INFORMATYKA
– MÓJ SPOSÓB NA POZNANIE I OPISANIE ŚWIATA

PROGRAM NAUCZANIA INFORMATYKI Z ELEMENTAMI PRZEDMIOTÓW MATEMATYCZNO-PRZYRODNICZYCH

Informatyka – poziom rozszerzony

Algorytmy i logarytmy. O złożoności obliczeniowej

Paweł Perekietka

Człowiek - najlepsza inwestycja

KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

Warszawska Wyższa Szkoła Informatyki

UNIA EUROPEJSKA
EUROPEJSKI FUNDUSZ SPOŁECZNY

Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego

Tytuł: ***Algorytmy i logarytmy. O złożoności obliczeniowej***

Autor: ***Paweł Perekietka***

Redaktor merytoryczny: ***prof. dr hab. Maciej M. Sysło***

Materiał dydaktyczny opracowany w ramach projektu edukacyjnego
Informatyka – mój sposób na poznanie i opisanie świata.
Program nauczania informatyki z elementami przedmiotów matematyczno-przyrodniczych

www.info-plus.wwsi.edu.pl

infoplus@wwsi.edu.pl

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

Projekt graficzny: *Marzena Kamasa*

Warszawa 2013
Copyright © Warszawska Wyższa Szkoła Informatyki 2013
Publikacja nie jest przeznaczona do sprzedaży

Człowiek - najlepsza inwestycja

KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

Warszawska Wyższa Szkoła Informatyki

UNIA EUROPEJSKA
EUROPEJSKI FUNDUSZ SPOŁECZNY

Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego

SCENARIUSZ TEMATYCZNY

ALGORYTMY I LOGARYTMY. O ZŁOŻONOŚCI OBLICZENIOWEJ

INFORMATYKA – POZIOM ROZSZERZONY

OPRACOWANY W RAMACH PROJEKTU:

INFORMATYKA – MÓJ SPOSÓB NA POZNANIE I OPISANIE ŚWIATA.

PROGRAM NAUCZANIA INFORMATYKI

Z ELEMENTAMI PRZEDMIOTÓW MATEMATYCZNO-PRZYRODNICZYCH

Streszczenie

Algorytm jest procedurą, która gwarantuje osiągnięcie określonego celu. Termin algorytm pochodzi od zlatynizowanej wersji imienia arabskiego matematyka al-Khwarizmiego (IX w. n. e.).

Zwykle jest tak, że ten sam efekt można otrzymać za pomocą wielu algorytmów. Różne algorytmy mogą wymagać jednak różnego czasu na wykonanie zadania. Niektóre mają inne zalety związane z ich komputerową realizacją, np. mogą używać bardzo małej części pamięci komputera. Jak pisze Daniel Hillis: „Jedną z największych przyjemności w programowaniu komputerowym jest odkrywanie nowych, szybszych i bardziej wydajnych algorytmów do wykonywania jakichś zadań". W praktyce odkrycie nowego algorytmu może oznaczać możliwość rozwiązania problemu, który wcześniej znajdował się poza zasięgiem możliwości nawet najszybszych komputerów (zwłaszcza dla większej liczby danych).

Ponieważ algorytm może być stosowany dla różnego rozmiaru danych wejściowych i jego komputerowa realizacja może być uruchomiona na różnych komputerach, nie sposób ocenić szybkości algorytmu, mierząc po prostu czas działania programu komputerowego. Zamiast tego na ogół opisuje się szybkość algorytmu przez oszacowanie tego, w jaki sposób czas potrzebny do wykonania zadania rośnie przy zwiększaniu rozmiaru danych na wejściu algorytmu. Mówimy np.: algorytm jest (algorytmem o złożoności obliczeniowej) rzędu n^2, jeśli czas wymagany na wykonanie zadania jest proporcjonalny do kwadratu rozmiaru problemu, a faktycznie rozmiaru danych wejściowych.

Jednym z największym wyzwań stojących przed informatykami jest poszukiwanie pomysłów na szybsze algorytmy do wykonywania stawianych zadań. W praktyce oznacza to bowiem, że nieefektywny algorytm, służący rozwiązaniu złożonego problemu, może wymagać jego działania przez kilka tygodni (co jest zazwyczaj niepraktyczne), a optymalny (najlepszy z możliwych) algorytm – kilku minut.

W tym miejscu warto podkreślić, że tytuł cyklu lekcji „Algorytmy i logarytmy" jest grą słów – oba wyrazy w języku polskim składają się z tych samych liter (są anagramami – to znaczy, że po uporządkowaniu liter w każdym z wyrazów otrzymamy ten sam ciąg liter). Ściśle poprawny tytuł powinien brzmieć: Logarytmy w analizie algorytmów.

Czas realizacji

4 x 45 minut

Tematy lekcji

1. Jak Jaś parował skarpetki. O złożoności obliczeniowej (1 x 45 minut)
2. Gra w „Tak i Nie". Przeszukiwanie binarne (1 x 45 minut)
3. Dobry wojak Szwejk i rekruci. Minimum i maksimum w zbiorze. Sortowanie szybkie (2 x 45 minut)

LEKCJA NR 1

TEMAT: Jak Jaś parował skarpetki. O złożoności obliczeniowej

Streszczenie

W podstawie programowej przedmiotu informatyka w gimnazjum, wśród wymagań szczegółowych znajduje się następujące sformułowanie: „Uczeń wyjaśnia pojęcie algorytmu, podaje przykłady algorytmów rozwiązywania różnych problemów" (wymaganie 5.1.) i dalej: „Uczeń opisuje sposób znajdowania wybranego elementu w zbiorze nieuporządkowanym i uporządkowanym, opisuje algorytm porządkowania zbioru elementów" (5.5).

Kształceniu umiejętności stosowania podejścia algorytmicznego do rozwiązywania problemów i podejmowania decyzji (z wykorzystaniem komputera) poświęcony jest też cały punkt 5 wymagań szczegółowych w podstawie programowej liceum dla poziomu podstawowego.

W praktyce często uczniowie w wieku 16 lat nie mają jednak głębszego pojęcia na temat znaczenia algorytmiki w informatyce. Dlatego już w pierwszych tygodniach zajęć poziomu rozszerzonego w liceum należy przedstawić im wagę zagadnienia (algorytmika jako kamień węgielny informatyki).

Poniższa propozycja ma służyć właśnie pogłębieniu wiedzy uczniów na temat zagadnienia: algorytm i jego efektywność. Wprowadzone zostaną w szczególności pojęcia: najbardziej kosztownej operacji algorytmu oraz rzędu złożoności obliczeniowej algorytmu.

Podstawa programowa

Etap edukacyjny: III i IV, przedmiot: informatyka (poziom podstawowy i rozszerzony)

Cele kształcenia – wymagania ogólne

III. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, z zastosowaniem podejścia algorytmicznego.

