

KOMPLET 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.

W tym pliku są wszystkie materiały do lekcji: scenariusz z kluczem do karty pracy i kryteriami oceny, prezentacja, karta pracy, materiały dla ucznia, kartka dla rodzica oraz quiz w wersji dla ucznia i z odpowiedziami.

Co znajdziecie w środku

1. Scenariusz lekcji
2. Prezentacja na tablicę
3. Karta pracy do druku
4. Materiały dla ucznia
5. Kartka dla rodzica
6. Quiz dla ucznia
7. Quiz z odpowiedziami

SCENARIUSZ LEKCJI

Algorytm wyszukiwania w zbiorze danych

Cel lekcji

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 wróćcie 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 szukanym. 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 szukanym. 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, próby. Środek liczy się blokiem „podłoga z” (zaokrąglenie w dół), nie zwykłym blokiem „zaokrąglij”, bo ten zaokrągli 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 zwężają zakres o jeden.

4**Sprawdzenie, 5 min**

Nauczyciel rysuje na tablicy tabelę z trzema szukanymi 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
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.

UMIEJĘTNOŚĆ	PODSTAWOWY	DOBRY	BARDZO DOBRY
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 wybrałby 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 zwężał 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.

PREZENTACJA NA TABLICĘ

Algorytm wyszukiwania w zbiorze danych

Slajdów: 8. Notatka pod slajdem jest dla nauczyciela, uczniowie nie widzą jej na tablicy.

SLAJD 1

Zgadnij liczbę od 1 do 1000

- Pytania tylko o za mało albo za dużo
- Liczymy pytania na tablicy
- Po kolei to do 1000 pytań

NOTATKA DLA NAUCZYCIELA

Zagrajcie naprawdę, zanim cokolwiek wyjaśnicie. Liczba kresek na tablicy jest lepszym wstępem niż definicja.

SLAJD 2

Czym jest algorytm

- Skończony ciąg jednoznacznych kroków
- Ten sam wynik przy każdym wykonaniu
- Wykonuje go ktoś bez wiedzy o zadaniu

NOTATKA DLA NAUCZYCIELA

Podkreście słowo jednoznacznych. Cała praca uczniów sprawdza właśnie to.

SLAJD 3

Nasz zbiór danych

- 16 numerów inwentarza biblioteki
- Uporządkowane rosnąco
- Pozycje ponumerowane od 1 do 16

NOTATKA DLA NAUCZYCIELA

Wyświetlcie listę i zostawcie ją na ekranie przez całą lekcję. Uczniowie będą do niej wracać.

SLAJD 4

Algorytm A, szukanie po kolei

- Sprawdzaj pozycje od pierwszej do ostatniej
- Trafienie kończy działanie
- Koniec listy oznacza odpowiedź nie ma

NOTATKA DLA NAUCZYCIELA

Zapiszcie te kroki numerami na tablicy, nie zdaniami ciągłymi. Uczniowie mają je przepisać.

SLAJD 5

Algorytm B, połowienie zakresu

- Środek to suma granic przez 2, w dół
- Za mało podnosi dolną, za dużo obniża górną
- Dolna większa od górnej kończy szukanie

NOTATKA DLA NAUCZYCIELA

Wykonajcie ten algorytm na żywo dla numeru 1401. Wychodzą 2 kroki i klasa od razu widzi różnicę.

SLAJD 6

Warunek zakończenia

- Algorytm musi umieć powiedzieć nie ma
- Bez tego warunku wykonanie się nie kończy
- Sprawdzamy na numerze 1300 spoza listy

NOTATKA DLA NAUCZYCIELA

To najczęściej pomijany element w uczniowskich zapisach. Wracajcie do niego przy sprawdzaniu prac.

SLAJD 7

Ile kroków przy większym zbiorze

- 16 pozycji to do 5 kroków połowienia
- 1000 pozycji to do 10 kroków
- 1 000 000 pozycji to do 20 kroków

NOTATKA DLA NAUCZYCIELA

Poproście klasę o dokończenie ciągu samodzielnie. Reguła podwajania jest tu ważniejsza niż same liczby.

SLAJD 8

Kiedy połowienie nie pomaga

- Lista musi być uporządkowana
- Dana blisko początku znajduje się szybciej po kolei
- Porządkowanie zbioru też kosztuje czas

NOTATKA DLA NAUCZYCIELA

Bez tego slajdu uczniowie wyjdą z przekonaniem, że jedna metoda jest zawsze lepsza. Podajcie kontrprzykład z lekcji, numer 1199.

KARTA PRACY DO DRUKU

Algorytm wyszukiwania w zbiorze danych

Imię ucznia: Klasa: Data:

Dostajesz listę 16 numerów inwentarza ze szkolnej biblioteki, uporządkowaną od najmniejszego do największego. Twoim zadaniem jest wykonać na niej dwa algorytmy wyszukiwania i policzyć, który potrzebuje mniej kroków.

1

Przepisz z tablicy algorytm B, czyli połowienie zakresu, jako ponumerowane kroki. Podkreśl ten krok, który kończy działanie algorytmu, gdy szukanego numeru nie ma na liście.

Wskazówka: Krok kończący zaczyna się od słowa „jeśli” i porównuje dolną z górną. Bez niego algorytm nigdy się nie zatrzyma.

2

Wykonaj oba algorytmy dla numerów 1199, 1560 i 1305. Do tabeli wpisz w kolejnych kolumnach: szukany numer, liczbę kroków algorytmu A, liczbę kroków algorytmu B, wynik czyli pozycję albo słowo „nie ma”.

Wskazówka: Jeden krok to jedno porównanie szukanego numeru z numerem z listy. Numer 1305 nie występuje na liście i to nie jest pomyłka w druku.

3

Napisz jednym zdaniem, przy którym z trzech numerów przewaga połowienia była najmniejsza i dlaczego akurat przy nim.

