**Temat : Podstawy programowania.**

Program komputerowy i jego działanie jest w rękach i głowie programisty. Programista musi przewidzieć wszystkie możliwe zachowania użytkownika.

1. Problem – jest to zadanie do rozwiązania postawione przed programistą (np. Jaki będzie koszt pomalowania pomieszczenia? )
2. Algorytm – przepis na rozwiązanie problemu w skończonej liczbie operacji elementarnych . Przepis na przygotowanie potrawy to nic innego tylko algorytm.
3. Cechy algorytmu:
   1. skończony – skończona liczba kroków,
   2. jednoznaczny (poprawny) – dla takich samych danych takie same wyniki,
   3. szczegółowy – wykonawca (człowiek, maszyna) rozumie opisane czynności i potrafi je wykonać,
   4. uniwersalny – rozwiązuje grupę zadań a nie tylko dla jednych, określonych wartości (np. oblicza pole dowolnego prostokąta a nie tylko jednego o określonych bokach)
   5. zupełny – uwzględnia wszystkie możliwe teoretycznie warianty.
4. Schemat blokowy – jest to graficzne przedstawienie algorytmu. Każdy schemat rozpoczyna owal z poleceniem START a kończy owal z poleceniem STOP.
5. Program komputerowy - jest to pewna liczba rozkazów (poleceń wykonania operacji elementarnej), które ma wykonać komputer. Wykona je w takiej kolejności w jakiej umieścimy je w programie. Istnieją cztery podstawowe typy rozkazów: pisz, czytaj, oblicz, porównaj.
6. Języki programowania
   1. język niskiego poziomu – to taki [język programowania](https://pl.wikipedia.org/wiki/J%C4%99zyk_programowania), w którym jednej operacji elementarnej odpowiada najczęściej jedna operacja elementarna rzeczywistego procesora komputera, często nazywany jest językiem maszynowym,
   2. język wysokiego poziomu - ([ang.](https://pl.wikipedia.org/wiki/J%C4%99zyk_angielski) high-level programming language) – to taki typ [języka programowania](https://pl.wikipedia.org/wiki/J%C4%99zyk_programowania), którego składnia i [słowa kluczowe](https://pl.wikipedia.org/wiki/S%C5%82owo_kluczowe_(informatyka)) mają maksymalnie ułatwić rozumienie kodu programu przez człowieka, tym samym zwiększając poziom [abstrakcji](https://pl.wikipedia.org/wiki/Abstrakcja_(programowanie)) i dystansując się od sprzętowych niuansów. Większość z nich można zaliczyć do języków uniwersalnych, służących do rozwiązywania szerokiego zakresu problemów. Kod napisany w języku wysokiego poziomu nie jest bezpośrednio „zrozumiały” przez [procesor](https://pl.wikipedia.org/wiki/Procesor) komputera – większość kodu stanowią normalne słowa, jak w [języku naturalnym](https://pl.wikipedia.org/wiki/J%C4%99zyk_naturalny). Aby umożliwić wykonanie programu napisanego w takim języku należy wpierw przetłumaczyć go na język maszynowy czyli poddać go procesowi [kompilacji](https://pl.wikipedia.org/wiki/Kompilator). Do języków wysokiego poziomu zalicza się m.in. [Pascal](https://pl.wikipedia.org/wiki/Pascal_(j%C4%99zyk_programowania)), [C](https://pl.wikipedia.org/wiki/C_(j%C4%99zyk_programowania)), [C++](https://pl.wikipedia.org/wiki/C%2B%2B), [Logo](https://pl.wikipedia.org/wiki/Logo_(j%C4%99zyk_programowania)), [BASIC](https://pl.wikipedia.org/wiki/BASIC), Python