Treści nauczania – wymagania szczegółowe

5. Rozwiazywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejscia algorytmicznego. Uczeń:
   - 5.2. stosuje podejście algorytmiczne do rozwiązywania problemu;
   - 5.18. oblicza liczbę operacji wykonywanych przez algorytm.

Cel

Uczeń:

- analizuje i modeluje sytuację problemową, do rozwiązania której można zastosować myślenie algorytmiczne,
- opisuje dwie różne metody do rozwiązania tego samego problemu i dostrzega ich różną efektywność,
- określa, na czym polega złożoność obliczeniowa, porównując efektywność dwóch metod,
- dostrzega zastosowanie matematyki w zliczaniu liczby operacji wykonywanych przez algorytm.

Słowa kluczowe

algorytm, dane wejściowe i ich rozmiar, najbardziej kosztowna operacja, efektywność (obliczeniowa) algorytmu, algorytm naiwny, algorytm optymalny, rząd (złożoności obliczeniowej) algorytmu

Co przygotować?

- Prezentacja „Jak Jaś parował skarpetki"

Przebieg zajęć

Wprowadzenie (5-10 minut)

1. Na początku zajęć nauczyciel podaje uczniom temat cyklu lekcji „Algorytmy i logarytmy" i pyta uczniów o znajomość terminów logarytm i logarytmowanie (jako siódmego działania algorytmicznego – drugiego, obok pierwiastkowania, z działań odwrotnych do potęgowania). Uczniowie mogą podać przykłady obliczeń z użyciem logarytmów i ich własności. Następnie nauczyciel pyta, o zastosowania logarytmów. Uczniowie wskazują przykłady zastosowań logarytmów (np. stężenie *pH* w chemii, natężenie dźwięku w fizyce). Nauczyciel stwierdza, że to działanie algebraiczne, znajduje też szerokie zastosowanie w informatyce.
2. Następnie pyta uczniów o rozumienie terminu algorytm (uczniowie powinni znać go z lekcji w gimnazjum) oraz podanie przykładów prostych algorytmów.
3. Nauczyciel informuje uczniów, że w co najmniej dwóch przykładach z lekcji, w analizie algorytmu pojawi się pojęcie logarytmu. W tym momencie może przywołać przykład przeszukiwania binarnego, zwanego również logarytmicznym (przykład algorytmu powinien być znany uczniom z gimnazjum, ale prawdopodobnie tylko pod pierwszą z nazw), a także zapowiedzieć, że rozwinięcie tego wątku będzie przedmiotem drugiej lekcji.
4. Na koniec wprowadzenia nauczyciel zwraca uwagę – jeśli uczniowie sami tego nie dostrzegli wcześniej – na ciekawą okoliczność: oba słowa (algorytm i logarytm) są... anagramami. W razie potrzeby krótko to objaśnia.

Zasadnicza część lekcji (20-25 minut)

Zasadnicza część pierwszej lekcji rozpoczyna się od opowiedzenia historii, która ma na celu zapoznanie uczniów z pojęciami związanymi z analizą efektywności algorytmów, ale w oderwaniu od algorytmów komputerowych.

1. Nauczyciel, posługując się prezentacją „Jak Jaś parował skarpetki", wprowadza w sposób poglądowy pojęcia: operacja podstawowa (elementarna) w algorytmie, najbardziej kosztowna operacja algorytmu, algorytm „naiwny" i efektywność algorytmu.
2. Następnie nauczyciel przedstawia, na czym polega określanie rzędu (złożoności obliczeniowej) algorytmu. Może tego dokonać w następujący sposób:

   - Przyjmujemy, że w koszu z praniem jest n skarpet;
   - Zauważamy, że w pierwszym algorytmie odnalezienie dwóch pasujących do siebie skarpet wymaga wyciągnięcia i włożenia z powrotem (na spód) przeciętnie połowy pozostałych skarpet.

   Mamy więc sumę (dla $n > 5$):

   $(1 + (n - 1)/2) + (1 + (n - 3)/2) +... + (1 + 3/2) + 2 = (n/2 - 1) + (n - 1 + n - 3 +... + 3)/2 + 2 = (n + 2)/2 + ((n - 1 + 3)/2 * (n/2 - 1))/2 > (n + 2)(n - 2)/8 > C * n^2$ (dla pewnego wymiernego C);

   - Sposób wyznaczenia sumy $n - 1 + n - 3 +... + 3$ (kolejnych liczb nieparzystych) nauczyciel może przedstawić w sposób geometryczny – jako pole wielokąta:

   - W drugim algorytmie każda ze skarpet jest wybierana tylko jeden raz. Algorytm jest więc rzędu n.

3. Nauczyciel wyjaśnia, jak należy rozumieć zapis typu $C * n^2$: oznacza to, że jeśli mamy dziesięć razy więcej skarpet, to pierwszy algorytm będzie potrzebował ok. 100 razy więcej czasu niż drugi (w tym przypadku ilość czasu jest po prostu wprost proporcjonalna do liczby skarpet). W tym momencie dla uczniów powinno być jasne, że drugi algorytm jest bardziej efektywnym algorytmem.

   Uwaga: Przyjmujemy w czasie zajęć, że najbardziej męczące, przynajmniej dla mięśni, jest wyciąganie skarpetek, a nie ich porównywanie przez „rzut oka", czyli tzw. subityzację.

   Pewną wątpliwość nauczyciela może budzić to, czy tak dobrany przykład nie będzie mieć ujemnego wpływu później na dostrzeganie roli porównań w przypadku algorytmów komputerowych. Doświadczenie pokazuje, że tak nie jest, o ile nauczyciel we właściwy sposób poprowadzi kolejne lekcje informatyki (na jednej z nich okaże się, że np. niektóre algorytmy sortowania są niepraktyczne, np. porządkowanie przez wstawianie, właśnie ze względu na koszt operacji innych niż porównywanie).

   Warto, aby nauczyciel i zainteresowani uczniowie wiedzieli, że liczba porównań w przypadku pierwszego algorytmu parowania skarpet jest właściwie równa liczbie wyciąganiętych skarpet, czyli jest rzędu n^2, a w drugim przypadku jest od tej liczby większa – w najgorszym przypadku, gdy górną połowę stosu skarpet stanowią skarpety różnego koloru, tj. nie do pary, a porządek skarpet w dolnej połowie stosu jest dokładnie odwrotny niż w górnej (w sensie kolorów), to liczba porównań może być oszacowana tak: $(n/2 - 1) + (n/2 + (n/2 - 1) + (n/2 - 2) +... + 1) > (n/2 + 1)/2 * n/2 = (n+2)*n/4 > C * n^2$ dla pewnego wymiernego C, czyli jest rzędu n^2.