Wskazówka: Sprawdź, na której pozycji od początku listy leży ten numer.

4

W edytorze Scratch zbuduj skrypt zgadujący liczbę od 1 do 100 metodą połowienia. Ustaw zmienne dolna na 1, gorna na 100, proby na 0. W pętli powtarzanej aż odpowiedź będzie równa słowu trafione wykonaj: ustaw strzał na podłoga z ((dolna plus gorna) podzielone przez 2), czyli zaokrąglenie w dół, zmień proby o 1, zapytaj „Czy Twoja liczba to strzał? Odpowiedz: za mało, za dużo albo trafione” i czekaj, jeżeli odpowiedź to za mało ustaw dolna na strzał plus 1, jeżeli odpowiedź to za dużo ustaw gorna na strzał minus 1.

Wskazówka: Bloki zmiennych znajdziesz w kategorii Zmienne, blok zapytaj i czekaj w kategorii Czujniki, a blok „podłoga z” w kategorii Wyrażenia: to blok z listą rozwijaną, w której domyślnie stoi „pierwiastek”, wybierz w niej „podłoga”. Nie używaj bloku „zaokrąglaj”, bo zaokrągli 5,5 do 6, a algorytm z tablicy zaokrągli w dół. Pamiętaj o plus 1 i minus 1, bez nich zakres przestanie się zmniejszać.

5

Przetestuj skrypt na trzech różnych liczbach i zapisz największą liczbę prób, jaką uzyskałeś. Dopisz, ile prób potrzebowałby ten sam skrypt przy zakresie od 1 do 1000.

Wskazówka: Za każdym razem gdy zakres rośnie dwukrotnie, dochodzi dokładnie jedna próba. Policz, ile razy trzeba podwoić liczbę 1, żeby przekroczyć 1000: 1, 2, 4, 8 i dalej. Liczba podwojeń to liczba prób w najgorszym przypadku.

MATERIAŁY DLA UCZNIA

Algorytm wyszukiwania w zbiorze danych

Czego się nauczysz

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.

Kiedy zgadujesz liczbę od 1 do 1000, a druga osoba mówi tylko „za mało” albo „za dużo”, zwykle trafiasz po kilkunastu pytaniach, choć liczb jest tysiąc. To nie jest szczęście, tylko konkretna metoda, którą da się zapisać tak dokładnie, żeby wykonał ją ktoś zupełnie nieświadomy sprytu, na przykład komputer. Z tego materiału dowiesz się, czym różni się algorytm od pomysłu i dlaczego przy milionie danych połowienie zakresu potrzebuje około dwudziestu kroków.

Co odróżnia algorytm od zwykłego pomysłu

Pomysł brzmi „pytaj o środek i zawężaj”. To wystarczy dla człowieka, który zna zasadę gry, ale nie wystarczy dla wykonawcy, który nie wie nic. Algorytm to skończony ciąg kroków na tyle jednoznacznych, że wykonawca nie musi niczego się domyślać ani niczego wybierać na oko, i który przy tych samych danych zawsze daje ten sam wynik.

Jednoznaczność sprawdza się prostym testem: dajesz zapis drugiej osobie i każesz wykonać dosłownie, bez dopowiadania. Krok „sprawdź środek” ten test oblewa, bo przy zakresie od 1 do 16 środek może wypaść na 8 albo na 9. Krok „ustaw środek na wynik dzielenia sumy dolnej i górnej przez 2, zaokrąglony w dół” test przechodzi, bo nie zostawia miejsca na wybór.

Druga rzecz, bez której zapis nie jest algorytmem, to warunek zakończenia. Algorytm musi umieć zakończyć się także wtedy, gdy szukanej danej w zbiorze nie ma, i podać wtedy sensowną odpowiedź. Bez takiego kroku wykonanie kręci się w kółko, bo żaden warunek go nie przerywa.

Zapisując algorytm, numeruj kroki i używaj tych samych nazw przez cały czas. Jeśli w kroku drugim mówisz „dolna”, nie pisz w kroku szóstym „początek zakresu”, bo wykonawca nie ma prawa wiedzieć, że chodzi o to samo.

SPRÓBUJ SAM

Napisz instrukcję wiązania butów w ponumerowanych krokach i daj ją komuś, kto wykona ją dosłownie, bez poprawiania Twoich niedomówień. Zaznacz każdy krok, przy którym musiał się czegoś domyślić. To są dokładnie te miejsca, w których Twój zapis nie był jeszcze algorytmem.

Szukanie po kolei

Najprostsza metoda wyszukiwania polega na sprawdzaniu pozycji jedna po drugiej, od pierwszej do ostatniej. Zapisana jako algorytm wygląda tak: 1. Ustaw pozycję na 1. 2. Porównaj numer na tej pozycji z szukanym. 3. Jeśli równy, wypisz pozycję i zakończ. 4. Jeśli pozycja to ostatnia, wypisz „nie ma” i zakończ. 5. Zwiększ pozycję o 1 i wróć do kroku 2.

Zwróć uwagę, że kroki 3 i 4 to dwa różne zakończenia: jedno dla trafienia, drugie dla wyczerpania listy. Oba są potrzebne. Gdyby zabrakło kroku 4, algorytm szukałby danej, której nie ma, poza końcem listy.

Ta metoda ma dwie ważne zalety. Po pierwsze, działa na zbiorze w dowolnej kolejności, bo niczego nie zakłada o uporządkowaniu. Po drugie, jest krótka w zapisie i trudno się w niej pomylić. Ma jednak jedną poważną wadę: liczba kroków rośnie tak samo szybko jak zbiór. Przy 16 pozycjach to do 16 porównań, przy tysiącu do tysiąca, przy milionie do miliona.

