

SCENARIUSZ LEKCJI

OPRACOWANY W RAMACH PROJEKTU:
INFORMATYKA – MÓJ SPOSÓB NA POZNANIE I OPISANIE ŚWIATA.
PROGRAM NAUCZANIA INFORMATYKI
Z ELEMENTAMI PRZEDMIOTÓW MATEMATYCZNO-PRZYRODNICZYCH

Autorzy scenariusza: [Paweł Królikowski](#), [Agnieszka Brzostek](#)

TEMAT LEKCJI:

Ciąg Fibonacciego jako szczególny przykład ciągu określonego rekurencyjnie. Przykłady rekurencji w informatyce

Streszczenie

Na zajęciach uczniowie poznają rekurencyjny sposób opisywania ciągów liczbowych. Głównym celem będzie jednak nabycie umiejętności analizowania zależności rekurencyjnych pod kątem zauważania własności ciągów. Na zajęciach omówiony zostanie ciąg Fibonacciego ze zwróceniem szczególnej uwagi na powszechność występowania liczb Fibonacciego w przyrodzie, architekturze, sztuce, muzyce. Istotą zajęć jest wskazanie powiązań między rekurencją matematyczną i rekurencją w ujęciu informatycznym. W tym celu uczniowie będą mieli za zadanie przedstawić rozwiązania poruszanych problemów rekurencyjnie w języku C++ oraz MS Excel. Oczywiście jest, że do każdego przykładu uczniowie powinni samodzielnie zdefiniować zależność rekurencyjną. Do omawianych zagadnień wymagana jest umiejętność budowania funkcji w języku C++.

Czas realizacji

3 x 45 minut

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:

4. Opracowywanie informacji za pomocą komputera, w tym: rysunków, tekstów, danych liczbowych, animacji, prezentacji multimedialnych i filmów.

Uczeń:

- wykorzystuje arkusz kalkulacyjny do obrazowania zależności funkcyjnych i do zapisywania algorytmów.

5. Rozwiązywanie problemów i podejmowanie decyzji z wykorzystaniem komputera, stosowanie podejścia algorytmicznego.

Uczeń:

- 5.1. analizuje, modeluje i rozwiązuje sytuacje problemowe z różnych dziedzin;
- 5.2. stosuje podejście algorytmiczne do rozwiązywania problemu;
- 5.3. formułuje przykłady sytuacji problemowych, których rozwiązanie wymaga podejścia algorytmicznego i użycia komputera;
- 5.4. posługuje się podstawowymi technikami algorytmicznymi;
- 5.9. stosuje rekurencję w prostych sytuacjach problemowych;
- 5.11. opisuje podstawowe algorytmy i stosuje:
 - a) algorytmy na liczbach całkowitych, np.:
 - reprezentacja liczb w dowolnym systemie pozycyjnym, w tym w dwójkowym i szesnastkowym,
 - sprawdzanie, czy liczba jest liczbą pierwszą, doskonałą,
 - rozkładanie liczby na czynniki pierwsze,
 - iteracyjna i rekurencyjna realizacja algorytmu Euklidesa,
 - iteracyjne i rekurencyjne obliczanie wartości liczb Fibonacciego,
 - wydawanie reszty metodą zachłanną,
- 5.13. stosuje metodę zstępującą i wstępującą przy rozwiązywaniu problemu;
- 5.21. przeprowadza komputerową realizację algorytmu i rozwiązania problemu;
- 5.23. stosuje podstawowe konstrukcje programistyczne w wybranym języku programowania, instrukcje iteracyjne i warunkowe, rekurencję, funkcje i procedury, instrukcje wejścia i wyjścia, poprawnie tworzy strukturę programu.

Podstawa programowa

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

Cele kształcenia – wymagania ogólne:

- II. Wykorzystanie i interpretowanie reprezentacji.
- IV. Użycie i tworzenie strategii.