Podsumowanie lekcji (5-10 minut)

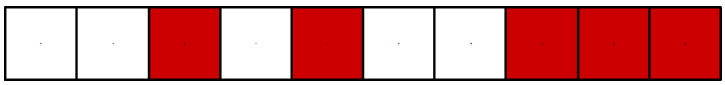
Nauczyciel może postawić pytanie: Czy jest to algorytm optymalny, tzn. najlepszy z możliwych, czyli wymagający minimalnej liczby operacji podstawowych? Uczniowie mogą wypowiedzieć się w tej sprawie. Nauczyciel powinien uwypuklić, że ścisłe wykazanie optymalności jakiegoś algorytmu to tak naprawdę zadanie matematyczne. Zwykle trudne.

Warto by nauczyciel podkreślił, że ponieważ dany algorytm może być stosowany dla różnego rozmiaru danych wejściowych i jego komputerowa realizacja może być uruchomiona na różnego rodzaju sprzęcie, nie sposób ocenić szybkości algorytmu, mierząc po prostu czas działania programu komputerowego. Zamiast tego zwykle opisuje się szybkość działania algorytmu przez oszacowanie tego, w jaki sposób czas potrzebny do wykonania zadania rośnie przy zwiększaniu rozmiaru danych na wejściu algorytmu.

Zadanie domowe

W ramach przygotowania się do trzeciej i czwartej lekcji tego cyklu uczniowie powinni zapoznać się z problemem algorytmicznym, nazywanym czasami w literaturze, *problemem polskiej flagi*.

Mamy N pionków ustawionych w jednym szeregu: niektóre są białe, a niektóre czerwone.



Zaprojektuj efektywny algorytm, zapewniający takie przestawienia pionków, aby wszystkie białe poprzedzały czerwone. Przyjmij, że operacją elementarną jest porównywanie kolorów dwóch pionków i – w razie potrzeby – zamiana dwóch pionków miejscami.

Zadaniem uczniów jest przygotowanie prezentacji – na kolejnych slajdach powinny pojawić się późniejsze etapy wykonania algorytmu (tzn. ilustracja szeregu pionków po dokonaniu kolejnego porównania).

Nauczyciel może udostępnić uczniom pierwszy slajd prezentacji (jako wzór). Powinien zaznaczyć, że na liczby zapisane w kolejnych polach uczniowie nie powinni zwracać uwagi – będzie to jasne po następnej lekcji.

Ocenianie

Nauczyciel może oceniać osiągnięcia uczniów na podstawie obserwacji ich pracy i zaangażowania na lekcji oraz na podstawie prac przygotowanych w ramach zadania domowego.

Dostępne pliki

1. Prezentacja 1 „Jak Jaś parował skarpetki"
2. Szablon prezentacji do zadania domowego 1 Problem polskiej flagi (materiały pomocnicze)
3. Zadanie domowe 1 Problem polskiej flagi

Materiały źródłowe

Źródło przykładu wykorzystanego w tym scenariuszu nauczyciel znajdzie w rozdziale 5. („Algorytmy i heurystyki") książki pt. *Wzory na krzemowej płytce*, autorstwa Daniela Hillisa (wydanej w Polsce przez wydawnictwo CIS w roku 2000).

LEKCJA NR 2

TEMAT: Gra w „Tak i Nie”. Przeszukiwanie binarne

Streszczenie

W czasie zajęć uczniowie zapoznają się z poglądową metodą analizy złożoności obliczeniowej algorytmu o złożoności logarytmicznej na przykładzie algorytmu przeszukiwania binarnego zbiorów uporządkowanych (zwanego czasami właśnie przeszukiwaniem logarytmicznym). Algorytm będzie przedstawiony jako rozwiązanie konkretnego problemu – jako strategia gry dla jednej z wersji klasycznej gry w „Tak i Nie”. To szczególny przypadek zastosowania jednej ze strategii algorytmicznych, zwanej dziej i zwyciężaj (lub zmniejsz i zwyciężaj).

Z metodycznego punktu widzenia ważne jest to, aby sprawdzić, czy uczniowie wiedzą z lekcji informatyki w gimnazjum o przeszukiwaniu liniowym (zbiorów nieuporządkowanych) oraz przeszukiwania binarnego (zbiorów uporządkowanych). Jeśli nauczyciel ma pewność, że temat jest uczniom nieznany, to scenariusz tej lekcji (jego część wstępna) powinien być odpowiednio zmodyfikowany.

Podstawa programowa

Etap edukacyjny: IV, przedmiot: informatyka (poziom rozszerzony)

Cele kształcenia – wymagania ogólne

III. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, z zastosowaniem podejścia algorytmicznego.

Treści nauczania – wymagania szczegółowe

5. Rozwiazywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejscia algorytmicznego. Uczeń:
   - 5.2. stosuje podejście algorytmiczne do rozwiązywania problemu;
   - 5.8. posługuje się metodą dziel i zwyciężaj w rozwiązywaniu problemów;
   - 5.18. oblicza liczbę operacji wykonywanych przez algorytm.

Cel

Uczeń:

- analizuje i modeluje sytuację problemową, do rozwiązania której można zastosować myślenie algorytmiczne;
- posługuje się pojęciem drzewa binarnego (i jego własności);
- określa, na czym polega logarytmiczna złożoność obliczeniowa, posługując się pojęciem drzewa binarnego (i jego własności);
- postrzega algorytm przeszukiwania binarnego jako przykład zastosowania techniki dziel i zwyciężaj.

Słowa kluczowe

drzewo binarne, drzewo decyzyjne, logarytm, przeszukiwanie binarne, metoda dziel i zwyciężaj

Co przygotować

- Egzemplarz książki *Skafander i motyl* Jean-Dominique'a Bauby lub opis książki w jednej z księgarni internetowych
- Film „Skarpety św. Mikołaja" (http://www.youtube.com/watch?v=wVPCT1VjySA) z napisami w języku polskim

Przebieg zajęć

Wprowadzenie (10-15 minut)

1. Nauczyciel rozpoczyna lekcję od przywołania książki *Skafander i motyl*. Może się posłużyć krótkim opisem książki, dostępnym np. na stronie www.ksiegarniapowszechna.pl:

   „Jean-Dominique Bauby w wyniku wylewu doznał paraliżu całego ciała. Był 8 grudnia 1995 roku. Świadomość odzyskał w styczniu. Ale była to świadomość zamurowana jak w więzieniu, w znieruchomiałym ciele, nad którym nie miał żadnej kontroli. Widział, słyszał, nie stracił powonienia – ale stracił swoje ciało. Nie mógł nawet przełykać pokarmów ani oddychać bez pomocy aparatury medycznej, a jedynym zewnętrznym wyrazem obecnego w ciele życia była mrugająca jak motyl lewa powieka i nieznaczne poruszenie głową. Jego oknem na świat stała się lewa powieka, którą wymrugał opowieść o cierpieniu, samotności, miłości i śmierci. Tworzenie książki stało się walką o istnienie. Poczucie humoru, ironia i dystans wobec otaczającego świata i własnych przeżyć spowodowały, że jego opowieść jest wzruszającą afirmacją życia. Jean-Dominique Bauby zmarł nagle 9 marca 1997 roku, trzy dni przed ukazaniem się *Skafandra i motyla*. Książka rozeszła się we Francji w nakładzie 800 000 egzemplarzy, przełożono ją na kilkadziesiąt języków".

