

SCENARIUSZ LEKCJI

Algorytm wyszukiwania w zbiorze danych

Informatyka

Klasy VII i VIII

45 minut

Lekcja otwarta

Uczeń zapisuje algorytm wyszukiwania jako ciąg jednoznacznych kroków i porównuje liczbę kroków szukania po kolei oraz przez połowienie zakresu na tym samym zbiorze danych.

Czego uczy ta lekcja

Zapis algorytmu w krokach

Umiesz zapisać rozwiązanie jako ponumerowane kroki, które da się wykonać bez zgadywania, i wskazać w nich warunek zakończenia.

Wyszukiwanie po kolei i przez połowienie

Potrafisz policzyć, ile kroków zajmuje znalezienie danej dwiema metodami, i powiedzieć, kiedy połowienie zakresu się opłaca, a kiedy nie.

Potrzebny sprzęt

laptopy szkolne z przeglądarką i dostępem do internetu, po jednym na parę uczniów, projektor lub monitor do pokazu, wydruk karty pracy z listą 16 numerów inwentarza dla każdej pary, długopisy. Edytor Scratch otwiera się w przeglądarce bez zakładania konta pod adresem scratch.mit.edu/projects/editor. Jeśli szkolna sieć blokuje ten adres, część komputerową robi się w dowolnym arkuszu kalkulacyjnym zainstalowanym na laptopie, a przy braku laptopów działa wariant bez sprzętu.

Podstawa programowa

Informatyka, Szkoła podstawowa, klasy VII i VIII

Cele kształcenia, wymagania ogólne

- I. Rozumienie, analizowanie i rozwiązywanie problemów na bazie logicznego i abstrakcyjnego myślenia, myślenia algorytmicznego i sposobów reprezentowania informacji.

Treści nauczania, wymagania szczegółowe

- I.1. formułuje problem w postaci specyfikacji (czyli opisuje dane i wyniki) oraz wyróżnia kroki w algorytmicznym rozwiązywaniu problemów. Stosuje różne sposoby przedstawiania algorytmów, w tym w języku naturalnym, w postaci listy kroków; (Pełne pokrycie wątku zapisu algorytmu: specyfikacja (dane i wyniki) oraz przedstawianie algorytmu w postaci listy kroków.)
- I.2 b). wyszukiwania i porządkowania: wyszukuje element w zbiorze nieuporządkowanym oraz porządkuje elementy w zbiorze metodą przez proste wybieranie; (Uwaga na brzmienie, ważna przy rozmowie z nauczycielem. Wersja z 2024 r. (Dz.U. 2024 poz. 996) mówi tylko o wyszukiwaniu elementu "w zbiorze nieuporządkowanym". Pierwotne brzmienie z 2017 r. obejmowało zbiór uporządkowany i nieuporządkowany. Wyszukiwanie przez połowienie zakresu w zbiorze uporządkowanym nie ma dziś oparcia w podstawie dla klas VII i VIII. Wraca ono do klas VII i VIII z nową podstawą (Dz.U. 2026 poz. 378, wymaganie 1.4) oraz występuje w szkole ponadpodstawowej w zakresie rozszerzonym. Wyszukiwanie po kolei jest pokryte w pełni.)

NOWA PODSTAWA PROGRAMOWA, DZ.U. 2026 POZ. 378

1.4. stosuje podstawowe algorytmy do wyszukiwania, zliczania i porządkowania danych, w tym wyszukiwanie w ciągu uporządkowanym i nieuporządkowanym oraz porządkowanie przez wybór lub metodą bąbelkową;

Podstawa prawna: Rozporządzenie Ministra Edukacji Narodowej z dnia 14 lutego 2017 r. w sprawie podstawy programowej wychowania przedszkolnego oraz podstawy programowej kształcenia ogólnego dla szkoły podstawowej, w tym dla uczniów z niepełnosprawnością intelektualną w stopniu umiarkowanym lub znacznym, kształcenia ogólnego dla branżowej szkoły I stopnia, kształcenia ogólnego dla szkoły specjalnej przysposabiającej do pracy oraz kształcenia ogólnego dla szkoły policealnej, Dz.U. 2017 poz. 356.

Brzmienie: Załącznik nr 2 do Dz.U. 2017 poz. 356 w brzmieniu nadanym przez Dz.U. 2024 poz. 996, część KLASY VII i VIII

Lista kontrolna przed lekcją

Do odhaczenia dzień wcześniej i 10 minut przed dzwonkiem.

