych na olbrzymią skalę.
6. Michael Dell to jeden z największych producentów komputerów osobistych marki Dell.
7. Steve Jobs i Steve Wozniak twórcy firmy Apple w 1983 roku, producenci pierwszych komputerów osobistych. Dzisiaj Steve Jobs jest jednym z czołowych innowatorów w dziedzinie powszechnych zastosowań technologii komputerowej i informacyjno-komunikacyjnej. Do flagowych produktów firmy Apple należą ostatnio: iPod. iPhone i iPad. Produkty tej firmy charakteryzują się wysoką jakością rozwiązań technologicznych, niezmiernie przyjaznym interfejsem, jak i nienagannym projektem użytkowym. Wszystkie elementy produktów firmy Apple, techniczne i programistyczne w tym również sieciowe, wychodzą „spod jednej igły" głównego pomysłodawcy. Firma Apple jest niedoścignionym wzorem dla innych firm.

Obok księgarni internetowej Amazon istnieje wiele internetowych serwisów aukcyjnych, takich jak eBay i Allegro.

Najpopularniejszym serwisem społecznościowym w Polsce był przez pewien okres serwis Nasza klasa (obecnie nk), utworzony przez studentów Instytutu Informatyki Uniwersytetu Wrocławskiego. Jego celem było „umożliwienie użytkownikom odnalezienie osób ze swoich szkolnych lat i odnowienie z nimi kontaktu". Obecnie jest to jeden z serwisów społecznościowych. Główny pomysłodawca tego serwisu, Maciej Popowicz, uczynił ten serwis przedmiotem swojej pracy magisterskiej.

Dużą popularnością cieszy się serwis społecznościowy Twitter, w którym jest udostępnione mikroblogowanie, polegające na wysyłaniu i odczytywaniu krótkich wiadomości, tzw. *twittów*.

Warto wspomnieć jeszcze o telefonii internetowej Skype, dzięki której jest możliwa darmowa komunikacja *on-line* między dowolnymi komputerami podłączonymi do sieci.

Jak napisaliśmy wcześniej, informatyczne kariery „z kasą" są rezultatem realizacji wynalazków, pomysłów i innowacyjnych rozwiązań, konsekwentnego ich wdrażania i rozwijania. Droga do dobrobytu twórców tych rozwiązań wiedzie przez zrozumienie oraz twórcze i innowacyjne działanie w obszarze rozwiązań informatycznych, które mają swoje społeczne oddziaływanie na dużą skalę.

5 WYZWANIA

W tym rozdziale przedstawiamy kilka problemów i wyzwań, które stoją przed informatyką i informatykami. Wyzwania te wyznaczają kierunek innowacyjnych działań na polu informatyki, zarówno w zakresie rozwiązań teoretycznych, jak i praktycznych. Wiele z tych problemów to trudne wyzwania, które dotychczas opierały się wszelkim próbom rozwiązania. To nie znaczy jednak, że nie istnieją dla nich rozwiązania. Być może jest potrzebne odnowione spojrzenie, nowatorska metoda, całkiem nietypowe podejście. Historia pokazała – niektóre takie przypadki prezentujemy wcześniej – że droga do odkryć i innowacji bywa na ogół bardzo nietypowa i mogą na nią natknąć się osoby nie obarczone olbrzymim zasobem wiedzy, wręcz nowicjusze. A zatem

unlock you potencial!

odblokuj swoje możliwości.

5.1 WSPÓŁPRACA W SIECI

Dla wielu nierozwiązanych, trudnych lub bardzo złożonych problemów obliczeniowych istnieją w sieci serwisy, których celem jest utworzenie społeczności sieciowych zajmujących się rozwiązywaniem takich problemów. Wymieńmy tutaj najpopularniejsze z nich.

1. Szukanie dużych liczb pierwszych – liczb Mersenea – http://www.mersenne.org/ – na znalazcę kolejnej dużej liczby pierwszej czeka nagroda 50 000/100 000 $. Ostatnie duże liczby pierwsze znajdowali studenci

na swoich komputerach osobistych. Na podanej stronie można zapoznać się z projektem (istnieje wersja po polsku) i pobrać oprogramowanie do swojego komputera, które w wolnym czasie będzie wykonywać postawione mu przez serwis poszukiwanie.