2. Nauczyciel stawia pytanie: Co to znaczy, że autor „wymrugał" opowieść? Prosi uczniów o to, by potraktowali to pytanie jako zadanie informatyczne (algorytmiczne). Uczniowie wypowiadają się. Formułowane są różne pomysły na temat tego, jak mogła być „pisana" taka książka (np. „Liczba mrugnięć dla każdej litery odpowiada pozycji tej litery w alfabecie" lub „Rozmówca pokazuje kolejne litery alfabetu, aż do mrugnięcia"). Nauczyciel proponuje, by uczniowie w parach sprawdzili jedną z metod (można ograniczyć alfabet do zbioru liter: A, K, L, M, O, T i zadanie polega na wymruganiu w ciągu dwóch minut frazy „ALA MA KOTA").

3. Nauczyciel czyta fragment książki, który stanowi odpowiedź na pytanie:

   „E S A R I N T U L O M N P C F B V H G J Q Z Y X K W (...) To nie alfabet, to raczej lista przebojów, w której każda litera zajmuje miejsce zależne od częstości pojawiania się w języku francuskim. (...)

   System jest elementarny, Literuje mi się alfabet w kolejności »ES...« do chwili, kiedy mrugnę okiem. Wtedy mój rozmówca odpowiednią literę notuje. Rozpoczynamy operację, by ustalić kolejne litery, i jeśli nie wystąpi pomyłka, dość szybko otrzymujemy całe słowo, następnie fragmenty zdań, mniej więcej czytelnych. Taka jest teoria, sposób użycia, instrukcja obsługi. W rzeczywistości bywa różnie..." (s. 24).

   Nauczyciel pyta o rozumienie ostatnich słów: „W rzeczywistości bywa różnie...". W ten sposób podkreśla znaczenie „czynnika ludzkiego" (związanego również z ograniczoną liczbą mrugnięć, jakie można wykonać bez dużego zmęczenia).

   Następnie nauczyciel stawia problem: Czy można wyobrazić sobie sposób na szybsze napisanie książki? Chodzi przy tym o pogodzenie dwóch wymagań: jak najkrótszy czas, ale i jak najmniejsze zmęczenie mruganiem. Uczniowie wypowiadają się. Nauczyciel powinien zwrócić uwagę uczniów na to, że opisany w książce sposób przeszukiwania jest w zasadzie przeszukiwaniem liniowym, ale na uporządkowanym (w szczególny sposób) zbiorze danych.

   Nauczyciel informuje, że tematem lekcji jest poznanie własności przeszukiwania binarnego (zwanego też logarytmicznym), które pozwala na przeszukiwanie zbiorów uporządkowanych w sposób bardziej

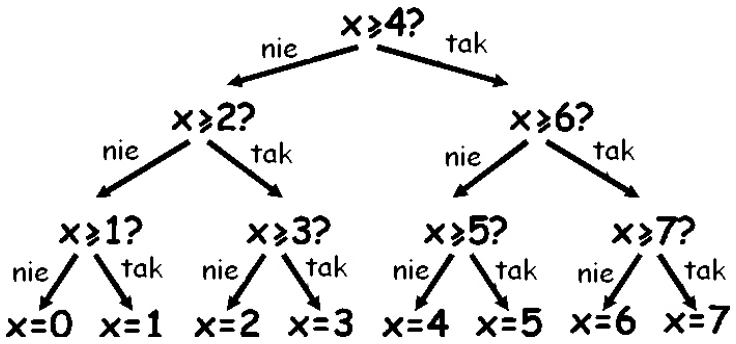
efektywny. W szczególności metoda pozwoli na to, by za pomocą pięciu pytań, na które odpowiada się „tak” lub „nie” (tej alternatywie odpowiada mrugnięcie lub jego brak) ustalić literę.

Zasadnicza część lekcji (25-30 minut)

1. Nauczyciel proponuje uczniom grę.
   - Nauczyciel prosi wybranego ucznia, aby wytypował w myślach liczbę między 1 a 100. Pozostali uczniowie zadają pytania typu: „Czy to jest liczba mniejsza od 23?”, „Czy to jest liczba 23?”lub „Czy to jest liczba większa od 23?” (aż do skutku), które pozwolą im ustalić wartość liczby. Na zadawane pytania pierwszy uczeń może odpowiadać tylko „tak” lub „nie”. Dodatkowo uczniowie powinni zliczać liczbę zadawanych pytań. Uczniowie powinni sami odkryć jak najlepszą strategię. Warto też, by powtórzyli grę (z inną liczbą) w parach lub trójkach – tak, aby nie było wątpliwości, na czym ona polega.
   - Nauczyciel – w razie potrzeby – powinien podkreślić, że do znalezienia liczby między 1 i 100 powinno wystarczyć siedem prób odgadnięcia. Pod warunkiem, że zastosujemy strategię przeszukiwania binarnego (przez połowienie), którego istota polega na tym, że po uzyskaniu każdej z odpowiedzi mamy dwa razy węższy przedział poszukiwań.
   - Następnie nauczyciel prosi o ustalenie liczby od 1 do 1000 i pyta o liczbę pytań, które powinny wystarczyć. Okazuje się, że dziesięciokrotne rozszerzenie zakresu liczb (do 1000) nie wymaga dziesięciokrotnego zwiększenia nakładu pracy – liczba potrzebnych pytań wzrasta co najwyżej o trzy (gdyż $2^7 = 128$, a $2^{10} = 1024$)!
   - Nauczyciel informuje, że gra w „Tak i Nie” jest w USA nazywana „dwadzieścia pytań” i polega na tym, że 20 pytań, na które odpowiedzią może być „tak” lub „nie”, musi wystarczyć do odgadnięcia pomyślanej rzeczy (osoby, liczby itp.).

   Nauczyciel proponuje uczniom, by ustalili zakres liczb (liczności zbioru), w którym można prowadzić grę „dwadzieścia pytań”.