Jest jeszcze jeden szczegół, który przyda się dalej. Liczba kroków szukania po kolei zależy od tego, gdzie leży szukana dana. Jeśli jest piąta od początku, wystarczy 5 kroków, niezależnie od tego, czy lista ma 16 pozycji, czy tysiąc. Najgorszy przypadek to dana na samym końcu albo dana, której w zbiorze nie ma.

SPRÓBUJ SAM

Weź listę 16 numerów inwentarza: 1042, 1087, 1130, 1156, 1199, 1204, 1233, 1290, 1315, 1342, 1388, 1401, 1455, 1470, 1512, 1560. Policz, ile kroków zajmie szukanie po kolei numeru 1199, numeru 1560 i numeru 1305, którego na liście nie ma. Zapisz trzy liczby i sprawdź, która z nich jest największa.

Połowienie zakresu

Druga metoda korzysta z jednej dodatkowej informacji: lista jest uporządkowana rosnąco. Dzięki temu porównanie z jedną pozycją mówi coś o wszystkich pozostałych. Jeśli numer ze środka jest mniejszy od szukanego, cała lewa połowa listy odpada, bo wszystko tam jest jeszcze mniejsze. Jednym porównaniem odrzucasz połowę zbioru.

Zapis wygląda tak: 1. Ustaw dolną na 1, górną na ostatnią pozycję. 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.

Najważniejszy i najczęściej psuty jest krok 6. Dolną ustawiasz na środek plus 1, a nie na środek, i górną na środek minus 1, a nie na środek. Powód jest praktyczny: pozycję ze środka właśnie sprawdziłeś, więc nie ma sensu trzymać jej w zakresie. Bez tych jedynek zakres przestaje się kurczyć, gdy zostaną dwie pozycje, i algorytm nigdy się nie kończy.

Sprawdźmy to na liście z poprzedniej sekcji dla numeru 1401. Zakres 1 do 16, środek 8, tam stoi 1290, czyli mniej, więc dolna idzie na 9. Zakres 9 do 16, środek 12, tam stoi 1401, trafione. Dwa kroki na szesnastu pozycjach. Dla numeru 1305, którego na liście nie ma, po czterech krokach dolna przekracza górną i algorytm kończy się odpowiedzią „nie ma”, dokładnie tak, jak przewiduje krok 2.

SPRÓBUJ SAM

Wykonaj połowienie zakresu na tej samej liście 16 numerów dla wartości 1199 i 1560, zapisując po każdym kroku trzy liczby: dolną, górną i środek. Porównaj wynik z liczbą kroków szukania po kolei z poprzedniej sekcji. Przy jednym z tych numerów przewaga połowienia będzie zaskakująco mała i warto sprawdzić, na której pozycji leży ten numer.

Ile kroków przy większym zbiorze

Skoro każdy krok zmniejsza zakres o połowę, to pytanie o liczbę kroków jest pytaniem o to, ile razy trzeba przepołowić zbiór, żeby zostać z jedną pozycją. Najłatwiej policzyć to od drugiej strony: ile razy trzeba podwoić liczbę 1, żeby dojść do wielkości zbioru. 1, 2, 4, 8, 16 to cztery podwojenia, więc zbiór 16 pozycji wymaga do 5 kroków razem z ostatnim sprawdzeniem.

Dalej ta sama arytmetyka daje wyniki, które na początku wydają się nieprawdopodobnie. Tysiąc pozycji to do 10 kroków. Milion pozycji to do 20 kroków. Miliard to do 30. Podwojenie wielkości zbioru dokładnie dokładnie jeden krok, a nie podwaja pracy, i to jest cała siła tej metody.

Porównaj to z szukaniem po kolei. Przy milionie pozycji jedna metoda potrzebuje do miliona porównań, a druga do dwudziestu. Ta różnica decyduje o tym, czy wyszukiwarka albo system biblioteczny odpowiada natychmiast, czy po długiej chwili. Nie chodzi o to, że komputer jest wolny. Chodzi o to, ile pracy mu zadałeś.

Warto też zauważyć, że liczby te dotyczą najgorszego przypadku. Połowienie może trafić w pierwszym kroku, jeśli szukana dana stoi dokładnie w środku. Szukanie po kolei może trafić w pierwszym kroku, jeśli dana stoi na początku. Metody porównuje się jednak po najgorszym przypadku, bo to on decyduje, czy system zadziała zawsze, a nie tylko czasem.

SPRÓBUJ SAM

Wypisz ciąg podwojeń: 1, 2, 4, 8, 16, 32 i dalej, aż przekroczysz 1000. Policz, ile razy podwoiłeś liczbę, czyli ile strzałek między kolejnymi liczbami zapisałeś, a nie ile liczb. Powinno wyjść 10. Potem zrób to samo do miliona i sprawdź, czy wynik zgadza się z liczbą 20. Na koniec odpowiedz: o ile kroków wzrośnie praca, gdy zbiór urośnie z miliona do dwóch milionów.

Kiedy połowienie nie pomaga

Połowienie ma jeden twardy warunek: zbiór musi być uporządkowany. Cała metoda opiera się na założeniu, że wszystko na lewo od środka jest mniejsze, a wszystko na prawo większe. Na liście w przypadkowej kolejności to założenie nie obowiązuje, więc porównanie ze środkiem nie mówi, po której stronie szukać. Algorytm wykona się do końca i poda wynik, tylko że ten wynik może być błędny, a to gorsze niż brak wyniku.

Dlatego przed wyborem metody warto dopisać sobie krok zerowy: sprawdź, czy zbiór jest uporządkowany. Jeśli nie jest, masz dwa wyjścia. Możesz szukać po kolei albo najpierw uporządkować zbiór, a porządkowanie też kosztuje czas, i to zwykle więcej niż jedno szukanie po kolei.