Obecnie największa liczba pierwsza jest równa $2^{43112609} - 1$. Liczba ta ma 12 978 189 cyfr. Zapisanie jej w edytorze teksu (75 cyfr w wierszu, 50 wierszy na stronie) zajęłoby 3461 stron.

2. Folding@Home – badanie proces zwijania białek – projekt prowadzony przez Uniwersytet Stanforda w Stanach Zjednoczonych. Projekt ma na celu zbadanie mechanizmów, które powodują choroby Alzheimera, Parkinsona i BSE (szalonych krów). Jest to największy projekt obliczeń rozporoszonych (czyli przebiegających w różnych komputerach, koordynowanych przez serwer projektu), w którym uczestniczą posiadacze Sony Play Station 3 i kompu≠terów osobistych. Moc komputerów uczestniczących w projekcie przekroczyła pięć razy moc najpotężniejszego superkomputera. Zainteresowanych odsyłamy do strony projektu w wersji polskiej http://wiki.zwijaj.pl/, gdzie można pobrać oprogramowanie na swój komputer i włączyć się do projektu.

W sieci istnieje wiele innych projektów obliczeniowych, polegających na rozproszeniu obliczeń na wiele niezależnych komputerów. W ten sposób np. NASA bada sygnały nadchodzące z kosmosu.

Innym projektem rozproszonym jest ... Wikipedia, tworzenie encyklopedii internetowej.

Jeszcze innym typem projektów są projekty związane z rozwojem wolnego oprogramowania, takiego jak systemy Linux, Moodle i inne.

Wszystkie wyżej opisane projekty są dostępne dla każdego użytkownika sieci. Zachęcamy do uczestnictwa w społecznościach, które realizują te projekty. Uczestnictwo nie polega tylko na udostępnieniu swojego komputera – jest jednocześnie okazją do aktywnego włączenia się w życie społeczności internetowych zajmujących się wybranym obszarem badań i działań.

5.2 KILKA TRUDNYCH PROBLEMÓW

Podamy tutaj kilka dość prostych problemów, dla których znalezienie rozwiązania nastręcza trudności nawet z użyciem najszybszych komputerów. Te problemy „czekają” na lepsze metody i algorytmy rozwiazywania.

5.2.1 NAJKRÓTSZA TRASA ZAMKNIĘTA

Jednym z najbardziej znanych problemów dotyczących wyznaczania tras przejazdu, jest **problem komiwojażera**, oznaczany zwykle jako **TSP**, od oryginalnej nazwy **Travelling Salesman Problem**. W tym problemie mamy dany zbiór miejscowości oraz odległości między nimi. Należy znaleźć drogę zamkniętą, przechodzącą przez każdą miejscowość dokładnie jeden raz, która ma najkrótszą długość.

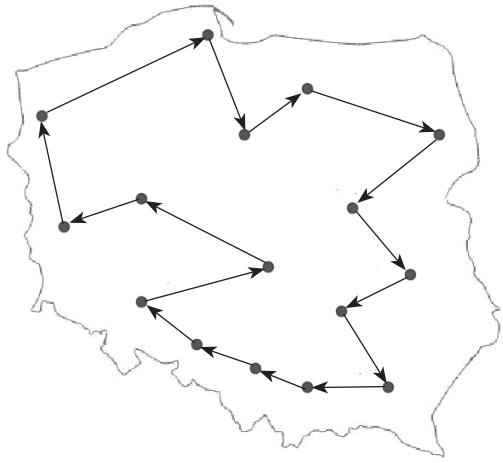
Przykładem zastosowania problemu TSP może być zadanie wyznaczenia najkrótszej trasy objazdu kraju przez prezydenta lub premiera po wszystkich stolicach województw (stanów – w Stanach Zjednoczonych, landów – w Niemczech itp.). Na tej trasie, prezydent wyjeżdża ze stolicy kraju, ma odwiedzić stolicę każdego województwa dokładnie jeden raz i wrócić do stolicy kraju.