Treści nauczania – wymagania szczegółowe:

- PR 5.1 Uczeń wyznacza wyrazy ciągu określonego wzorem rekurencyjnym.
- PP 5.2. Uczeń bada, czy dany ciąg jest arytmetyczny lub geometryczny.
- PP 5.3. Uczeń stosuje wzór na n -ty wyraz i na sumę n początkowych wyrazów ciągu arytmetycznego.
- PP 5.4. Uczeń stosuje wzór na n -ty wyraz i na sumę n początkowych wyrazów ciągu geometrycznego.

Cel

Umiejętność analizowania ciągów opisanych rekurencją. Umiejętność opisywania problemów zależnościami rekurencyjnymi. Umiejętność implementacji zależności rekurencyjnych w języku programowania C++ oraz w MS Excel.

Słowa kluczowe

ciąg, rekurencja, ciąg Fibonacciego, funkcja, C++, MS Excel

Co przygotować?

Prezentacja PowerPoint „Rekurencja w informatyce”

Materiały pomocnicze: Arkusz kalkulacyjny – fibonacci.xls, Program w C++ – fibonacci.cpp, Program w C++ – praca domowa1.cpp, Program w C++ – nwd.cpp, Arkusz kalkulacyjny – NWD.xls, Program w C++ – suma.cpp, Program w C++ – iloczyn.cpp, Program w C++ – silnia.cpp, Program w C++ – ciąg.cpp

Test wiedzy i umiejętności

Test wiedzy i umiejętności – rozwiązania

Arkusz kalkulacyjny – Test wiedzy i umiejętności

Zadania

Zadania – rozwiązania

Film – Tajemniczy ciąg Fibonacciego. Złota liczba. Boska proporcja (źródło: <http://www.youtube.com/watch?v=wb7kPaM8cfg>)

Materiał teoretyczny

Podczas analizowania własności funkcji istotną rolę odgrywa zbiór na jakim została ona określona czyli dziedziną funkcji. Jeśli ze wszystkich funkcji wyodrębnimy te, których dziedziną jest zbiór liczb naturalnych dodatnich, lub dowolny podzbiór kolejnych początkowych liczb naturalnych dodatnich, to taki rodzaj funkcji w matematyce nazywamy ciągami. Ta właściwość powoduje, że ciąg w matematyce można rozumieć jako listę uporządkowanych i ponumerowanych elementów pewnego zbioru. Bardziej naturalne staje się również określenie wartości ciągu. Zamiast mówić, że np. argumentowi dwa ciąg przyporządkowuje wartość 5, powiemy: „drugim wyrazem ciągu jest 5”.

Definicja 1.

Funkcję określoną na zbiorze liczb naturalnych dodatnich nazywamy nieskończonym ciągiem, wartości tej funkcji – wyrazami ciągu.

Definicja 2.

Ciągiem skończonym n-wyrazowym nazywamy funkcję określoną na zbiorze $\{1, 2, 3, \dots, n\}$.

Definicja 3.

Ciągiem liczbowym nazywamy ciąg, którego wyrazami są liczby rzeczywiste.



Dowolny ciąg liczbowy można określić na wiele sposobów, jednak najbardziej praktyczną metodą jest opisanie go za pomocą wzoru. Wzór na ogólny wyraz ciągu określa zależność między wartością wyrazu i miejscem na którym on stoi. Uniwersalność tego sposobu polega na tym, że możemy obliczyć wartość dowolnego wyrazu ciągu bez odnoszenia się do wartości wyrazów poprzednich. Ciągi można określać także wzorami rekurencyjnymi, podając najczęściej pierwszy wyraz ciągu oraz zależność między wartością wyrazu danego i wyrazu poprzedniego. Taki sposób więcej mówi nam o samej strukturze ciągu, wskazując jak on powstaje i jakie ma własności. Istotnym utrudnieniem w tym przypadku jest to, że do obliczenia wartości danego wyrazu musimy znać wartości poprzednich wyrazów ciągu.

