



Specyfikacja problemu algorytmicznego

- [Wprowadzenie](#)
- [Przeczytaj](#)
- [Schemat interaktywny](#)
- [Sprawdź się](#)
- [Dla nauczyciela](#)



Specyfikacja problemu algorytmicznego

Źródło: Rodion Kutsaev, domena publiczna.

W tym e-materiale powtarzamy wiadomości ze szkoły podstawowej.

Algorytm jest czymś formalnym i musi spełniać kilka własności. Nawet gdy czytasz przepis na potrawę, to te własności, co do zasady, są spełnione. Wiesz, jakie składniki są potrzebne do przygotowania potrawy oraz znasz wynik swoich działań, czyli samą potrawę. Wymienione elementy noszą miano specyfikacji i mają szczególne znaczenie w kontekście opisywania algorytmów. Poznaj przykłady specyfikacji problemu algorytmicznego.

Informacje o algorytmach znajdziesz również w pozostałych e-materiałach z serii:

- [Algorytmika](#),
- [Algorytmy na co dzień](#),
- [Własności algorytmów](#).

Więcej zadań z zakresu algorytmiki? Zajrzyj do [Algorytmika – zadania maturalne](#).

Twoje cele

- Wyjaśnisz, jak sformułować specyfikację algorytmu.
- Scharakteryzujesz, co powinna zawierać specyfikacja algorytmu.
- Przeanalizujesz podstawowe elementy specyfikacji algorytmu i wyjaśnisz, jak należy je rozumieć.

Przeczytaj

Specyfikacja problemu algorytmicznego

Każdy, nawet najprostszy **algorytm**, ma swoją **specyfikację** i rozwiązuje pewien problem. Każdy problem musi być jasno określony. Przykładowym rozwiązaniem może być obliczenie pierwiastka kwadratowego lub posortowanie listy jednowymiarowej. Problemy te mogą być ogólne, jak podane wcześniej, lub dużo bardziej specyficzne, jak choćby posortowanie listy jednowymiarowej, składającej się z liczb całkowitych dodatnich z przedziału od zera do pięciu tysięcy.

Dla każdego problemu musimy również wyznaczyć **dane wejściowe** oraz **wyjściowe**. Należy to zrobić w formalny sposób.

Najpierw jednak zastanówmy się nad tym, czy można przenieść te pojęcia do życia codziennego.

Przykłady specyfikacji algorytmu

Założmy, że jesteśmy inwestorami, którzy chcą postawić kilka domów. Problemem jest więc postawienie szeregu domów jednorodzinnych na danej działce budowlanej. W tym momencie, zależnie od poszczególnych kroków algorytmu, potrzebowalibyśmy zapewne co najmniej kilku informacji wejściowych.

Po pierwsze, musimy do algorytmu dostarczyć samą działkę budowlaną. Następnie powinniśmy dostarczyć projekt domków, będący osobnym algorytmem, który z kolei dla formy algorytmu nie ma znaczenia. Gdy już mamy projekt, musimy sobie zapewnić łańcuch dostaw materiałów budowlanych oraz podwykonawców – dla uproszczenia założmy, że wykonawca dostarczy wszystkie niezbędne materiały.

Wynikiem algorytmu będzie postawienie szeregu domów jednorodzinnych na wskazanej przez nas działce. Spróbujmy zapisać to w sposób formalny.

Przykład 1

Zapisz algorytm postawienia szeregu domów jednorodzinnych na wskazanej działce budowlanej.

Specyfikacja problemu:

Dane:

m – miejsce, na którym chcemy wybudować domki

- p – plan budowy domów oraz rozlokowania ich na działce
- b – podwykonawcy oraz materiały przez nich dostarczone

Wynik:

- d – domy wybudowane na działce
- n – liczba domów

Specyfikacja algorytmu jest bardzo uproszczona. Jednak wymienione przez nas elementy w danych wejściowych i wyjściowych są z pewnością konieczne.

Do samej specyfikacji problemu algorytmicznego nie musimy dostarczać kompletnej listy kroków – bardzo często to właśnie my, jako informatycy, taką listę kroków musimy sporządzić. Warto tutaj dodać, że jednym z typów zadań maturalnych jest zadanie, w którym do podanej specyfikacji algorytmu należy przygotować listę kroków.

Przykład 2

Zapisz algorytm obliczania pola prostokąta.

Specyfikacja problemu:

Dane:

- a – dłuższy bok prostokąta, liczba rzeczywista dodatnia
- b – krótszy bok prostokąta, liczba rzeczywista dodatnia

Wynik:

- p – pole prostokąta.