Na Rysunku 3 jest przedstawiona jedna z możliwych tras, nie ma jednak pewności, czy jest ona najkrótsza. Obsługa biura prezydenta może jednak chcieć znaleźć najkrótszą trasę. W tym celu postanowiono generować wszystkie możliwe trasy – zastanówmy się, ile ich jest. To łatwo policzyć. Z Warszawy można się udać do jednego z 15 miast wojewódzkich. Będąc w pierwszym wybranym mieście, do wyboru mamy jedno z 14 miast. Po wybraniu drugiego miasta na trasie, kolejne miasto można wybrać spośród 13 miast i tak dalej. Gdy osiągamy ostatnie miasto, to czeka nas tylko powrót do Warszawy. A zatem wszystkich możliwych wyborów jest: 15*14*13*...*2*1. Oznaczmy tę liczbę następująco:

$$15! = 15*14*13*...*2*1$$

a ogólnie

$$n! = n*(n-1)*(n-2)*...*2*1$$



Rysunek 3.
Przykładowa trasa przejazdu prezydenta po stolicach województw

Oznaczenie *n*! czytamy **„n silnia”**, a zatem *n*! jest iloczynem kolejnych liczb całkowitych, od jeden do *n*. Wartości tej funkcji dla kolejnych n rosną bardzo szybko, patrz Tabela 2.

Tabela 2.
Wartości funkcji *n*!

n	*n*!
10	3628800
15	$1.30767*10^{12}$
20	$2.4329*10^{18}$
25	$1.55112*10^{25}$
30	$2.65253*10^{32}$
40	$8.15915*10^{47}$
48	$1.24139*10^{61}$
100	$9.3326*10^{157}$

Z wartości umieszczonych w Tabeli 2 wynika, że posługując się superkomputerem w realizacji naszkicowanej metody, służącej do znalezienia najkrótszej trasy dla prezydenta Polski, otrzymanie takiej trasy zabrałoby mniej niż sekundę. Jednak w olbrzymim kłopocie znajdzie się prezydent Stanów Zjednoczonych chcąc taką samą metodą znaleźć najkrótszą trasę objazdu po stolicach wszystkich kontynentalnych stanów (jest ich 49, z wyjątkiem Hawajów). Niewiele zmieni jego sytuację przyspieszenie superkomputerów.

Znane są metody rozwiązywania problemu komiwojażera szybsze niż naszkicowana powyżej, jednak problem TSP pozostaje bardzo trudny. W takich przypadkach często są stosowane metody, które służą do szybkiego znajdowania rozwiązań przybliżonych, nie koniecznie najkrótszych. Jedna z takich metod, zwana **metodą najbliższego sąsiada**, polega na przejeżdżaniu w każdym kroku do miasta, które znajduje się najbliżej miasta, w którym się znajdujemy. W rozwiązaniu naszego problemu tą metodą, pierwszym odwiedzonym miastem powinna być Łódź, później Kielce, Lublin, Rzeszów, .., a nie jak na Rysunku 3 mamy Lublin, Rzeszów, Kraków ..., a Łódź gdzieś w trakcie podróży. Trasa otrzymana metodą najbliższego sąsiada jest krótsza niż trasa naszkicowana na Rysunku 3. W ogólnym przypadku ta metoda nie gwarantuje, że zawsze znajdzie najkrótszą trasę.

5.2.2 ROZKŁAD LICZBY NA CZYNNIKI PIERWSZE

Liczby pierwsze stanowią w pewnym sensie „pierwiastki" wszystkich liczb, każdą bowiem liczbę całkowitą można jednoznacznie przedstawić, z dokładnością do kolejności, w postaci iloczynu liczb pierwszych. Na przykład, 4 = 2*2; 10 = 2*5; 20 = 2*2*5 = 2*5*2 = 5*2*2; 23 = 23; dla liczb pierwszych te iloczyny składają się z jednej liczby.