   Odpowiedź wynika z następującej zależności $2^{19} = 524\,288 < 1\,000\,000 < 1\,048\,576 = 2^{20}$. Uczniowie znają pojęcie logarytmu – nauczyciel powinien zwrócić im uwagę na to, że liczba pytań stanowi logarytm o podstawie 2 z ilości liczb, wchodzących w zakres danej gry.
2. Nauczyciel rysuje na tablicy przykład tzw. drzewa decyzyjnego gry dla zakresu od 0 do 7:



Nauczyciel nazywa to drzewo drzewem binarnym (i wyjaśnia, co to znaczy).

- Nauczyciel zwraca uwagę na to, że liczba poziomów decyzyjnych w drzewie stanowi logarytm o podstawie 2 z ilości liczb z zakresu od 0 do 7 (tj. liczby liści w drzewie), tj. $3 = \log_2 2^3 = \log_2 8$.
- Następnie uczniowie proszeni są, by zastąpili etykiety w drzewie: „nie” cyfrą 0, a „tak” – 1. Okazuje się, że liczby 0, 1..., 7 (zapisane w „liściach” drzewa) zostały w sposób jednoznaczny zapisane 0-1 (zerojedynkowo), odpowiednio: 000, 001..., 111. Otrzymaliśmy więc zapis binarny (reprezentację dwójkową) tych liczb.

Warto, by w tym momencie nauczyciel przywołał nazwę przeszukiwanie binarne – w przypadku przeszukiwania ciągu liczb ze zbioru $\{0, 1, 2..., 2^k - 1\}$ (dla pewnego k) jej użycie jest dodatkowe umotywowane tym, że ciąg pytań „tak" lub „nie" jest ciągiem pytań o kolejne bity reprezentacji dwójkowej (binarnej) liczby.

- Nauczyciel proponuje utworzenie drzewa decyzyjnego, służącego do odgadywania wieku danej (pomyślanej) osoby – zakładamy, że nadal obowiązuje zasada: na każde pytanie o wiek osoby odpowiadamy „tak" lub „nie". Po krótkiej wymianie zdań, np. na temat górnego ograniczenia wieku (jako punkt wyjścia dla dyskusji można przyjąć, że to będzie 128) uczniowie pracują w parach nad pomysłami (jeśli wybrano liczbę mniejszą od 128, to drzewo nie będzie pełnym drzewem binarnym). Wybrany uczeń zapisuje rozwiązanie na tablicy.

Nauczyciel podkreśla, że drzewo, którego można by użyć do odgadywania wieku pytanej osoby, powinno być właściwie drzewem niezrównoważonym (tzn. w jego korzeniu powinno pojawić się pytanie dot. stosunkowo małej liczby lat). Wynika to z tego, że mamy różną liczbę ludzi w różnym wieku.

Końcowa faza lekcji. Podsumowanie (5-10 minut)

Nauczyciel podkreśla, że stosowane w czasie lekcji przeszukiwanie binarne jest szczegółowym przykładem zastosowania techniki algorytmicznej, znanej pod nazwą dziel i zwyciężaj (właściwie jej odmiany – zmniejsz i zwyciężaj).

Nauczyciel powinien zauważyć, że w czasie zajęć przyjęto założenie o uporządkowaniu przeszukiwanej informacji. Powinien podkreślić, że integralną częścią informacji jest jej uporządkowanie. Inaczej mówiąc, fakt uporządkowania daje możliwość zastosowania bardziej efektywnej metody rozwiązania jakiegoś problemu.

Na zakończenie nauczyciel proponuje uczniom obejrzenie filmu „Skarpety św. Mikołaja", który ukazuje inny przykład zastosowania strategii dziel i zwyciężaj (zmniejsz i zwyciężaj). Po wyświetleniu filmu nauczyciel może zapowiedzieć cykl lekcji, który będzie poświęcony różnym strategiom algorytmicznym.

Zadanie domowe

1. Uczniowie tworzą drzewo decyzyjne dla odgadywania liczb:
   a) od 0 do 15,
   b) od 0 do 19.

2. Uczniowie powinni powrócić do pytania postawionego na początku lekcji – o sposób ulepszenia metody „wymrugania" liter tworzących wyrazy w książce *Skafander i motyl*. Dla ułatwienia warto spolszczyć zadanie (uwzględniając liczbę liter w polskim alfabecie, tj. $32 = 2^5$). Zadanie uczniów polega na tym, że projektują drzewo decyzyjne dla binarnego przeszukiwania polskiego alfabetu w dwóch wersjach:

– bez uwzględniania częstości występowania liter w języku polskim,

– z uwzględnianiem częstości występowania liter w języku polskim,

tak, aby można było za pomocą pięciu pytań ustalić poszukiwaną literę.

Nauczyciel powinien wskazać wiarygodne internetowe źródło informacji o częstości występowania liter w języku polskim.

3. Przed wykonaniem zadania warto, aby zapoznali się z fragmentem filmu „Measuring information" z zasobów Akademii Khana (wystarczy obejrzenie ok. trzech minut, począwszy od 180 sekundy filmu).

https://www.khanacademy.org/math/applied-math/informationtheory/moderninfotheory/v/how-do-we-measure-information---language-of-coins-10-12

Celem (ukrytym) zadania jest sprowokowanie uczniów do zapoznania się z jednym z wymagań podstawy programowej, tj. odkrycie własności kodu Morse'a.

Ocenianie

Nauczyciel może oceniać osiągnięcia uczniów na podstawie obserwacji ich pracy i zaangażowania na lekcji oraz w oparciu o prace przygotowane w ramach zadania domowego.

Dostępne pliki

1. Zadanie domowe 2 Drzewa decyzyjne

Materiały źródłowe

1. Na koncepcję lekcji miała wpływ książka *Skafander i motyl*, której autorem jest Jean-Dominique Bauby (Słowo/Obraz Terytoria, Gdańsk 2013).
2. Informację na temat gry w „Tak i Nie" nauczyciel może znaleźć np. w scenariuszu „Gra w 20 pytań" (dostępnym w zasobach portalu csunplugged.org), w książce *Piramidy, szyszki i inne konstrukcje algorytmiczne* prof. Macieja M. Sysły (rozdz. 10, http://mmsyslo.pl/Materialy/Ksiazki-i-podreczniki/Ksiazki oraz w wielu publikacjach internetowych: np. http://www.deltami.edu.pl/temat/matematyka/gry_zagadki_paradoksy/2011/03/24/Gra_w_20_pytan/).
3. Film „Skarpety św. Mikołaja" pochodzi z zasobów projektu Computer Science Unplugged.

LEKCJA NR 3-4

TEMAT: Dobry wojak Szwejk i rekruci. Sortowanie szybkie

Streszczenie

Zajęcia mają służyć zapoznaniu uczniów z kluczowym w algorytmice problemem porządkowania (sortowania) danych. Na przykładzie analizy pewnej łamigłówki uczniowie przekonają się o tym, że umiejętność rozwiązania tego problemu może prowadzić do rozwiązania innego problemu – wyrażonego wprost w treści jakiegoś zadania (to znaczy, że np. umiejętność rozwiązywania problemu porządkowania nie jest celem samym w sobie, ale może być środkiem do celu).