Algorytm na obliczanie pola prostokąta jest jednym z najbardziej podstawowych. Sprecyzowaliśmy, że a oraz b jako dane wejściowe muszą spełniać pewne warunki, w tym przypadku muszą być liczbami rzeczywistymi dodatnimi. Dodanie tej informacji może wydawać się zbędne, jednak określamy w ten sposób, że tylko takie dwie liczby a i b uznajemy za poprawne dane wejściowe.

Oczywiście możemy uwzględnić w algorytmie sytuację, w której ktoś podałyby inne dane i wtedy moglibyśmy również inaczej sformułować specyfikację algorytmu.

Pamiętajmy, że dla niepoprawnych danych wejściowych wynik działania algorytmu może nie być poprawny.

Przykład 3

Zapisz algorytm obliczania pierwiastka kwadratowego.

Specyfikacja problemu:

Dane:

- a – liczba całkowita

Wynik:

p – pierwiastek kwadratowy z liczby a

W powyższym przykładzie mamy dane wejściowe, co do których można mieć wątpliwości. Zapis *liczba całkowita* nie implikuje, że liczba ma być dodatnia.

Dlatego uzasadnione jest pytanie, czy algorytm w tej postaci zwróci poprawny wynik dla pierwiastka z liczby ujemnej.

Jednakże, z punktu widzenia formalnego, nie mamy podstaw do wątpliwości, ponieważ możemy obliczyć pierwiastek z liczby ujemnej (z użyciem liczb urojonych). Dlatego też algorytm z taką specyfikacją może istnieć i zwracać poprawne wyniki.

Ważne!

Czasami obok danych wejściowych i wyjściowych wymienia się również dane pomocnicze, które ułatwiają implementację algorytmu.

Słownik

algorytm

przepis postępowania prowadzący (w skończonym czasie) do rozwiązania ustalonego problemu

dane wejściowe (dane)

zbiór danych potrzebnych do rozpoczęcia pracy algorytmu; na ich podstawie oraz po wykonaniu kolejnych kroków algorytmu, otrzymywane są dane wyjściowe; definicja danych wejściowych podaje również ograniczenia dla tych danych, jeśli takie ograniczenia istnieją; dla algorytmu obliczania pola figury byłyby to liczby rzeczywiste dodatnie, oznaczające długości boków

dane wyjściowe (wynik)

zbiór danych będący wynikiem algorytmu; dla algorytmu obliczania pola figury byłoby to pole figury

specyfikacja problemu algorytmicznego (specyfikacja algorytmu)

opis problemu, który ma zostać rozwiązany wraz z danymi wejściowymi i danymi wyjściowymi

Schemat interaktywny

Polecenie 1

Zapisz za pomocą listy kroków algorytm, który wyznaczy iloczyn liczb od 1 do n (silnię z liczby n). Liczbę naturalną $n > 0$ podaje użytkownik.

Specyfikacja problemu:

Dane:

n – liczba naturalna

Wynik:

w – wynik obliczenia silni z liczby n ; liczba naturalna

Polecenie 2

Zapisz algorytm, wykorzystując wybrany język programowania.

1

1



Polecenie 3

Zapisz trzy przykłady innych problemów z lekcji matematyki, do rozwiązania których można użyć algorytmu.

Polecenie 4

Zapisz jeden z algorytmów wymienionych powyżej w postaci listy kroków. Przeanalizuj rozwiązanie osoby siedzącej najbliżej Ciebie.

1

Sprawdź się

Pokaż ćwiczenia:   

Ćwiczenie 1



Wskaż, co musi posiadać każda specyfikacja algorytmu. Zaznacz wszystkie poprawne odpowiedzi.

☐ przykładowe dane

☐ dane pomocnicze

☐ dane wejściowe

☐ dane wyjściowe

Ćwiczenie 2



Wskaż poprawne definicje.

☐ Dane wejściowe to zbiór danych będący wynikiem algorytmu.

☐ Dane wyjściowe to dane potrzebne do rozpoczęcia algorytmu i przetwarzane w jego trakcie.

☐ Dane wyjściowe to zbiór danych będący wynikiem algorytmu.

☐ Dane wejściowe to dane potrzebne do rozpoczęcia algorytmu i przetwarzane w jego trakcie.

Ćwiczenie 3



Zaznacz dane wejściowe, dane wyjściowe oraz opis problemu, który rozwiązuje dana specyfikacja.



dane wejściowe



dane wyjściowe



opis problemu

- Obliczenie pola trójkąta.
- a – długość boku trójkąta, h – wysokość trójkąta
- p – pole trójkąta

Ćwiczenie 4



Wskaż element nieobowiązkowy w specyfikacji algorytmu.



dane wejściowe



dane wyjściowe



opis problemu



dane pomocnicze

Ćwiczenie 5



Pogrupuj poniższe elementy, zgodnie z poniższym opisem problemu.