Matematycy interesowali się liczbami pierwszymi od dawna. Pierwsze spisane rozważania i twierdzenia dotyczące tych liczb znajdujemy w działach Euklidesa. Obecnie liczby pierwsze znajdują ważne zastosowania w kryptografii, m.in. w algorytmach szyfrujących. Do najważniejszych pytań, problemów i wyzwań, związanych z liczbami pierwszymi, należą następujące zagadnienia, które krótko komentujemy:

1. Dana jest dodatnia liczba całkowita *n* – czy *n* jest liczbą pierwszą (złożoną)?

   Ten problem ma bardzo duże znaczenie zarówno praktyczne (w kryptografii), jak i teoretyczne. Dopiero w 2002 roku został podany algorytm, który jednak jest interesujący głównie z teoretycznego punktu widzenia. Jego złożoność, czyli liczba wykonywanych operacji, zależy wielomianowo od rozmiaru liczby *n*, czyli od liczby bitów potrzebnych do zapisania liczby *n* w komputerze (równej $\log_2 n$). „Słabą" stroną większości metod, które udzielają odpowiedzi na pytanie: „czy *n* jest liczba pierwszą czy złożoną" jest udzielanie jedynie odpowiedzi „Tak" lub „Nie". Na ogół najszybsze metody dające odpowiedź na pytanie, czy *n* jest liczbą złożoną, w przypadku odpowiedzi „Tak" nie podają dzielników liczby *n* – dzielniki jest znacznie trudniej znaleźć niż przekonać się, że liczba ma dzielniki.

2. Dana jest dodatnia liczba całkowita *n* – rozłóż *n* na czynniki pierwsze.

   Ten problem ma olbrzymie znaczenie w kryptografii. Odpowiedź „Nie" udzielona na pytanie nr 1 nie pomaga w jego rozwiązaniu. Dysponujemy jednak prostym algorytmem, który polega na dzieleniu *n* przez kolejne liczby naturalne i wystarczy dzielić tylko przez liczby nie większe niż $\sqrt{n}$, gdyż liczby *n* nie można rozłożyć na iloczyn dwóch liczb większych od $\sqrt{n}$. Algorytm ten ma bardzo prostą postać:

```
var i,n:integer;
i:=2;
while i*i <= n do begin
   if n mod i = 0 then return 1;  {n jest podzielne przez i}
   i=i+1
end;
return 0    {n jest liczba pierwszą}
```

   Ten fragment programu zwraca 0, jeśli *n* jest liczbą pierwszą, i 1, gdy *n* jest liczbą złożoną. W tym drugim przypadku znamy także podzielnik liczby *n*.

   Liczba operacji wykonywanych przez powyższy program jest w najgorszym przypadku (gdy *n* jest liczbą pierwszą) proporcjonalna do $\sqrt{n}$, a więc jeśli $n = 10^m$, to wykonywanych jest $10^{m/2}$ operacji. Zatem są niewielkie szanse, by tym algorytmem rozłożyć na czynniki pierwsze liczbę, która ma kilkaset cyfr, lub stwierdzić, że się jej nie da rozłożyć.

   Rozkładem liczby złożonej na czynniki pierwsze mogą być zainteresowani ci, którzy starają się złamać szyfr RSA. Wiadomo, że jedna z liczb tworzących kluch publiczny i prywatny jest iloczynem dwóch liczb pierwszych. Znajomość tych dwóch czynników umożliwia utworzenie klucza prywatnego (tajnego). Jednak ich wielkość – są to liczby pierwsze o kilkuset cyfrach (200-300) stoi na przeszkodzie w rozkładzie *n*. Żaden z istniejących algorytmów, przy obecnej mocy komputerów, nie umożliwia rozkładania na czynniki pierwsze liczb, które mają kilkaset cyfr. Szyfr RSA pozostaje więc nadal bezpiecznym sposobem utajniania wiadomości, w tym również przesyłanych w sieciach komputerowych. Co więcej, wzrost mocy komputerów w najbliższej przyszłości nie jest w stanie tego zmienić.

3. Dana jest dodatnia liczba całkowita *m* – znajdź wszystkie liczby pierwsze mniejsze lub równe *m*.

To zadanie zyskało swoją popularność dzięki algorytmowi pochodzącemu ze Starożytności, który jest znany jako **sito Eratosthenesa**. Generowanie kolejnych liczb pierwszych tą metodą ma jednak niewielkie znaczenie praktyczne i jest uznawane raczej za ciekawostkę.

4. Znajdź największą liczbę pierwszą, a faktycznie, znajdź liczbę pierwszą większą od największej znanej liczby pierwszej. (Zgodnie z twierdzeniem Euklidesa, liczb pierwszych jest nieskończenie wiele, a zatem nie istnieje największa liczba pierwsza.). O tym problemie piszemy w p. 5.1.