Stąd praktyczny wniosek: jeśli szukasz w zbiorze tylko raz i zbiór jest krótki, prostsze i szybsze bywa szukanie po kolei. Połowienie opłaca się wtedy, gdy zbiór jest duży, uporządkowany i przeszukiwany wiele razy. Dokładnie tak działają systemy biblioteczne czy sklepowe: dane trzyma się w uporządkowanej postaci, bo szuka się w nich bez przerwy.

Jest też przypadek pośredni, który łatwo przeoczyć. Jeśli szukana dana leży blisko początku listy, szukanie po kolei potrafi wygrać nawet na uporządkowanym zbiorze. Na liście szesnastu numerów szukanie piątej pozycji zajmuje po kolei 5 kroków, a połowieniem 4, więc przewaga niemal znika. Metoda nie jest lepsza sama w sobie, jest lepsza dla konkretnych danych i konkretnego sposobu użycia.

SPRÓBUJ SAM

Weź listę 16 numerów i przestaw je w przypadkowej kolejności. Wykonaj na tak przestawionej liście algorytm połowienia dla numeru 1130, wykonując kroki dosłownie, bez zaglądania na całą listę. Zapisz, jaką odpowiedź podała algorytm i czy jest zgodna z prawdą.

Słowniczek

algorytm

Skończony ciąg jednoznacznych kroków, który przy tych samych danych zawsze daje ten sam wynik i da się wykonać bez domysłu.

warunek zakończenia

Krok, który przerywa wykonanie algorytmu, także wtedy, gdy szukanej danej w zbiorze nie ma.

krok

Jedno porównanie szukanej wartości z wartością ze zbioru; jednostka, w której mierzymy pracę algorytmu wyszukiwania.

szukanie po kolei

Metoda sprawdzająca pozycje jedna po drugiej od początku zbioru, działająca także na zbiorze nieuporządkowanym.

połowienie zakresu

Metoda odrzucająca po każdym porównaniu połowę zbioru, wymagająca zbioru uporządkowanego.

zbiór uporządkowany

Zbiór ustawiony rosnąco albo malejąco, w którym pozycja wartości mówi coś o wszystkich pozostałych.

najgorszy przypadek

Największa liczba kroków, jakiej metoda może potrzebować dla danego zbioru; po nim porównuje się metody.

Częste pomyłki

- Stosowanie połowienia do listy w przypadkowej kolejności. Algorytm poda wtedy wynik, który bywa błędny, a to trudniej wykryć niż brak wyniku. Zawsze sprawdź uporządkowanie, zanim wybierzesz metodę.
- Pominięcie kroku „jeśli dolna jest większa od górnej, wypisz nie ma i zakończ”. Bez niego szukanie danej spoza zbioru nie ma jak się skończyć i wykonanie kręci się w kółko.
- Ustawianie dolnej na środek zamiast na środek plus 1. Przy zakresie od 5 do 6 środek zawsze wypada na 5, więc zakres przestaje się kurczyć i algorytm pyta w nieskończoność.

- Zaokrąglanie środka raz w górę, raz w dół. Te same dane dają wtedy różne liczby kroków i nie da się porównać wyników. Zapisz regułę zaokrąglania obok algorytmu i trzymaj się jej do końca.

Sprawdź, czy rozumiesz

1. Na uporządkowanej liście 16 pozycji szukasz wartości leżącej na pozycji trzeciej. Która metoda wypadnie lepiej i czy Twoja odpowiedź zmieni się, gdy lista urośnie do tysiąca pozycji?
2. Kolega zapisał algorytm połowienia i ustawia dolną na środek zamiast na środek plus 1. Na jakim najmniejszym zakresie pokażesz mu, że jego zapis nigdy się nie kończy?
3. Masz zbiór miliona nieuporządkowanych danych i musisz znaleźć w nim jedną wartość dokładnie raz. Czy opłaca się najpierw go uporządkować i od czego zależy Twoja odpowiedź?

Wyzwanie dla chętnych

PONAD PROGRAM

Zbuduj w Scratchu wersję odwrotną: to Ty zgadujesz liczbę od 1 do 1000 pomyślaną przez komputer (blok „losuj od 1 do 1000”), a program liczy Twoje próby i odpowiada „za mało” albo „za dużo”. Potem zagraj dziesięć razy i zapisz największą liczbę prób, jaką potrzebowałeś.

Jak sprawdzisz, że Ci wyszło: Jeśli za każdym razem pytasz o środek zakresu, żadna z dziesięciu gier nie przekroczy 10 prób. Jeśli któraś przekroczyła, wróć do jej zapisu i znajdź pytanie, które nie było o środek.

Zadanie do domu na trzy poziomy

Wybierz jeden poziom. Możesz zacząć od łatwego i wrócić po trudniejszy.

POZIOM: ŁATWE

Ułóż 8 karteczek z liczbami 3, 8, 15, 21, 27, 34, 40, 48 rosnąco, liczbami do dołu, i znajdź liczbę 34 połowieniem, odwracając tylko wskazaną karteczkę. Zapisz, które karteczki odwróciłeś i ile ich było. Powinny wyjść dokładnie 2.

POZIOM: ŚREDNIE

Na liście 16 numerów z lekcji policz kroki obu metod dla numerów 1042, 1342 i 1600 (tej ostatniej na liście nie ma). Zapisz wyniki w tabeli i podkreśl numer, przy którym szukanie po kolei wygrało.

POZIOM: TRUDNE

Zbuduj w Scratchu skrypt zgadujący liczbę od 1 do 1000 i dopisz sprawdzenie uczciwości: jeśli dolna przekroczy górną, a gracz nie powiedział „trafione”, program wypisuje, że odpowiedzi były sprzeczne. Wyjaśnij w dwóch zdaniach, dlaczego ten sam warunek w algorytmie wyszukiwania oznacza „nie ma”.

Samoocena

Zaznacz, co już potrafisz. To tylko dla Ciebie, nikt tego nie ocenia.