Pierwsza część lekcji stanowi niezbędne wprowadzenie i jest wartościowa sama w sobie – również dlatego, że pojawiają się w niej wymagania z podstawy programowej.

Podstawa programowa

Etap edukacyjny: IV, przedmiot: informatyka (poziom rozszerzony)

Cele kształcenia – wymagania ogólne

V. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, z zastosowaniem podejścia algorytmicznego.

Treści nauczania – wymagania szczegółowe

5. Rozwiazywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejscia algorytmicznego. Uczeń:
   - 5.2. stosuje podejście algorytmiczne do rozwiązywania problemu;
   - 5.11b. opisuje i stosuje algorytmy:
     - jednoczesne znajdowanie największego i najmniejszego elementu w zbiorze: algorytm naiwny i optymalny;
     - algorytm sortowania przez wybór i szybki;
   - 5.18. oblicza liczbę operacji wykonywanych przez algorytm.

Cel

Uczeń:

- analizuje i modeluje sytuację problemową, do rozwiązania której można zastosować myślenie algorytmiczne;
- określa, na czym polega złożoność obliczeniowa, porównując efektywność dwóch metod;
- dostrzega zastosowanie matematyki (ciągów liczbowych) w zliczaniu liczby operacji wykonywanych przez algorytm.

Słowa kluczowe

algorytm naiwny, algorytm optymalny, sortowanie przez wybór, sortowanie szybkie

Co przygotować?

- Film „Dobry wojak Szwejk i rekruci”
- Prezentacja 2 „MINiMAX” do zadania domowego 3
- Film „Sorting algorithms” (http://www.youtube.com/watch?v=cVMKXKoGu_Y) z napisami w języku polskim

Przebieg zajęć

Wprowadzenie (5-10 minut)

1. Na początku zajęć nauczyciel formułuje temat lekcji „Dobry wojak Szwejk i rekruci” (oczywiście w dzienniku lekcyjnym nauczyciel powinien zapisać też podtytuł). Następnie wyświetla fragment filmu o tym tytule, w którym pojawia się treść łamigłówki, będącej zasadniczym punktem odniesienia w czasie dwóch lekcji. Nauczyciel przerywa wyświetlanie filmu w momencie, kiedy zaprezentowany zostanie w nim pomysł Szwejka (ok. 50 sekundy).
2. W czasie dyskusji uczniowie powinni wypowiedzieć swoje opinie na temat zrozumienia treści łamigłówki (rozkazu przełożonego wojaka Szwejka, a dokładniej określenia „średnia różnica wzrostu”) oraz propozycji jej rozwiązania (interpretacji rozkazu przez Szwejka). Nauczyciel stwierdza, że pomysł Szwejka będzie przedmiotem pogłębionej analizy w czasie drugiej części pierwszej lekcji.

Zasadnicza część lekcji (50-75 minut)

1. Nauczyciel prosi pięcioro uczniów, aby podeszli do tablicy i stanęli obok siebie w jednej linii (w przypadkowej kolejności), zwróceni ku reszcie klasy i weszli w rolę rekrutów. Każdy z nich otrzymuje kartkę A4 z jednym z imion lub pseudonimem (np. Krzysztof, Radek, „Kyś”, Kostek, Weronika) oraz dwucyfrową liczbą oznaczającą cyfrę dziesiątek i jedności liczby centymetrów wzrostu (np. odpowiednio: 84, 82, 78, 66, 62). (Można posłużyć się rzeczywistymi imionami uczniów i zapisać na kartkach informację o ich wzroście. Oryginalny pomysł wiąże się z tym, że na filmie stosowane są ww. imiona).
2. Pierwsze zadanie uczniów polega na tym, aby dokonali – jeśli to potrzebne – ewentualnej korekty w rozdaniu kartek-etykiet, tzn. aby porządek liczb na kartkach odpowiadał porządkowi wzrostu ich „właścicieli”. Jest to niepowtarzalna okazja, aby nauczyciel zwrócił uwagę na to, że stosują algorytm *spaghetti*, tzn. posługują się tzw. rzutem oka (subityzacja).
3. Nauczyciel stawia problem komputerowej realizacji pomysłu Szwejka: Jak ustawić rekrutów według pomysłu Szwejka? Dokładniej: Jak zrealizować komputerowo szukanie najniższego rekruta? Jak zrealizować szukanie najwyższego? Zaznacza przy tym, że za dozwolone operacje (podstawowe) uznawać będziemy porównanie wysokości dwóch osób (w praktyce: uczniowie porównywani w danym momencie powinni podnosić kartkę do góry) i ewentualną ich zamianę miejscami. W niedługim czasie uczniowie powinni sformułować pomysł na algorytm. Nauczyciel powinien zlecić powtórzenie realizacji algorytmu dla co najmniej dwóch ustawień początkowych. Uczniowie powinni zliczać liczbę wykonywanych porównań.
4. Nauczyciel wyświetla pozostałą część filmu – jest na nim przedstawiona realizacja (w postaci slajdów – na każdym zliczamy jedno porównanie) tzw. algorytmu naiwnego poszukiwania najmniejszego elementu i największego elementu w zbiorze. Następuje porównanie efektywności algorytmów: wymyślonego przez uczniów i przedstawionego na filmie (film prezentuje ich 7).
5. Nauczyciel poleca uczniom jako zdanie domowe sformułowanie i przedstawienie jego realizacji (w postaci slajdów prezentacji) bardziej efektywnego algorytmu od tego przedstawionego na filmie. Udostępnia też uczniom odpowiedni szablon prezentacji (MINiMAX).

   Uwaga: Uczniowie powinni bez większego problemu znaleźć rozwiązanie (sześć porównań) – w tym momencie (ani w czasie tej lekcji) celem nie jest sformułowanie rozwiązania w postaci ogólnej (tj. zapis wyrażenia 3n/2-2, gdzie n jest liczbą elementów z zbiorze), tylko znalezienie rozwiązania dla zbioru pięciu osób.

6. Druga część lekcji będzie stanowić pogłębioną analizę łamigłówki „Dobry wojak Szwejk i rekruci". Nauczyciel prosi wcześniej wybranych uczniów, aby wrócili i weszli ponownie w rolę rekrutów. Następnie wskazuje jeszcze jednego ucznia, którego zadaniem będzie zapisywanie na tablicy wyrażeń arytmetycznych, które pojawią się w czasie analizy.