Najciekawszym spośród ciągów określonych rekurencyjnie jest niewątpliwie ciąg Fibonacciego, nazwany tak na cześć włoskiego matematyka Leonarda z Pizy zwanego Fibonaccim. On to w swoim dziele *Liber abbaci* z 1202 r. określił wyżej wymieniony ciąg jako rozwiązanie zadania o rozmnażaniu się królików. Podstawowy ciąg Fibonacciego określony jest rekurencją:

$$\begin{cases} a_1 = 1 \\ a_2 = 1 \\ a_n = a_{n-1} + a_{n-2} \text{ dla } n \in N^+ \text{ i } n > 2 \end{cases}.$$

Z analizy rekurencji wynika, że każdy wyraz ciągu począwszy od drugiego powstaje z sumowania wartości dwóch wyrazów poprzednich. Wyrazy ciągu nazywa się liczbami Fibonacciego. Przy dzieleniu wyrazu tego ciągu przez wyraz poprzedni otrzymujemy liczbę oscylującą wokół wartości 1,618. W miarę zwiększania się wyrazów ciągu odchylenia od tej wartości odpowiednich ilorazów są coraz mniejsze. Wartość graniczną nazywamy złotą liczbą i oznaczamy symbolem Φ . Dokładna wartość złotej liczby to $\Phi = \frac{\sqrt{5}+1}{2}$.

Ciąg Fibonacciego występuje w wielu aspektach przyrody, architektury i sztuki.

Tak jak w przypadku funkcji, tak również w przypadku ciągów możemy określać ich monotoniczność. W matematyce definiujemy ciągi rosnące, malejące, stałe, nierosnące, niemalejące. Definicje te są jednak pojęciowo dużo prostsze. Intuicyjnie ciąg rosnący określimy jako ciąg, w którym każdy wyraz (oprócz pierwszego) jest większy od wyrazu poprzedniego.

Definicja 4.

- 1) Ciąg (a_n) jest ciągiem rosnącym $\Leftrightarrow \bigwedge_{n \in N^+} a_{n+1} > a_n$
- 2) Ciąg (a_n) jest ciągiem malejącym $\Leftrightarrow \bigwedge_{n \in N^+} a_{n+1} < a_n$
- 3) Ciąg (a_n) jest ciągiem stałym $\Leftrightarrow \bigwedge_{n \in N^+} a_{n+1} = a_n$
- 4) Ciąg (a_n) jest ciągiem nierosnącym $\Leftrightarrow \bigwedge_{n \in N^+} a_{n+1} \leq a_n$
- 5) Ciąg (a_n) jest ciągiem niemalejącym $\Leftrightarrow \bigwedge_{n \in N^+} a_{n+1} \geq a_n$

Ciągi: malejące, rosnące, niemalejące, nierosnące noszą wspólną nazwę ciągów monotonicznych, przy czym o ciągach malejących i rosnących mówimy, że są ciągami ściśle monotonicznymi.

Definicja 5.

Ciąg arytmetyczny to ciąg co najmniej trzywyrazowy, w którym każdy wyraz począwszy od drugiego powstaje poprzez dodanie do wyrazu poprzedniego stałej liczby r , zwanej różnicą ciągu.

Ciąg (a_n) nazywamy ciągiem arytmetycznym $\Leftrightarrow \forall_{r \in \mathbb{R}} \wedge_{n \in \mathbb{N}^+} a_{n+1} = a_n + r$.

Wzór ogólny ciągu jest postaci: $a_n = a_1 + (n - 1)r$.

Sumę n -początkowych wyrazów ciągu arytmetycznego określa wzór:

$$S_n = \frac{2a_1 + (n-1)r}{2} n = \frac{a_1 + a_n}{2} n.$$

W ciągu arytmetycznym każdy wyraz oprócz pierwszego (i ostatniego, jeśli ciąg jest ciągiem skończonym) jest średnią arytmetyczną wyrazu poprzedniego i następnego.

Definicja 6.

Ciąg (a_n) nazywamy ciągiem geometrycznym, jeśli jest co najmniej trzywyrazowy i każdy jego wyraz począwszy od wyrazu drugiego powstaje poprzez pomnożenie wyrazu poprzedniego przez stałą wielkość q zwaną ilorazem ciągu.