- ☐ Dzień wcześniej: wydrukujcie kartę pracy dla każdej pary, sprawdźcie, że tabela w zadaniu 2 ma cztery kolumny i trzy wiersze na numery 1199, 1560 i 1305.
- ☐ Dzień wcześniej: otwórzcie na szkolnym laptopie scratch.mit.edu/projects/editor i sprawdźcie, czy sieć nie blokuje adresu. Jeśli blokuje, przygotujcie arkusz kalkulacyjny albo wariant bez sprzętu.
- ☐ Dzień wcześniej: zbudujcie sami skrypt z zadania 4 karty pracy i zapiszcie go na pendrive jako wzorzec oraz jako plik z dwiema lukami dla wariantu łatwiejszego.
- ☐ Dzień wcześniej: przygotujcie 16 kartek z numerami inwentarza na wariant bez sprzętu, nawet jeśli laptopy mają działać, bo zdarza się, że część z nich nie wstaje.

- ☐ 10 minut przed lekcją: włączcie projektor, otwórzcie prezentację na slajdzie 3 z listą 16 numerów i sprawdźcie, że numery pozycji od 1 do 16 są widoczne z ostatniej ławki.
- ☐ 10 minut przed lekcją: zapiszcie w notatniku liczbę z zakresu od 1 do 1000 do gry na wejście i przygotujcie miejsce na tablicy na kreski oraz na dwa algorytmy zapisane krokami.
- ☐ 10 minut przed lekcją: zapiszcie na tablicy definicję kroku „jeden krok to jedno porównanie szukanego numeru z numerem z listy” i regułę „środek zaokrąglamy w dół”, bo bez nich wyniki par się nie zgodzą.
- ☐ 10 minut przed lekcją: sprawdźcie, że na każdym laptopie działa przeglądarka i że para ma gdzie zapisać plik projektu, bo bez konta Scratch niczego nie zapamiętuje.

Rozgrzewka na start

BEZ SPRZĘTU, 3 MIN

Powiedzcie klasie: „Zgubiłem długopis gdzieś w tej sali. Podajcie mi instrukcję szukania go tak dokładnie, żeby wykonał ją robot, który nie wie, co to długopis ani sala”. Zapiszcie na tablicy pierwsze trzy propozycje dosłownie, tak jak padły.

Czego się spodziewać: Uczniowie zaczynają od zdań typu „rozejrzyj się” albo „sprawdź pod ławkami”, a po dopytaniu „którą ławkę pierwszą, a którą drugą” sami dochodzą do tego, że instrukcja musi mieć kolejność, numerowane miejsca i moment, w którym wolno przestać szukać. To trzy cechy algorytmu, do których wrócicie na slajdzie drugim.

Przebieg lekcji

Pięć etapów, razem 45 minut.

1

Wejście, 5 min

Nauczyciel zapisuje w notatniku liczbę z zakresu od 1 do 1000 i zaprasza klasę do zgadywania. Po każdej próbie odpowiada tylko „za mało” albo „za dużo”, a na tablicy stawia kreskę za każde pytanie. Gra kończy się zwykle po mniej niż piętnastu pytaniach. Nauczyciel pyta: ile pytań trzeba by zadać, gdybyśmy sprawdzali po kolei 1, 2, 3 i tak dalej. Odpowiedź tysiąc zostaje na tablicy obok liczby kresek. Zapowiada, że przez najbliższe 40 minut klasa zapisze regułę dobrego zgadywania tak dokładnie, żeby wykonał ją ktoś, kto nic o niej nie wie.

2

Pokaz, 10 min

Nauczyciel wyświetla uporządkowaną rosnąco listę 16 numerów inwentarza ze szkolnej biblioteki: 1042, 1087, 1130, 1156, 1199, 1204, 1233, 1290, 1315, 1342, 1388, 1401, 1455, 1470, 1512, 1560. Na tablicy zapisuje dwa algorytmy jako ponumerowane kroki. Algorytm A, szukanie po kolei: 1. Ustaw pozycję na 1. 2. Porównaj numer na tej pozycji z szukany. 3. Jeśli równy, wypisz pozycję i zakończ. 4. Jeśli pozycja to 16, wypisz „nie ma” i zakończ. 5. Zwiększ pozycję o 1 i wróć do kroku 2. Algorytm B, połowienie zakresu: 1. Ustaw dolną na 1, górną na 16. 2. Jeśli dolna jest większa od górnej, wypisz „nie ma” i zakończ. 3. Ustaw środek na wynik dzielenia sumy dolnej i górnej przez 2, zaokrąglony w dół. 4. Porównaj numer ze środka z szukany. 5. Jeśli równy, wypisz środek i zakończ. 6. Jeśli numer ze środka jest mniejszy, ustaw dolną na środek plus 1, w przeciwnym razie ustaw górną na środek minus 1. 7. Wróć do kroku 2. Nauczyciel wykonuje algorytm B na głos dla numeru 1401 i dochodzi do wyniku w 2 krokach, potem dla numeru 1300, którego na liście nie ma, i pokazuje, że po 4 krokach dolna przekracza górną, więc algorytm kończy się odpowiedzią „nie ma”. Podkreśla trzy cechy dobrego zapisu: każdy krok jednoznaczny, warunek zakończenia obecny, wynik podany także wtedy, gdy danej nie ma.