5.3 PRAWDZIWE WYZWANIE

Z dotychczasowych rozważań wynika, że dla problemów, które chcemy rozwiązać za pomocą komputera, albo istnieje algorytm dość szybki, albo takiego algorytmu nie ma. Co to znaczy, szybki algorytm? Przyjęło się w informatyce, że algorytm jest **szybki** (a problem jest **łatwy**), jeśli liczba wykonywanych działań jest ograniczona przez wielomian od ilości danych. Na przykład porządkowanie jest problemem łatwym, bo n liczb można szybko uporządkować za pomocą co najwyżej n^2 działań lub nawet $n \log_2 n$ działań. Szybkie są też: algorytm Euklidesa, schemat Hornera i podnoszenie do potęgi. Problemy łatwe zaliczamy do klasy oznaczanej literą **P**.

Z kolei do klasy **NP** zaliczamy problemy **trudne**, czyli takie, które mają wielomianowy algorytm sprawdzania, czy rozwiązanie jest poprawne, ale nie mają wielomianowego algorytmu rozwiązywania. Do tej klasy należy na przykład tzw. decyzyjna wersja problemu komiwojażera, czyli pytanie, czy dla ustalonej liczby k istnieje droga komiwojażera o długości nie większej niż k.

Jednym z otwartych problemów jest pytanie, czy **P = NP**, czyli czy te dwie klasy problemów składają się z tych samych problemów. Równość jest mało prawdopodobna, bo po blisko 40 latach badań, dla żadnego z problemów należących do klasy **NP** nie udało się podać wielomianowego algorytmu rozwiązywania. Pozytywnym skutkiem braku równości tych klas jest to, że system kryptograficzny RSA jest bezpieczny, bo nie istnieje łatwy algorytm znajdowania klucza prywatnego dysponując kluczem publicznym.

Problem, czy **P = NP**, jest jednym z siedmiu problemów matematycznych, tak zwanych Problemów Milenijnych. Instytut Matematyczny Claya w MIT w Stanach Zjednoczonych ogłosił siedem problemów za fundamentalne dla rozwoju matematyki i za podanie rozwiązania któregokolwiek z tych problemów ustanowiono nagrodę w wysokości miliona dolarów. Jeden z tych problemów został już rozwiązany trzy lata temu. Ponad rok temu pojawił się dowód, że **P** nie jest równe **NP**, ale nie został on pozytywnie zweryfikowany.

LITERATURA

1. Cormen T.H., Leiserson C.E., Rivest R.L., *Wprowadzenie do algorytmów*, WNT, Warszawa 1997
2. Gurbiel E., Hard-Olejniczak G., Kołczyk E., Krupicka H., Sysło M.M., *Informatyka, Część 1 i 2, Podręcznik dla LO,* WSiP, Warszawa 2002-2003
3. Harel D., *Algorytmika. Rzecz o istocie informatyki*, WNT, Warszawa 1992
4. Knuth D.E., *Sztuka programowania*, Tomy 1 – 3, WNT, Warszawa 2003
5. Sysło M.M., *Algorytmy,* WSiP, Warszawa 1997
6. Sysło M.M., *Piramidy, szyszki i inne konstrukcje algorytmiczne*, WSiP, Warszawa 1998. Kolejne rozdziały tej książki są zamieszczone na stronie: http://mmsyslo.pl/Materialy/Ksiazki-i-podreczniki/Ksiazki/Ksiazka-Piramidy-szyszki-i
7. Wirth N., *Algorytmy + struktury danych = programy*, WNT, Warszawa 1980

W projekcie **Informatyka +**, poza wykładami i warsztatami, przewidziano następujące działania:

- 24-godzinne kursy dla uczniów w ramach modułów tematycznych
- 24-godzinne kursy metodyczne dla nauczycieli, przygotowujące do pracy z uczniem zdolnym
- nagrania 60 wykładów informatycznych, prowadzonych przez wybitnych specjalistów i nauczycieli akademickich
- konkursy dla uczniów, trzy w ciągu roku
- udział uczniów w pracach kół naukowych
- udział uczniów w konferencjach naukowych
- obozy wypoczynkowo-naukowe.

Szczegółowe informacje znajdują się na stronie projektu

www.informatykaplus.edu.pl