Ciąg (a_n) nazywamy ciągiem geometrycznym $\Leftrightarrow \forall_{q \in \mathbb{R}} \wedge_{n \in \mathbb{N}^+} a_{n+1} = a_n q$.

Wzór ogólny ciągu jest postaci: $a_n = a_1 q^{n-1}$.

Sumę n -początkowych wyrazów ciągu geometrycznego można obliczyć ze wzoru:

$$S_n = \begin{cases} n \cdot a_1 & \text{gdy } q = 1 \\ a_1 \cdot \frac{1-q^n}{1-q} & \text{gdy } q \neq 1 \end{cases}$$

W ciągu geometrycznym kwadrat każdego wyrazu oprócz wyrazu pierwszego (i ostatniego – jeśli ciąg jest skończony) jest iloczynem wyrazu poprzedniego i następnego.

Pojęcie rekurencji w informatyce

W informatyce rekurencja (łac. *recursio* = powracanie) polega na wywoływaniu funkcji samej przez siebie, a więc uruchomienie funkcji odbywa się wewnątrz niej samej (argumentem funkcji jest ta sama funkcja).

Budowa rekurencji

Aby zdefiniować rekurencję należy spełnić dwa warunki:

1. Należy określić początkowy zbiór skończony – warunek kończący wywołania rekurencyjne.
2. Wartości następujących po sobie elementów ciągu wyznaczone są za pomocą poprzednich – tak zwane określenie wzoru rekurencyjnego

Rekurencja jest szeroko stosowaną techniką programistyczną. Wymaga ona jednak dobrej znajomości budowania funkcji np. w języku C++.

Przykładem rekurencji jest ciąg Fibonacciego omawiany w trakcie pierwszej lekcji. Przypomnijmy wzór rekurencyjny:

$$\text{Fibonacci: } \begin{cases} a_1 = a_2 = 1 \\ a_n = a_{n-1} + a_{n-2} \end{cases}$$

1. Początkowym zbiorem skończonym (warunkiem kończącym kolejne wywołania) jest:

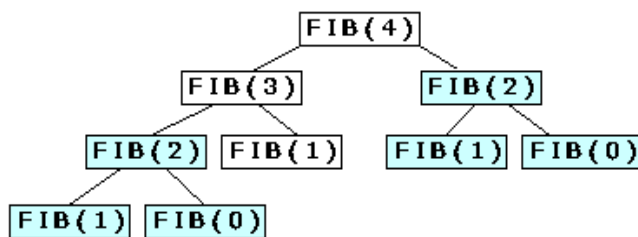
$$a_1 = 1$$

$$a_2 = 1$$

2. Kolejne wyrazy ciągu są wyznaczane na podstawie wzoru rekurencyjnego:

$$a_n = a_{n-1} + a_{n-2}$$

Kolejne wywołania rekurencyjne ciągu Fibonacciego:



Źródło: http://chem.arch.ug.edu.pl/informacje/dla_studentow/materialy/programowanie/rekurencja.html

Program w C++

Aby zbudować program w języku C++, należy zdefiniować funkcję określając poprawnie warunki kończące wywołania rekurencyjne.

Uwaga:

Źle zdefiniowane warunki doprowadzą do przepełnienia stosu (funkcja będzie się wywoływać w nieskończoność).

Funkcja Fibonacciego

```
int fibonacci (int n)
```

```
{
```

```
if (n==1 || n==2) return 1;
```

```
return fibonacci(n-1)+fibonacci(n-2);
```

```
}
```

Przebieg zajęć:

Lekcja 1

1. Wprowadzenie (5 minut)

Zapoznanie uczniów z tematyką zajęć. Powtórzenie definicji i własności ciągów: arytmetycznego i geometrycznego. Omówienie sposobów opisywania ciągów.

2. Rozwiązywanie zadań dotyczących ciągów określonych rekurencyjnie (20 minut)

Uczniowie pod nadzorem nauczyciela rozwiązują zadania umieszczone w pliku z zadaniami. Jedna osoba przedstawia rozwiązania na tablicy.