Oblicz pole trójkąta, posiadając jedynie informacje na temat położenia jego wierzchołków w układzie kartezjańskim.

dane wejściowe

b – punkt w układzie kartezjańskim (różne od a i c)

p – pole trójkąta

dane wyjściowe

a – punkt w układzie kartezjańskim (różne od b i c)

c – punkt w układzie kartezjańskim (różne od a i b)

Ćwiczenie 6



Zaznacz, które stwierdzenie jest prawdziwe, a które fałszywe.

Stwierdzenie	Prawda	Fałsz
Każda specyfikacja algorytmu musi jasno określać, jaki problem rozwiązuje.	☐	☐
Każda specyfikacja algorytmu musi mieć co najmniej dwa argumenty wejściowe.	☐	☐
Istnieją poprawne specyfikacje algorytmów bez danych wyjściowych.	☐	☐

Ćwiczenie 7



Dokończ zdanie.

Specyfikacja problemu algorytmicznego to inaczej...

- ☐ definicja algorytmu.
- ☐ opis algorytmu.
- ☐ dane wejściowe i wyjściowe.
- ☐ specyfikacja algorytmu.

Ćwiczenie 8



Zapisz specyfikację algorytmu obliczania średniej arytmetycznej dla ciągu n liczb.

Dane:

Wynik:

Dla nauczyciela

Autor: Zespół autorski Contentplus.pl sp. z o.o.

Przedmiot: Informatyka

Temat: Specyfikacja problemu algorytmicznego

Grupa docelowa:

Liceum ogólnokształcące i technikum, Szkoła podstawowa IV-VIII, liceum ogólnokształcące, szkoła podstawowa, technikum, zakres podstawowy

Podstawa programowa:

Cele kształcenia – wymagania ogólne

- I. Rozumienie, analizowanie i rozwiązywanie problemów na bazie logicznego i abstrakcyjnego myślenia, myślenia algorytmicznego i sposobów reprezentowania informacji.
- II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera oraz innych urządzeń cyfrowych: układanie i programowanie algorytmów, organizowanie, wyszukiwanie i udostępnianie informacji, posługiwanie się aplikacjami komputerowymi.

Cele kształcenia – wymagania ogólne

- I. Rozumienie, analizowanie i rozwiązywanie problemów na bazie logicznego i abstrakcyjnego myślenia, myślenia algorytmicznego i sposobów reprezentowania informacji.
- II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera oraz innych urządzeń cyfrowych: układanie i programowanie algorytmów, organizowanie, wyszukiwanie i udostępnianie informacji, posługiwanie się aplikacjami komputerowymi.

Treści nauczania – wymagania szczegółowe

- I. Rozumienie, analizowanie i rozwiązywanie problemów.

Zakres podstawowy. Uczeń:

- 1) planuje kolejne kroki rozwiązywania problemu, z uwzględnieniem podstawowych etapów myślenia komputacyjnego (określenie problemu, definicja modeli i pojęć, znalezienie rozwiązania, zaprogramowanie i testowanie rozwiązania).

Treści nauczania – wymagania szczegółowe

Klasy IV–VI

I. Rozumienie, analizowanie i rozwiązywanie problemów. Uczeń:

2) formułuje i zapisuje w postaci algorytmów polecenia składające się na:

- a) rozwiązywanie problemów z życia codziennego i z różnych przedmiotów, np. liczenie średniej, pisemne wykonanie działań arytmetycznych, takich jak dodawanie i odejmowanie,
- b) osiągnięcie postawionego celu, w tym znalezienie elementu w zbiorze nieuporządkowanym lub uporządkowanym, znalezienie elementu najmniejszego i największego,

3) w algorytmicznym rozwiązywaniu problemu wyróżnia podstawowe kroki: określenie problemu i celu do osiągnięcia, analiza sytuacji problemowej, opracowanie rozwiązania, sprawdzenie rozwiązania problemu dla przykładowych danych, zapisanie rozwiązania w postaci schematu lub programu.

II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera i innych urządzeń cyfrowych. Uczeń:

1) projektuje, tworzy i zapisuje w wizualnym języku programowania:

- a) pomysły historyjek i rozwiązania problemów, w tym proste algorytmy z wykorzystaniem poleceń sekwencyjnych, warunkowych i iteracyjnych oraz zdarzeń,

2) testuje na komputerze swoje programy pod względem zgodności z przyjętymi założeniami i ewentualnie je poprawia, objaśnia przebieg działania programów;

Klasy VII i VIII

I. Rozumienie, analizowanie i rozwiązywanie problemów. Uczeń:

1) formułuje problem w postaci specyfikacji (czyli opisuje dane i wyniki) i wyróżnia kroki w algorytmicznym rozwiązywaniu problemów. Stosuje różne sposoby przedstawiania algorytmów, w tym w języku naturalnym, w postaci schematów blokowych, listy kroków;