- ☐ Potrafię zapisać algorytm jako ponumerowane kroki, które kolega wykona bez domyślania.
- ☐ Potrafię wskazać w algorytmie krok, który kończy działanie, gdy szukanej danej nie ma.
- ☐ Potrafię policzyć kroki szukania po kolei i połowienia dla podanego numeru na liście 16 pozycji.
- ☐ Potrafię powiedzieć, kiedy połowienie się nie opłaca albo w ogóle nie działa.

KARTKA DLA RODZICA

Algorytm wyszukiwania w zbiorze danych

Szanowni Rodzice, dziecko miało dziś lekcję z Cyfrowej Pracowni. Poniżej w dwóch zdaniach, czego się nauczyło, i jedno pytanie, które możecie zadać przy kolacji.

Czego dziecko nauczyło się na tej lekcji

Wasze dziecko potrafi zapisać sposób rozwiązania zadania jako ponumerowane kroki, które da się wykonać bez domyślania, i wie, że taki zapis musi mieć się zakończyć. Umie też porównać dwie metody szukania w uporządkowanej liście i policzyć, że w zbiorze tysiąca pozycji wystarczy około dziesięciu sprawdzeń.

Pytanie na kolację

Pomyślę liczbę od 1 do 100, a Ty zgadnij ją w jak najmniejszej liczbie pytań. Powiesz mi potem, dlaczego zawsze pytasz o środek?

Zadanie domowe do wyboru

Dziecko wybiera jeden poziom. Każdy jest do zrobienia bez pomocy nauczyciela.

POZIOM: ŁATWE

Ułóż 8 karteczek z liczbami 3, 8, 15, 21, 27, 34, 40, 48 rosnąco, liczbami do dołu, i znajdź liczbę 34 połowieniem, odwracając tylko wskazaną karteczkę. Zapisz, które karteczki odwróciłeś i ile ich było. Powinny wyjść dokładnie 2.

POZIOM: ŚREDNIE

Na liście 16 numerów z lekcji policz kroki obu metod dla numerów 1042, 1342 i 1600 (tej ostatniej na liście nie ma). Zapisz wyniki w tabeli i podkreśl numer, przy którym szukanie po kolei wygrało.

POZIOM: TRUDNE

Zbuduj w Scratchu skrypt zgadujący liczbę od 1 do 1000 i dopisz sprawdzenie uczciwości: jeśli dolna przekroczy górną, a gracz nie powiedział „trafione”, program wypisuje, że odpowiedzi były sprzeczne. Wyjaśnij w dwóch zdaniach, dlaczego ten sam warunek w algorytmie wyszukiwania oznacza „nie ma”.

QUIZ DLA UCZNIA

Algorytm wyszukiwania w zbiorze danych

Zaznacz jedną odpowiedź w każdym pytaniu. Wynik nigdzie się nie zapisuje.

1. Który zapis jest algorytmem, a nie zwykłym opisem pomysłu?**1 Z 10, POZIOM PODSTAWOWY**

- ☐ A. Poszukaj numeru gdzieś w środku listy, a potem sprawdzaj dalej
- ☐ B. 1. Ustaw dolną na 1, górną na 16. 2. Jeśli dolna większa od górnej, wypisz nie ma i zakończ. 3. Sprawdź numer ze środka zakresu
- ☐ C. Trzeba znaleźć numer możliwie szybko i nie sprawdzać niepotrzebnych pozycji

2. Uczeń zapisał algorytm połowienia bez kroku „jeśli dolna jest większa od górnej, wypisz nie ma i zakończ”. Co się stanie przy szukaniu numeru, którego nie ma na liście?**2 Z 10, POZIOM PODSTAWOWY**

- ☐ A. Algorytm zwróci pozycję najbliższego numeru
- ☐ B. Algorytm wypisze komunikat o błędzie danych
- ☐ C. Algorytm nie będzie miał jak się zakończyć i będzie się wykonywał w kółko

3. Na uporządkowanej liście 16 numerów szukamy ostatniego z nich. Ile kroków potrzebuje szukanie po kolei, a ile połowienie zakresu?**3 Z 10, POZIOM ROZSZERZONY**

- ☐ A. Po kolei 16 kroków, połowienie 5 kroków
- ☐ B. Po kolei 5 kroków, połowienie 16 kroków
- ☐ C. Obie metody potrzebują po 8 kroków
- ☐ D. Po kolei 1 krok, połowienie 1 krok

4. Lista 16 numerów jest ustawiona w przypadkowej kolejności. Co da zastosowanie do niej algorytmu połowienia zakresu?**4 Z 10, POZIOM PODSTAWOWY**

- ☐ A. Ten sam wynik, tylko wolniej
- ☐ B. Wynik może być błędny, bo porównanie ze środkiem nie mówi, po której stronie leży szukany numer
- ☐ C. Algorytm sam uporządkuje listę przed szukaniem

5. Zbiór ma 1000 uporządkowanych pozycji. Ilu kroków potrzebuje połowienie zakresu w najgorszym przypadku?

5 Z 10, POZIOM ROZSZERZONY

- ☐ A. Do 500 kroków, czyli połowy zbioru
- ☐ B. Do 100 kroków
- ☐ C. Do 10 kroków, bo każdy krok zmniejsza zakres o połowę
- ☐ D. Dokładnie tyle samo co szukanie po kolei

6. Krok „sprawdź środek listy” oblewa test jednoznaczności. Dlaczego?

6 Z 10, POZIOM PODSTAWOWY

- ☐ A. Bo przy 16 pozycjach środek może wypaść na 8 albo na 9 i wykonawca musi sam wybrać
- ☐ B. Bo lista ma za mało pozycji, żeby miała środek
- ☐ C. Bo środek listy wolno sprawdzać dopiero na końcu

7. Uczeń zapisał szukanie po kolei tak: 1. Ustaw pozycję na 1. 2. Porównaj numer na tej pozycji z szukany. 3. Jeśli równy, wypisz pozycję i zakończ. 4. Zwiększ pozycję o 1 i wróć do kroku 2. Wykonawca szuka numeru 1305 na liście 16 numerów z lekcji. Co się stanie?