6.1. Uczniowie ustawiają się w przypadkowej kolejności (najniższy i najwyższy nie powinni stać w tym ustawieniu na przeciwnych końcach). Nauczyciel pyta klasę o to, czym będzie średnia różnica wzrostu rekrutów. W czasie dyskusji nauczyciel zwraca uwagę na to, że wojak Szwejk miał swoją interpretację (jaką?).

Po ustaleniu zasad analizy, zapisujemy dwie sumy:

- prostą sumę różnic wzrostu (jak zrobiłby to Szwejk);
- sumę wartości bezwzględnych różnic wzrostu (jak zrobiłby to przełożony Szwejka).

Dla uczniów powinno być jasne, że dzielenie przez 5 (liczbę rekrutów) można pominąć. Wybrany uczeń zapisuje obie sumy na tablicy, np.

(Kr – Ra) + (Ky – Kr) + (We – Ky) + (Ko – We) = 2 – 6 – 16 + 4 = -16 |Kr – Ra| + |Ky – Kr| + |We – Ky| + |Ko – We| = 2 + 6 + 16 + 4 = 28

6.2. Następnie najniższy i najwyższy z rekrutów stają na przeciwnych końcach. Nauczyciel stawia pytanie: Na ile sposobów można ustawić pozostałych rekrutów? (Odp. 6). I stwierdza, że dla każdego z tych ustawień dokonana zostanie analiza sum różnic.

6.3. „Rekruci" stają w odpowiedniej kolejności, a klasa oblicza sumy. Wszystkie informacje zapisane zostają na tablicy (również informacje o ustawieniu rekrutów).

Oto przykład tabeli:

kolejność rekrutów	prosta suma	suma wartości bezwzględnych
Radek, Krzysztof, Kyś, Weronika, Kostek	–16	28
Krzysztof, Radek, Kyś, Kostek, Weronika	–22	22
Krzysztof, Radek, Kostek, Kyś, Weronika	–22	46
Krzysztof, Kostek, Kyś, Radek, Weronika	–22	54
Krzysztof, Kostek, Radek, Kyś, Weronika	–22	54
Krzysztof, Kyś, Radek, Kostek, Weronika	–22	30
Krzysztof, Kyś, Kostek, Radek, Weronika	–22	54

6.4. Następuje analiza siedmiu zapisów na tablicy (z pkt. 7.1 i 7.3) – uczniowie powinni bez problemu dostrzec, że:

- suma według Szwejka jest identyczna w każdym przypadku (za wyjątkiem pierwszego);
- obie sumy, jeśli pominąć znak, są równe tylko w jednym przypadku.

Nauczyciel pyta o wnioski. Uczniowie powinni bez problemu dostrzec, na czym polegała racja Szwejka oraz zauważyć, że intencją przełożonego Szwejka było uporządkowanie rekrutów według wzrostu – to znaczy, że rozkaz został niepotrzebnie skomplikowany (w tym znaczeniu, że rozkaz „ustawić rekrutów według wzrostu" trudniej byłoby wykonać opacznie...).

Uwaga: W przypadku pięciu rekrutów można było zastosować metodę „brutalną", tj. rozpatrzyć wszystkie przypadki (permutacje) – ścisły dowód (na poziomie matematyki szkolnej) można znaleźć np. w książce A. i M. Levitin, podanej w bibliografii.

7. Trzecia część lekcji związana jest z próbą sformułowania efektywnego algorytmu uporządkowania rekrutów.

   7.1. Nauczyciel pyta: Jak w efektywny sposób uporządkować rekrutów? Uczniowie powinni formułować głośno kilka pomysłów na metodę porządkowania (sortowania). Co najmniej jeden z nich powinien być im znany z lekcji w gimnazjum.

   7.2. Nauczyciel podsumowuje wypowiedzi uczniów – wskazuje na co najmniej jedną metodę, być może wymienioną przez uczniów, tj. porządkowanie przez wybór. Zaznacza przy tym, że jedną z wersji może być zastosowanie algorytmu jednoczesnego znajdowania najmniejszego elementu i największego elementu w zbiorze.

   7.3. Uczniowie realizują metodę porządkowania przez wybór na grupie rekrutów. Zliczają liczbę wykonywanych porównań.

   7.4. Nauczyciel prezentuje uczniom film „Sorting algorithms" (z napisami w języku polskim), który ukazuje inną metodę porządkowania (sortowania), tj. Quicksort (szybkie sortowanie), która w większości przypadków (dla dużej liczby danych) okazuje się najszybszą i jest stosowana w praktyce (w realizacjach komputerowych). Na filmie efektywność tej metody zestawiona jest z efektywnością (pracochłonnością) porządkowania przez wybór.

   Uwaga: Warto, aby uczniowie mieli w czasie lekcji możliwość posłużenia się wagą szalkową i odważnikami, podobnymi do tych na filmie – mogą je sami wcześniej przygotować w domu.

   7.5. Po obejrzeniu filmu powinna nastąpić dyskusja, której celem jest zweryfikowanie rozumienia metody. Film może być wyświetlony ponownie. Uczniowie powinni przekonać się, że algorytmy porządkowania mają złożoność obliczeniową nieliniową, tzn. liczba wykonywanych operacji nie jest wprost proporcjonalna do liczby porządkowych elementów. Cennym komentarzem nauczyciela będzie zauważenie, że algorytm porządkowania przez wybór ma niezmienną liczbę podstawowych operacji dla danego n danych. W tym celu uczniowie powinni wyznaczyć – wzorując się na obliczeniach z lekcji 1 – liczbę wykonywanych porównań (dla liczby danych n) algorytmu porządkowania przez wybór.

   Uczniowie powinni dostrzec związek między *problemem polskiej flagi* a problemem porządkowania. To znaczy – zrozumienie algorytmu rozwiązującego pierwszy problem (porządkowania kolorów) ułatwia zrozumienie algorytmu dla drugiego problemu (porządkowania liczb).

   7.6. Nauczyciel powinien podkreślić, że metoda jest pomysłem brytyjskiego informatyka Sir Charlesa Antony'ego Richarda Hoare'a (dziś emerytowanego profesora z Oxfordu) i została opublikowana w 1962 roku.

   (W Internecie można znaleźć wspomnienia naukowca – przedstawia w nich historię odkrycia, które zostało dokonane przy okazji zajmowania się problemem tłumaczenia maszynowego z języka rosyjskiego na język angielski).

Podsumowanie lekcji (10 minut)

Na zakończenie lekcji warto wyświetlić film „Qucik sort with Hungarian" (http://www.youtube.com/watch?v=3San3uKKHgg), który można potraktować jako rozwiązanie łamigłówki „Dobry wojak Szwejk i rekruci" dla dziesięciu rekrutów.