3. Omówienie ciągu Fibonacciego. Podanie definicji rekurencyjnej (15 minut)

Nauczyciel podaje rekurencyjną definicję ciągu Fibonacciego oraz omawia własność wyrazów tego ciągu wprowadzając pojęcie złotej liczby. Następnie uczniowie wykorzystując arkusz kalkulacyjny obliczają dowolne wyrazy ciągu Fibonacciego i na tej podstawie budują jego wykres (fibonacci.xls).

4. Podsumowanie danego etapu lekcji (5 minut)

Pracą domową jest wyszukanie informacji dotyczących występowania ciągu Fibonacciego w otaczającym nas świecie.

Lekcja 2

1. Nawiązanie do poprzedniej części zajęć (20 minut)

W tej części zajęć uczniowie dzielą się wyszukаныmi informacjami dotyczącymi występowania ciągu Fibonacciego. Podsumowaniem tej części zajęć będzie obejrzenie fragmentów filmu: „Tajemniczy ciąg Fibonacciego. Złota liczba. Boska proporcja” (<http://www.youtube.com/watch?v=wb7kPaM8cfg>).

2. Praca w grupach (20 minut)

Ta część zajęć poświęcona jest na grupowe opracowanie programu realizującego obliczanie dowolnych wyrazów ciągu Fibonacciego. Przed przystąpieniem do pracy nauczyciel na podstawie ciągu Fibonacciego omawia pojęcie rekurencji w ujęciu informatycznym oraz przedstawia sposób budowania zależności rekurencyjnych (prezentacja „Rekurencja w informatyce, fibonacci.cpp). Uczniowie pracują w dwuosobowych grupach.



3. Zadanie pracy domowej (5 minut)

Napisać program do przykładu nr 1 z lekcji 1

$$\text{Przykład z lekcji} = \begin{cases} a_1=3 \\ a_n=2 \cdot a_{n-1} - 5 \cdot a_{n-1} + 1 \end{cases}$$

Rozwiązanie pracy domowej – plik: praca domowa 1.cpp

Lekcja 3

1. Sprawdzenie pracy domowej (5 minut)

Wybrany uczeń przedstawia rozwiązanie zadania z pracy domowej.

2. Część ćwiczeniowa (35 minut)

Przed przystąpieniem do pracy nauczyciel przypomina pojęcie największego wspólnego dzielnika dwóch liczb naturalnych oraz algorytm jego wyznaczania. Następnie uczniowie zapisują omawiany algorytm używając rekurencji z uwzględnieniem warunków początkowych. Kolejnym etapem jest samodzielne opracowanie algorytmu wyznaczania nwd dwóch liczb naturalnych w Ms Excel (NWD.xls) oraz implementacja go w języku C++ (nwd.cpp). Następnie uczniowie analizują poprawność napisanego przez siebie programu.

3. Podsumowanie zajęć i zadanie pracy domowej (5 minut)

Pracą domową jest rozwiązanie testu Test wiedzy i umiejętności.

Ocenianie

Ocenie podlega poprawność rozwiązania testu zadanego w ramach pracy domowej.

Dostępne pliki

Zadania

Zadania – rozwiązania

Prezentacja „Rekurencja w informatyce”

Materiały pomocnicze: Arkusz kalkulacyjny – fibonacci.xls, Program w C++ – fibonacci.cpp, Program w C++ – praca domowa1.cpp, Program w C++ – nwd.cpp, Arkusz kalkulacyjny – NWD.xls, Program w C++ – suma.cpp, Program w C++ – iloczyn.cpp, Program w C++ – silnia.cpp, Program w C++ – ciąg.cpp

Test wiedzy i umiejętności

Test wiedzy i umiejętności – rozwiązania

Arkusz kalkulacyjny – Test wiedzy i umiejętności

Film – Tajemniczy ciąg Fibonacciego. Złota liczba. Boska proporcja (źródło: <http://www.youtube.com/watch?v=wb7kPaM8cfg>)