7 Z 10, POZIOM ROZSZERZONY

- ☐ A. Po 16 porównaniach algorytm wypisze „nie ma” i zakończy się
- ☐ B. Algorytm zakończy się po 9 porównaniach, bo numer 1315 jest większy od 1305
- ☐ C. Algorytm wypisze pozycję 8, bo 1290 jest najbliższe 1305
- ☐ D. Po 16 porównaniach wykonawca ustawi pozycję na 17 i spróbuje porównać numer, którego nie ma, bo brakuje kroku kończącego dla końca listy

8. Dwie pary wykonały ten sam zapis algorytmu połowienia dla numeru 1233 na liście 16 numerów z lekcji. Jedna para uzyskała 3 kroki, druga 4. Zapis mówił „ustaw środek na sumę dolnej i górnej podzieloną przez 2”, bez słowa o zaokrąglaniu. Co jest przyczyną różnicy?

8 Z 10, POZIOM ROZSZERZONY

- ☐ A. Jedna z par pomyliła się przy porównaniu i policzyła o jeden krok za mało
- ☐ B. Numer 1233 stoi na liście na dwóch różnych pozycjach
- ☐ C. Zapis nie był jednoznaczny: przy 17 podzielonym przez 2 zaokrąglenie w górę daje środek 9 i 3 kroki, a zaokrąglenie w dół daje środek 8 i 4 kroki
- ☐ D. Połowienie zawsze daje inną liczbę kroków przy każdym wykonaniu

9. Kiedy szukanie po kolei jest lepszym wyborem niż połowienie zakresu?**9 Z 10, POZIOM PODSTAWOWY**

- ☐ A. Gdy lista ma milion pozycji i szukamy w niej codziennie
- ☐ B. Gdy lista jest uporządkowana i szukamy w niej wiele razy
- ☐ C. Gdy lista jest nieuporządkowana, a szukamy w niej tylko raz

10. Na uporządkowanej liście 16 numerów z lekcji szukasz numeru 1087. Ile kroków potrzebuje szukanie po kolei, a ile połowienie zakresu, jeśli krok to jedno porównanie?**10 Z 10, POZIOM ROZSZERZONY**

- ☐ A. Po kolei 16 kroków, połowienie 5 kroków
- ☐ B. Po kolei 2 kroki, połowienie 3 kroki, więc tym razem wygrywa szukanie po kolei
- ☐ C. Po kolei 2 kroki, połowienie 2 kroki, remis
- ☐ D. Po kolei 2 kroki, połowienie 4 kroki

QUIZ Z ODPOWIEDZIAMI

Algorytm wyszukiwania w zbiorze danych

Wersja dla nauczyciela. Poprawna odpowiedź jest zaznaczona, pod każdym pytaniem jest wyjaśnienie.

1. Który zapis jest algorytmem, a nie zwykłym opisem pomysłu?**1 Z 10, POZIOM PODSTAWOWY**

- ☐ A. Poszukaj numeru gdzieś w środku listy, a potem sprawdzaj dalej
- ☒ B. **1. Ustaw dolną na 1, górną na 16. 2. Jeśli dolna większa od górnej, wypisz nie ma i zakończ. 3. Sprawdź numer ze środka zakresu**
- ☐ C. Trzeba znaleźć numer możliwie szybko i nie sprawdzać niepotrzebnych pozycji

Poprawna odpowiedź: B.

Tylko drugi zapis składa się z ponumerowanych kroków, które da się wykonać bez zgadywania, i zawiera warunek zakończenia. Pierwszy zapis każe szukać „gdzieś w środku”, więc wykonawca musi zgadywać, a trzeci mówi tylko, jaki ma być efekt, nie jak go osiągnąć.

Umiejętność: Zapis algorytmu w krokach

2. Uczeń zapisał algorytm połowienia bez kroku „jeśli dolna jest większa od górnej, wypisz nie ma i zakończ”. Co się stanie przy szukaniu numeru, którego nie ma na liście?**2 Z 10, POZIOM PODSTAWOWY**

- ☐ A. Algorytm zwróci pozycję najbliższego numeru
- ☐ B. Algorytm wypisze komunikat o błędzie danych
- ☒ C. **Algorytm nie będzie miał jak się zakończyć i będzie się wykonywał w kółko**

Poprawna odpowiedź: C.

Brak warunku zakończenia oznacza, że żaden krok nie przerywa wykonania, gdy szukanej danej w zbiorze nie ma, więc algorytm kręci się w kółko. Nie zwróci najbliższego numeru, bo żaden krok tego nie każe, i nie wypisze komunikatu o błędzie, bo taki krok nie został zapisany.

Umiejętność: Zapis algorytmu w krokach

3. Na uporządkowanej liście 16 numerów szukamy ostatniego z nich. Ile kroków potrzebuje szukanie po kolei, a ile połowienie zakresu?

3 Z 10, POZIOM ROZSZERZONY

- ☒ **A. Po kolei 16 kroków, połowienie 5 kroków**
- ☐ B. Po kolei 5 kroków, połowienie 16 kroków
- ☐ C. Obie metody potrzebują po 8 kroków
- ☐ D. Po kolei 1 krok, połowienie 1 krok

Poprawna odpowiedź: A.

Szukanie po kolei sprawdza wszystkie 16 pozycji, bo ostatni numer stoi na końcu. Połowienie sprawdza pozycje 8, 12, 14, 15 i 16, czyli 5 kroków. Odpowiedź z zamienionymi liczbami myli metody, po 8 kroków nie wychodzi w żadnej z nich, a 1 krok byłby możliwy tylko przy trafieniu od razu w pierwszą sprawdzaną pozycję.