Uczniowie – jeśli czas na to pozwala – mogą zastosować metodę dla uporządkowania pięciu rekrutów. Przy tak małej liczbie danych trudno jednak dostrzec piękno i „siłę" tej metody. Warto, by nauczyciel dopowiedział, że w przypadku małej liczby danych w praktyce komputerowej stosuje się zwykle metodę porządkowania przez scalanie, która też znajduje się wśród wymagań maturalnych.

Zadania domowe

MINiMAX

1. Nauczyciel poleca uczniom jako zdanie domowe sformułowanie bardziej efektywnego (od tego przedstawionego na filmie) algorytmu jednoczesnego znajdowania najmniejszego elementu i największego elementu dla zbioru pięcioelementowego oraz przedstawienie realizacji (przebiegu) algorytmu w postaci slajdów prezentacji. Nauczyciel udostępnia uczniom odpowiedni szablon prezentacji (MINiMAX).

QUICKSORT

2. W ramach zadania domowego uczniowie powinni zapoznać się z opisem oszacowania efektywności algorytmu szybkiego sortowania (analizy „optymistycznej" złożoności obliczeniowej). Nauczyciel powinien udostępnić uczniom fragment książce Davida Gelerntera *Mechaniczne piękno*:

   „Przypuśćmy, że leży przed nami stos ponumerowanych kartek, które mamy posortować. (...) Bierzemy ze stosu pierwszą kartkę. Powiedzmy, że ma ona numer 117. Następnie przeglądamy pozostałe kartki. Jeśli numer kolejnej jest mniejszy niż 117, kładziemy ją po lewej stronie pierwszej kartki, jeśli większy – po prawej.

   Teraz nasz stos jest podzielony na trzy grupy kartek: te z numerami mniejszymi od 117, kartkę z numerem 117 i kartki z numerami większymi.

   Powtarzamy całą operację dla dwóch mniejszych stosów, lewego i prawego.

   Stara lewa grupa zostaje w ten sposób podzielona znowu na trzy grupy; to samo dotyczy starej prawej grupy.

   Znowu powtarzamy operacje dla każdej nowo powstałej grupy i tak dalej, aż każda grupa będzie się składała z pojedynczej kartki. (...)

   Spróbujmy teraz dokonać oszacowania efektywności (...).

   Usuwamy pierwszą kartkę i dzielimy pozostałe na dwie grupy: „mniejszą" i „większą". Jeśli pierwsza kartka miała przeciętny numer, to te dwie grupy będą miały po mniej więcej tyle samo kartek – około połowy kartek ze stosu będzie miało numer mniejszy od numeru pierwszej, a połowa będzie miała numer większy. Następnie powtarzamy procedurę dla kolejnych podstosów i tak dalej. Przypuśćmy, że jakimś szczęśliwym trafem za każdym razem znajdujemy na szczycie kolejnej grupy przeciętną kartkę. Ile razy musielibyśmy przeglądać rosnący szereg grup?

   Za pierwszym razem podzieliliśmy stos na dwie grupy, po około ½ *n* kartek. Za drugim razem otrzymaliśmy cztery grupy po około ¼ *n* kartek, za trzecim – osiem grup po 1/8 *n* i tak dalej. Za którym razem otrzymamy szereg grup liczących tylko jedną kartkę każda? Odpowiedź brzmi: logarytm o podstawie 2 z *n* (...). Powiedzmy, że mamy stos 16 kartek; pierwszy podział daje dwie grupy po 8, drugi – cztery grupy po 4, trzeci – osiem grup po 2, a czwarty – 16 grup złożonych z jednej kartki. Potrzebowaliśmy czterech podziałów: 2 do potęgi 4 daje 16. (...)

   Za każdym razem, gdy dokonywaliśmy kolejnego podziału stosu (na dwie, cztery, osiem grup – i tak dalej), musieliśmy sprawdzać [prawie] wszystkie *n* kartki. Quicksort jest zatem algorytmem o złożoności *n*log*n*. (...)

   Dla miliona kartek QuickSort potrzebuje około 20 milionów operacji (...). Załóżmy teraz, że nasz komputer potrafi wykonywać 10 milionów takich operacji na sekundę. Na posortowanie listy zawierającej milion pozycji QuickSort potrzebuje około 2 sekund".

   Powyższy opis dotyczy sytuacji „jeśli mamy szczęście", tzn. że efektywność spada, jeśli rozkład danych jest mniej korzystny. Istnieją jednak sposoby, aby zwiększyć szanse na zadowalający rozkład danych (przed rozpoczęciem sortowania) i można udowodnić, że średnia złożoność obliczeniowa QucikSort opisana jest również z użyciem iloczynu *n*log*n* (gdzie *n* oznacza liczbę porządkowanych danych) i z tego powodu nazywana jest czasami złożonością podkwadratową, a nawet quasi-liniową (*n*log*n* „leży bliżej" *n* niż n^2).

Uczniowie mają okazję przeanalizować przykład rozumowania matematycznego (uproszczony, tzn. nie tak formalnie zapisany, jak w podręcznikach akademickich), które ukazuje, że przynajmniej w niektórych przypadkach złożoność QuickSort jest mniejsza niż metody przez wybór.

Warto, aby wykonali w domu odpowiedni eksperyment ze stosem kart.

Sprawdzianem zrozumienia analizy algorytmu może być wyjaśnienie sensu wtrącenia „[prawie]", jakie dodano w przedostatnim akapicie cytatu.

Ocenianie

Nauczyciel może oceniać osiągnięcia uczniów na podstawie obserwacji ich pracy i zaangażowania na lekcji oraz na podstawie prac przygotowanych w ramach zadania domowego.

Dostępne pliki

1. Prezentacja 2 „MINiMAX" do zadania domowego 3
2. Zadanie domowe 3 MINiMAX
3. Zadanie domowe 4 Analiza złożoności QuickSort

Materiały źródłowe

1. Wykorzystana w tym scenariuszu łamigłówka „Dobry wojak Szwejk i rekruci" pochodzi ze zbioru *Algorithmic puzzles*, napisanej przez Anany Levitin i Marię Levitin (Oxford Press 2010).
2. Wspomnienia Hoare'a – odkrywcy QucikSort – dostępne są w Internecie (http://www.csc.lsu.edu/~gb/csc7101/Handouts/p38-shustek.pdf).
3. Elementarna analiza algorytmu QuickSort wzorowana jest na książce Davida Gelerntera, *Mechaniczne piękno. Kryterium estetyczne w informatyce* (WAB 1999).

Człowiek - najlepsza inwestycja

KAPITAŁ LUDZKI
NARODOWA STRATEGIA SPÓJNOŚCI

Warszawska Wyższa Szkoła Informatyki

UNIA EUROPEJSKA
EUROPEJSKI
FUNDUSZ SPOŁECZNY

Projekt współfinansowany przez Unię Europejską w ramach Europejskiego Funduszu Społecznego