2) stosuje przy rozwiązywaniu problemów podstawowe algorytmy:

- a) na liczbach naturalnych: bada podzielność liczb, wyodrębnia cyfry danej liczby, przedstawia działanie algorytmu Euklidesa w obu wersjach iteracyjnych (z odejmowaniem i z resztą z dzielenia),

3) przedstawia sposoby reprezentowania w komputerze wartości logicznych, liczb naturalnych (system binarny), znaków (kody ASCII) i tekstów;

4) rozwija znajomość algorytmów i wykonuje eksperymenty z algorytmami, korzystając z pomocy dydaktycznych lub dostępnego oprogramowania do demonstracji działania algorytmów;

5) prezentuje przykłady zastosowań informatyki w innych dziedzinach, w zakresie pojęć, obiektów oraz algorytmów.

II. Programowanie i rozwiązywanie problemów z wykorzystaniem komputera i innych urządzeń cyfrowych. Uczeń:

1) projektuje, tworzy i testuje programy w procesie rozwiązywania problemów. W programach stosuje: instrukcje wejścia/wyjścia, wyrażenia arytmetyczne i logiczne, instrukcje warunkowe, instrukcje iteracyjne, funkcje oraz zmienne i tablice. W szczególności programuje algorytmy z działu I pkt 2;

Kształtowane kompetencje kluczowe:

- kompetencje cyfrowe;
- kompetencje osobiste, społeczne i w zakresie umiejętności uczenia się;
- kompetencje matematyczne oraz kompetencje w zakresie nauk przyrodniczych, technologii i inżynierii.

Cele operacyjne (językiem ucznia):

- Wyjaśniesz, jak sformułować specyfikację algorytmu.
- Scharakteryzujesz, co powinna zawierać specyfikacja algorytmu.
- Przeanalizujesz podstawowe elementy specyfikacji algorytmu i wyjaśniesz, jak należy je rozumieć.

Strategie nauczania:

- konstruktywizm;
- konektywizm.

Metody i techniki nauczania:

- dyskusja;
- rozmowa nauczająca z wykorzystaniem multimediu i ćwiczeń interaktywnych;
- metody aktywizujące.

Formy pracy:

- praca indywidualna;

- praca w parach;
- praca w grupach;
- praca całego zespołu klasowego.

Środki dydaktyczne:

- komputery z głośnikami, słuchawkami i dostępem do internetu;
- zasoby multimedialne zawarte w e-materiale;
- tablica interaktywna/tablica, pisak/kreda.

Przebieg lekcji

Przed lekcją:

1. **Przygotowanie do zajęć.** Nauczyciel loguje się na platformie i udostępnia e-materiał: „Specyfikacja problemu algorytmicznego”. Uczniowie zapoznają się z treściami w sekcji „Przeczytaj”.

Faza wstępna:

1. Nauczyciel wprowadza uczniów szczegółowo w temat lekcji i jej cele. Może posłużyć się wyświetloną na tablicy zawartością sekcji „Wprowadzenie”.
2. **Rozpoznanie wiedzy uczniów.** Uczniowie tworzą pytania dotyczące tematu zajęć, na które odpowiedzą w trakcie lekcji.

Faza realizacyjna:

1. **Praca z tekstem.** Jeżeli przygotowanie uczniów do lekcji jest niewystarczające, nauczyciel prosi o indywidualne zapoznanie się z treścią zawartą w sekcji „Przeczytaj”. Każdy uczestnik zajęć podczas cichego czytania wynotowuje najważniejsze kwestie poruszane w tekście.
2. **Praca z multimedium.** Nauczyciel wyświetla zawartość sekcji „Schemat interaktywny”, wybrany uczeń czyta treść polecenia nr 1 „Jeśli chcesz powtórzyć wiadomości ze szkoły podstawowej, za pomocą bloków stwórz program, który obliczy silnię z liczby n” i omawia przykładowe rozwiązanie postawionego problemu.
3. **Ćwiczenie umiejętności.** Liga zadaniowa - uczniowie wykonują indywidualnie na czas ćwiczenia nr 1-8 z sekcji „Sprawdź się”, a następnie omawiają zadania na forum.

Faza podsumowująca:

1. Wybrany uczeń podsumowuje zajęcia, zwracając uwagę na nabyte umiejętności, omawia ewentualne problemy podczas rozwiązywania ćwiczeń.

Praca domowa:

1. Uczniowie proponują alternatywny sposób rozwiązania problemów postawionych w sekcji „Przeczytaj”.

Wskazówki metodyczne:

- Nauczyciel może przedyskutować z uczniami, dlaczego przepis kulinarny nie jest algorytmem.