Umiejętność: Wyszukiwanie po kolei i przez połowienie

4. Lista 16 numerów jest ustawiona w przypadkowej kolejności. Co da zastosowanie do niej algorytmu połowienia zakresu?

4 Z 10, POZIOM PODSTAWOWY

- ☐ A. Ten sam wynik, tylko wolniej
- ☒ **B. Wynik może być błędny, bo porównanie ze środkiem nie mówi, po której stronie leży szukany numer**
- ☐ C. Algorytm sam uporządkuje listę przed szukaniem

Poprawna odpowiedź: B.

Połowienie opiera się na założeniu, że wszystko po lewej stronie środka jest mniejsze, a po prawej większe, a w przypadkowej kolejności to założenie nie obowiązuje, więc wynik bywa błędny. Nie będzie to „ten sam wynik, tylko wolniej”, bo algorytm odrzuca połowę listy, w której szukany numer może stać. Algorytm sam niczego nie porządkuje, bo nie ma w nim takiego kroku.

Umiejętność: Wyszukiwanie po kolei i przez połowienie

5. Zbiór ma 1000 uporządkowanych pozycji. Ilu kroków potrzebuje połowienie zakresu w najgorszym przypadku?

5 Z 10, POZIOM ROZSZERZONY

- ☐ A. Do 500 kroków, czyli połowy zbioru
- ☐ B. Do 100 kroków
- ☒ **C. Do 10 kroków, bo każdy krok zmniejsza zakres o połowę**
- ☐ D. Dokładnie tyle samo co szukanie po kolei

Poprawna odpowiedź: C.

Podwajanie od 1 przekracza 1000 po dziesięciu krokach (1024), więc dziesięć połowień wystarczy, żeby zawęzić zakres do jednej pozycji. Odpowiedzi 500 i 100 zakładają, że praca rośnie razem ze zbiorem, a rośnie tylko o jeden krok przy podwojeniu zbioru. Tyle samo co po kolei wyszłoby tylko przy bardzo krótkiej liście.

Umiejętność: Wyszukiwanie po kolei i przez połowienie

6. Krok „sprawdź środek listy” oblewa test jednoznaczności. Dlaczego?

6 Z 10, POZIOM PODSTAWOWY

☒ **A. Bo przy 16 pozycjach środek może wypaść na 8 albo na 9 i wykonawca musi sam wybrać**

☐ **B. Bo lista ma za mało pozycji, żeby miała środek**

☐ **C. Bo środek listy wolno sprawdzać dopiero na końcu**

Poprawna odpowiedź: A.

Przy 16 pozycjach suma granic 1 i 16 daje 17, a 17 podzielone przez 2 to 8,5, więc bez reguły zaokrąglania wykonawca musi wybrać między pozycją 8 a 9, a wybór to zgadywanie. Lista 16 pozycji ma środek, tylko trzeba go jednoznacznie wyznaczyć, a kolejność sprawdzania nie ma tu nic do rzeczy.

Umiejętność: Zapis algorytmu w krokach

7. Uczeń zapisał szukanie po kolei tak: 1. Ustaw pozycję na 1. 2. Porównaj numer na tej pozycji z szukanym. 3. Jeśli równy, wypisz pozycję i zakończ. 4. Zwiększ pozycję o 1 i wróć do kroku 2. Wykonawca szuka numeru 1305 na liście 16 numerów z lekcji. Co się stanie?

7 Z 10, POZIOM ROZSZERZONY

☐ **A. Po 16 porównaniach algorytm wypisze „nie ma” i zakończy się**

☐ **B. Algorytm zakończy się po 9 porównaniach, bo numer 1315 jest większy od 1305**

☐ **C. Algorytm wypisze pozycję 8, bo 1290 jest najbliższe 1305**

☒ **D. Po 16 porównaniach wykonawca ustawi pozycję na 17 i spróbuje porównać numer, którego nie ma, bo brakuje kroku kończącego dla końca listy**

Poprawna odpowiedź: D.

W zapisie brakuje kroku „jeśli pozycja to 16, wypisz nie ma i zakończ”, więc po ostatnim porównaniu krok 4 każe przejść na pozycję 17, której nie ma. Odpowiedź „nie ma” nie padnie, bo żaden krok jej nie wypisuje. Szukanie po kolei nie korzysta z uporządkowania, więc nie zatrzyma się na 1315, i nie wybiera numeru najbliższego, bo porównuje tylko na równość.

Umiejętność: Zapis algorytmu w krokach

- 8. Dwie pary wykonały ten sam zapis algorytmu połowienia dla numeru 1233 na liście 16 numerów z lekcji. Jedna para uzyskała 3 kroki, druga 4. Zapis mówił „ustaw środek na sumę dolnej i górnej podzieloną przez 2”, bez słowa o zaokrąglaniu. Co jest przyczyną różnicy?**

8 Z 10, POZIOM ROZSZERZONY

- ☐ A. Jedna z par pomyliła się przy porównaniu i policzyła o jeden krok za mało
- ☐ B. Numer 1233 stoi na liście na dwóch różnych pozycjach
- ☒ C. **Zapis nie był jednoznaczny: przy 17 podzielonym przez 2 zaokrąglenie w górę daje środek 9 i 3 kroki, a zaokrąglenie w dół daje środek 8 i 4 kroki**
- ☐ D. Połowienie zawsze daje inną liczbę kroków przy każdym wykonaniu

Poprawna odpowiedź: C.

Przy zaokrąglaniu w dół sprawdzane są pozycje 8, 4, 6, 7, czyli 4 kroki, a przy zaokrąglaniu w górę pozycje 9, 5, 7, czyli 3 kroki. Obie pary wykonały zapis poprawnie, tylko zapis dopuszczał dwa wykonania, a to oznacza, że nie był algorytmem. Numer 1233 stoi tylko na pozycji 7, a poprawnie zapisane połowienie przy tych samych danych zawsze daje tę samą liczbę kroków.