3

Praca uczniów, 20 min

Część pierwsza, 8 minut na papierze. Pary wykonują oba algorytmy z tablicy dla trzech numerów: 1199, 1560 oraz 1305, którego na liście nie ma. Do tabeli w karcie pracy wpisują liczbę kroków dla algorytmu A i dla algorytmu B oraz wynik. Jeden uczeń czyta kroki na głos, drugi wykonuje je dosłownie i nie wolno mu niczego domyślić. Część druga, 12 minut na laptopach. Pary otwierają scratch.mit.edu/projects/editor i budują skrypt, w którym to komputer zgaduje liczbę pomyślaną przez ucznia z zakresu od 1 do 100. Potrzebne zmienne: dolna, gorna, strzał, proby. Środek liczy się blokiem „podłoga z” (zaokrąglenie w dół), nie zwykłym blokiem „zaokrąglaj”, bo ten zaokrągla 5,5 do 6 i liczba prób przestaje się zgadzać z algorytmem z tablicy. Szkielet skryptu jest wypisany blok po bloku w karcie pracy. Para testuje skrypt na trzech różnych liczbach i zapisuje największą liczbę prób, jaka jej wyszła. Projekt zapisuje się poleceniem Plik, a następnie Zapisz na swoim komputerze, więc konto nie jest potrzebne. Nauczyciel chodzi po sali i sprawdza przede wszystkim, czy pary poprawnie zwiężają zakres o jeden.

4

Sprawdzenie, 5 min

Nauczyciel rysuje na tablicy tabelę z trzema szukany numerami i dwiema kolumnami na liczbę kroków. Trzy pary podają swoje wyniki, klasa porównuje. Dla numeru 1199 algorytm A potrzebuje 5 kroków, a B 4, dla numeru 1560 odpowiednio 16 i 5, dla nieobecnego 1305 algorytm A wykonuje 16 kroków, bo przechodzi całą listę i dopiero po ostatnim porównaniu wypisuje „nie ma”, a B kończy po 4 krokach. Nauczyciel pyta, dlaczego przy numerze 1199 przewaga połowienia prawie znika, i doprowadza klasę do odpowiedzi, że szukana dana leży blisko początku listy. Potem pyta o zbiór tysiąca pozycji i o zbiór miliona pozycji. Klasa dochodzi do liczb: do 10 kroków przy tysiącu i do 20 kroków przy milionie, bo każdy krok zmniejsza zakres o połowę.

5

Podsumowanie, 5 min

Nauczyciel wraca do gry z początku lekcji i pyta, czy klasa zgadywała zgodnie z algorytmem B. Zapisuje wniosek w trzech zdaniach: algorytm to kroki na tyle jednoznaczne, że wykonuje je ktoś bez wiedzy o zadaniu, każdy algorytm musi mieć warunek zakończenia, a połowienie działa tylko na zbiorze uporządkowanym. Dodaje uwagę praktyczną: uporządkowanie zbioru też kosztuje czas, więc przy jednym szukaniu w krótkiej liście prostsze jest szukanie po kolei. Zapowiada kolejną lekcję o bazach danych, na której klasa zaprojektuje tabelę biblioteki, czyli zbiór, w którym takie szukanie ma sens.

Do powiedzenia na głos

„Algorytm to nie pomysł w głowie. To kroki zapisane tak dokładnie, że wykona je ktoś, kto o zadaniu nie wie nic.”

„Za chwilę Wasz kolega będzie wykonywał Wasze kroki dosłownie. Jeśli w którymś miejscu musi się domyślić, to ten krok jest źle napisany.”

