

są zrozumiałe dla użytkownika, a kolejność ich wykonania jest jednoznacznie określona. Uniwersalność (ogólność) algorytmu wiąże się ściśle z problemem, jaki ten algorytm rozwiązuje. Chodzi o to, by algorytm umożliwiał rozwiązywanie wszystkich zadań zgodnych ze specyfikacją problemu, a różniących się jedynie danymi początkowymi.

Formalne przeprowadzenie dowodu poprawności algorytmu wymaga wiedzy matematycznej wykraczającej poza wymagania programowe dla szkół ponadgimnazjalnych. Trzeba bowiem dowieść, że pewne stwierdzenia zachodzą np. dla nieskończonej liczby poprawnych danych i że ciąg działań jest skończony.

Mimo że nie przeprowadzamy formalnego dowodu poprawności algorytmów, powinniśmy zawsze starać się dokładnie uzasadniać wszystkie ich kroki.



Przykład 5. Algorytm obliczania silni (realizacja iteracyjna)

Zadanie: Oblicz silnię liczby naturalnej n , stosując algorytm iteracyjny.

Dane: Dowolna liczba naturalna n .

Wynik: Wartość silni liczby naturalnej n (*silnia*).

Lista kroków:

1. Zaczynij algorytm.
2. Wprowadź wartość liczby n .
3. Zmiennej *silnia* przypisz wartość 1: $\text{silnia} := 1$.
4. Zmiennej i przypisz wartość 1: $i := 1$.
5. Zmiennej *silnia* przypisz jej wartość pomnożoną przez i : $\text{silnia} := \text{silnia} \cdot i$.
6. Powiększ wartość zmiennej i o 1: $i := i + 1$.
7. Jeśli $i > n$,
 przejdź do kroku 8,
 w przeciwnym wypadku przejdź do kroku 5.
8. Wyprowadź wynik: *silnia*.
9. Zakończ algorytm.

Ocena poprawności tego algorytmu powinna polegać na wykazaniu, że:

- jeśli dla każdej danej spełniającej warunek początkowy (n jest liczbą naturalną) w każdym wykonaniu algorytmu obliczenia dochodzą do końca, to wynik (*silnia*) spełnia warunek końcowy ($\text{silnia} = n!$);
- dla wszystkich danych spełniających warunek początkowy obliczenia w algorytmie kończą się.

Formalny dowód poprawności algorytmu obliczania silni pominiemy. Można jednak dokonać kilku obserwacji:

1. Algorytm daje poprawny wynik dla $n = 0$, $n = 1$ i $n = 2$. Stosując rozumowanie indukcyjne, można wywnioskować, że będzie dawał poprawne wyniki także dla $n > 2$.
2. Warunkiem zakończenia obliczeń jest $i > n$. Dla $n = 0$ i $n = 1$ warunek ten jest spełniony już przy pierwszym wykonaniu kroku 7. Dla $n \geq 2$ będzie on również spełniony, ponieważ jeśli w kroku 7. $i \leq n$, to następuje przejście do kroku 5. Ostatecznie więc po pewnej liczbie kroków algorytmu i przyjmie wartość większą od n , a wtedy nastąpi przejście do kroku 8. i zakończenie algorytmu.

Biorąc pod uwagę te obserwacje, można wywnioskować, że algorytm jest poprawny. Zaznaczamy jednak, że z formalnego punktu widzenia nie jest to dowód na poprawność algorytmu.