Umiejętność: Zapis algorytmu w krokach

- 9. Kiedy szukanie po kolei jest lepszym wyborem niż połowienie zakresu?**

9 Z 10, POZIOM PODSTAWOWY

- ☐ A. Gdy lista ma milion pozycji i szukamy w niej codziennie
- ☐ B. Gdy lista jest uporządkowana i szukamy w niej wiele razy
- ☒ C. **Gdy lista jest nieuporządkowana, a szukamy w niej tylko raz**

Poprawna odpowiedź: C.

Na liście nieuporządkowanej połowienie daje błędne wyniki, a porządkowanie jej dla jednego szukania kosztuje więcej niż przejrzenie wszystkiego po kolei. Milion pozycji przeszukiwanych codziennie to sytuacja, w której warto raz uporządkować zbiór i szukać połowieniem do 20 kroków. Lista uporządkowana i przeszukiwana wiele razy to również przypadek dla połowienia.

Umiejętność: Wyszukiwanie po kolei i przez połowienie

10. Na uporządkowanej liście 16 numerów z lekcji szukasz numeru 1087. Ile kroków potrzebuje szukanie po kolei, a ile połowienie zakresu, jeśli krok to jedno porównanie?

10 Z 10, POZIOM ROZSZERZONY

- ☐ **A.** Po kolei 16 kroków, połowienie 5 kroków
- ☒ **B.** Po kolei 2 kroki, połowienie 3 kroki, więc tym razem wygrywa szukanie po kolei
- ☐ **C.** Po kolei 2 kroki, połowienie 2 kroki, remis
- ☐ **D.** Po kolei 2 kroki, połowienie 4 kroki

Poprawna odpowiedź: B.

Numer 1087 stoi na pozycji 2, więc szukanie po kolei trafia w 2 porównaniach. Połowienie sprawdza pozycję 8 (1290, za dużo, górna na 7), potem 4 (1156, za dużo, górna na 3), potem 2 (1087, trafione), czyli 3 kroki. Wynik 16 i 5 dotyczy ostatniego numeru, a nie drugiego, remis nie wychodzi, bo połowienie potrzebuje trzeciego porównania, a 4 kroki wyszłyby tylko przy pomyłce w zaokrągleniu środka.

Umiejętność: Wyszukiwanie po kolei i przez połowienie

SKRYPT WIDEO DLA NAUCZYCIELA

Algorytm wyszukiwania w zbiorze danych

Nagranie powstaje na podstawie tego skryptu. Do czasu publikacji wideo możecie przeczytać skrypt przed lekcją.

0:00 DO 0:45

Przedstawiam cel lekcji dla nauczyciela. W 45 minut uczeń zapisze algorytm wyszukiwania jako listę kroków i porówna dwie metody szukania na tym samym zbiorze. Nie potrzebujecie umieć programować. Potrzebujecie tablicy, wydrukowanej karty pracy i laptopów z przeglądarką.

0:45 DO 1:40

Zaczynacie od gry. Zapisujecie liczbę od 1 do 1000 i pozwalacie klasie zgadywać, odpowiadając wyłącznie za mało albo za dużo. Kreska na tablicy za każde pytanie. Gra kończy się po kilkunastu pytaniach i to jest Wasz punkt odniesienia na całą lekcję. Nie tłumaczcie jeszcze niczego, niech klasa sama zobaczy liczbę kresek.

1:40 DO 2:50

Przechodzę przez pokaz. Wyświetlacie szesnaście numerów inwentarza uporządkowanych rosnąco i zapisujecie na tablicy dwa algorytmy numerowanymi krokami. Kluczowe jest to, żeby zapisać je krokami, a nie opisać zdaniami. Uczniowie mają je przepisać i potem wykonać dosłownie, więc każde niedopowiedzenie wróci do Was w pytaniu ucznia.

2:50 DO 3:50

Zwracam uwagę na warunek zakończenia. Wykonajcie połowienie dla numeru, którego na liście nie ma. Po czterech krokach dolna granica przekracza górną i algorytm odpowiada, że danej nie ma. To jest moment, w którym klasa rozumie, po co w ogóle zapisuje się taki warunek. W pracach uczniów brakuje go najczęściej.

3:50 DO 5:00

Omawiam pracę uczniów. Pierwsze osiem minut na papierze, w parach, jeden czyta kroki, drugi je wykonuje i nie wolno mu się niczego domyślać. Kolejne dwanaście minut w edytorze Scratch, gdzie para buduje skrypt zgadujący liczbę od 1 do 100. Edytor działa bez zakładania konta, a projekt zapisuje się na dysk poleceniem Plik, a potem Zapisz na swoim komputerze.

5:00 DO 5:55

Najczęstsza usterka w skrypcie to ustawienie dolnej granicy na wartość strzału zamiast na strzał plus 1. Zakres przestaje się wtedy kurczyć i program pyta w nieskończoność. Sprawdzajcie ten jeden blok przy każdej parze, bo to on odpowiada za większość zgłoszeń, że nie działa.

5:55 DO 6:40

Na sprawdzeniu zbieracie liczby kroków do wspólnej tabeli. Dla numeru z początku listy przewaga połowienia prawie znika i to jest najlepszy materiał na dyskusję. Potem pytacie o tysiąc i o milion pozycji. Odpowiedzi to do dziesięciu i do dwudziestu kroków, a klasa dochodzi do nich samodzielnie przez podwajanie.

6:40 DO 7:00

Zamykam. Wynikiem lekcji nie jest działający skrypt, tylko zapis algorytmu, który da się wykonać bez domyślania, i świadomość, że połowienie potrzebuje uporządkowanego zbioru. To zdanie prowadzi wprost do kolejnej lekcji o bazach danych.