„Każdy algorytm musi umieć się skończyć także wtedy, gdy szukanej danej po prostu nie ma.”

„Połowienie zakresu działa tylko na liście uporządkowanej. Na liście w przypadkowej kolejności daje złą odpowiedź, a nie wolniejszą.”

Typowe błędy uczniów i co z nimi zrobić

- Uczniowie stosują połowienie do listy w przypadkowej kolejności i dziwią się, że wynik jest błędny. Wprowadźcie krok zerowy: sprawdź, czy lista jest uporządkowana, i dopiero wtedy wybieraj algorytm.
- W zapisie algorytmu brakuje warunku zakończenia dla danej, której nie ma, więc wykonanie kręci się w kółko. Wymagajcie wprost kroku „jeśli dolna jest większa od górnej, wypisz nie ma i zakończ”.
- Pary liczą kroki różnymi sposobami, jedni porównania, drudzy podziały zakresu, i wyniki się nie zgadzają. Ustalcie na tablicy jedną definicję kroku przed rozpoczęciem pracy i trzymajcie ją do końca lekcji.
- Część uczniów zaokrągla środek w górę, część w dół, więc te same dane dają różne liczby kroków. Zapiszcie regułę zaokrąglania w dół obok algorytmu i wracajcie do niej przy sprawdzaniu.

- W Scratchu uczniowie ustawiają dolną na wartość strzału zamiast na strzał plus 1, przez co zakres przestaje się kurczyć i skrypt pyta w nieskończoność. Pokażcie to na przykładzie zakresu od 5 do 6, gdzie środek zawsze wypada na 5.

Warianty lekcji

Bez sprzętu i internetu

Nauczyciel przygotowuje 16 kartek z numerami inwentarza z listy z lekcji, układa je na ławce w kolejności rosnącej numerami do dołu i podpisuje pozycje od 1 do 16. Jeden uczeń jest wykonawcą algorytmu i może odwrócić tylko tę kartkę, którą wskazuje bieżący krok. Drugi uczeń czyta kroki na głos z kartki, a trzeci liczy odwrócone kartki, czyli kroki. Klasa przeprowadza w ten sposób oba algorytmy dla tych samych trzech numerów co w karcie pracy: 1199, 1560 i 1305 i zapisuje wyniki w tabeli na tablicy. Wniosek wychodzi identyczny, a dodatkowo widać, że wykonawca nie ma prawa podejrzec całej listy, dokładnie tak jak komputer.

Dla szybszych uczniów

Para wyznacza największą możliwą liczbę kroków połowienia dla zbiorów o wielkości 100, 1000 i 1 000 000 pozycji, zapisuje regułę słowami i sprawdza ją na swoim skrypcie, zmieniając górną granicę zakresu. Następnie dopisuje do skryptu kontrolę uczciwości: jeśli dolna przekroczy górną, a gracz nie powiedział „trafione”, program wypisuje komunikat, że podane odpowiedzi są sprzeczne. Para wyjaśnia w dwóch zdaniach, dlaczego ten sam warunek w algorytmie wyszukiwania oznacza „danej nie ma w zbiorze”.

Przy trudnościach

Uczeń pracuje na skróconej liście ośmiu numerów: 1042, 1130, 1199, 1233, 1315, 1388, 1455, 1512, i szuka tylko jednego numeru wskazanego przez nauczyciela. Kroki algorytmu B dostaje wydrukowane z dwiema lukami do uzupełnienia zamiast pełnego zapisu. W części komputerowej otwiera przygotowany przez nauczyciela plik projektu i wstawia dwa brakujące bloki, zamiast budować cały skrypt od zera. Liczbę kroków zapisuje w gotowej tabeli z podpisanymi wierszami.

Kryteria oceny

Trzy poziomy dla każdej umiejętności. Opis mówi, co uczeń pokazuje, a nie ile punktów dostaje.

UMIEJĘTNOŚĆ	PODSTAWOWY	DOBRY	BARDZO DOBRY
-------------	------------	-------	--------------

UMIEJĘTNOŚĆ	PODSTAWOWY	DOBRY	BARDZO DOBRY
Zapis algorytmu w krokach	Uczeń przepisuje z tablicy algorytm połowienia jako ponumerowane kroki i wskazuje krok, który kończy działanie, gdy danej nie ma.	Uczeń zapisuje własnymi słowami oba algorytmy z lekcji tak, że kolega wykonuje je dosłownie bez pytań, z jedną nazwą dla każdej zmiennej i regułą zaokrąglania.	Uczeń znajduje w cudzym zapisie krok niejednoznaczny albo brakujący warunek zakończenia, pokazuje na przykładzie danych, co się przez to psuje, i poprawia zapis.
Wyszukiwanie po kolei i przez połowienie	Uczeń wykonuje oba algorytmy na liście 16 numerów dla podanego numeru i poprawnie liczy kroki jako porównania.	Uczeń wypełnia całą tabelę dla trzech numerów, w tym nieobecnego, i tłumaczy, dlaczego przy numerze z początku listy przewaga połowienia prawie znika.	Uczeń wyznacza największą liczbę kroków połowienia dla zbioru 1000 i miliona pozycji regułą podwajania i uzasadnia, kiedy mimo to wybraćby szukanie po kolei, wskazując koszt porządkowania.

Pytania do dyskusji

Na podsumowanie lekcji. Nie mają jednej poprawnej odpowiedzi, chodzi o uzasadnienie.

1. Papierowy słownik i katalog biblioteki są uporządkowane, a półka z gramy w domu zwykle nie. Co decyduje o tym, czy warto ponosić koszt porządkowania?
2. Kiedy zgadywaliście liczbę na początku lekcji, ktoś zaproponował pytanie o środek zakresu i część klasy protestowała. Dlaczego pytanie o środek wydaje się gorsze, skoro jest najlepsze?
3. Czy zapis kroków dla człowieka i dla komputera powinien wyglądać tak samo? Co człowiek dopowie sobie sam, a czego komputerowi nie wolno pominąć?

Częste pytania uczniów

Po co mam się uczyć połowienia, skoro komputer i tak szuka błyskawicznie?

Bo szybkość komputera kończy się tam, gdzie zaczynają się duże zbiory. Przy milionie pozycji szukanie po kolei to do miliona sprawdzeń, a połowienie do dwudziestu. Ta różnica decyduje o tym, czy wyszukiwarka odpowiada od razu, czy po minucie.

Czy muszę mieć konto w Scratchu, żeby zapisać swój skrypt?

Nie. Edytor działa w przeglądarce bez logowania, a projekt zapisujesz poleceniem Plik, a potem Zapisz na swoim komputerze. Powstaje plik, który skopiujesz na pendrive albo wyślesz sobie pocztą. Bez konta projekt nie zostaje na stronie po zamknięciu karty.

Czy połowienie zawsze jest szybsze od szukania po kolei?

Nie. Jeśli szukana dana leży na początku listy albo lista ma pięć pozycji, szukanie po kolei bywa krótsze. Połowienie wygrywa przy dużych zbiorach i w najgorszym przypadku, ale wymaga listy uporządkowanej, a samo uporządkowanie też kosztuje czas.

Mini projekt na kolejne lekcje

Na 2 do 3 kolejne lekcje albo na kółko. Buduje na tym, co uczniowie zrobili na tej lekcji.

Wyszukiwarka w katalogu klasowej biblioteczki

Klasa buduje w Scratchu prosty katalog z listą uporządkowanych numerów i wyszukiwarką, która działa połowieniem i pokazuje liczbę wykonanych kroków. Projekt zajmuje dwie do trzech lekcji albo dwa spotkania kółka i buduje na skrypcie zgadywania z tej lekcji.

1. Lekcja 1: para spisuje 20 do 30 rzeczywistych tytułów z klasowej półki albo z domu, nadaje im numery i porządkuje listę rosnąco na kartce, a potem wpisuje numery do listy w Scratchu (kategoria Zmienne, przycisk Utwórz listę).
2. Lekcja 2: para przerabia skrypt zgadywania z tej lekcji tak, żeby zamiast pytać gracza, sam porównywał szukany numer z elementem listy na pozycji strzał i zwracał zakres; wynikiem jest pozycja albo komunikat „nie ma”.
3. Lekcja 3: para dopisuje licznik kroków i drugi przycisk z szukaniem po kolei, po czym dla pięciu numerów zapisuje w tabeli, ile kroków zajęła każda metoda.
4. Prezentacja: każda para pokazuje klasie jeden numer, przy którym połowienie wygrało wyraźnie, i jeden, przy którym różnica była mała, i tłumaczy dlaczego.

Co powstaje na końcu: Działający projekt Scratch z listą numerów, dwiema metodami wyszukiwania i licznikiem kroków oraz tabela porównawcza dla pięciu numerów, gotowa do pokazania na następnej lekcji o bazach danych.