



Komputerowe wspomaganie prac inżynierskich

Dane, typy danych, struktury danych

Zakres zajęć

- Dane – informacje – wiedza.
- Standardowe typy danych.
- Niestandardowe typy danych.
- Struktury danych.
- Kolejowanie w komputerowym przetwarzaniu danych.



Dane, Informacje, Wiedza

Dane – informacja – wiedza

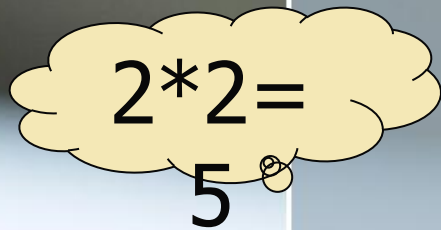
Istotne jest oddzielenie od siebie pojęć:

DANE, INFORMACJA,
WIEDZA



Problemy interpretacyjne

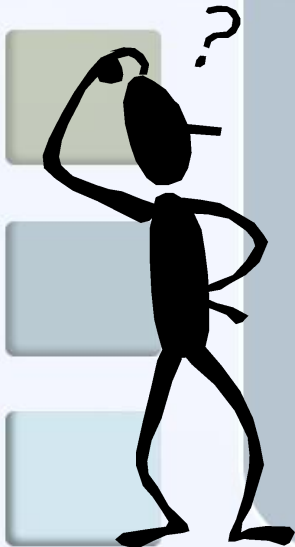
Analizując określony przypadek zawsze należy zastanowić się nad odpowiedziami na dwa zasadnicze pytania:



- dla **kogo** konkretna informacja/wiedza jest przeznaczona?
- w **jakim celu** informacja/wiedza jest generowana?

Cel pozyskiwania informacji/wiedzy implikuje:

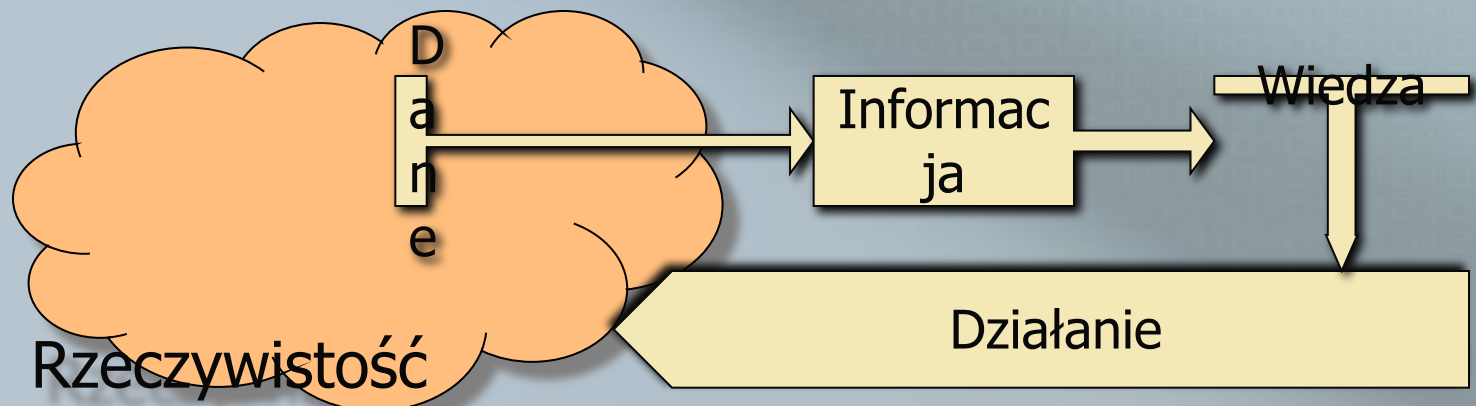
- postać informacji/wiedzy,
- jej zakres,
- zawartość,
- cenę.



Dane – informacja – wiedza

P. Beynon-Davies stwierdza, że:

- dane to **fakty**,
- informacja to **zinterpretowane dane**,
- wiedza jest **wynikiem zintegrowania informacji z wiedzą istniejącą**,
- informacja ma **charakter subiektywny**.



Aspekty jakościowy i ilościowy

Aspekt jakościowy

1. Dane stanowią odwzorowanie pewnego obszaru rzeczywistości.
2. Informacja stanowić będzie nadanie owemu obrazowi określonej wartości jakościowej, wynikającej z przetworzenia danych.
3. Wiedza odnosi się do sfery praktycznego wykorzystania tegoż obrazu w jego nowej postaci i ewentualnej modyfikacji istniejącej rzeczywistości (poprzez działanie).

Aspekt ilościowy

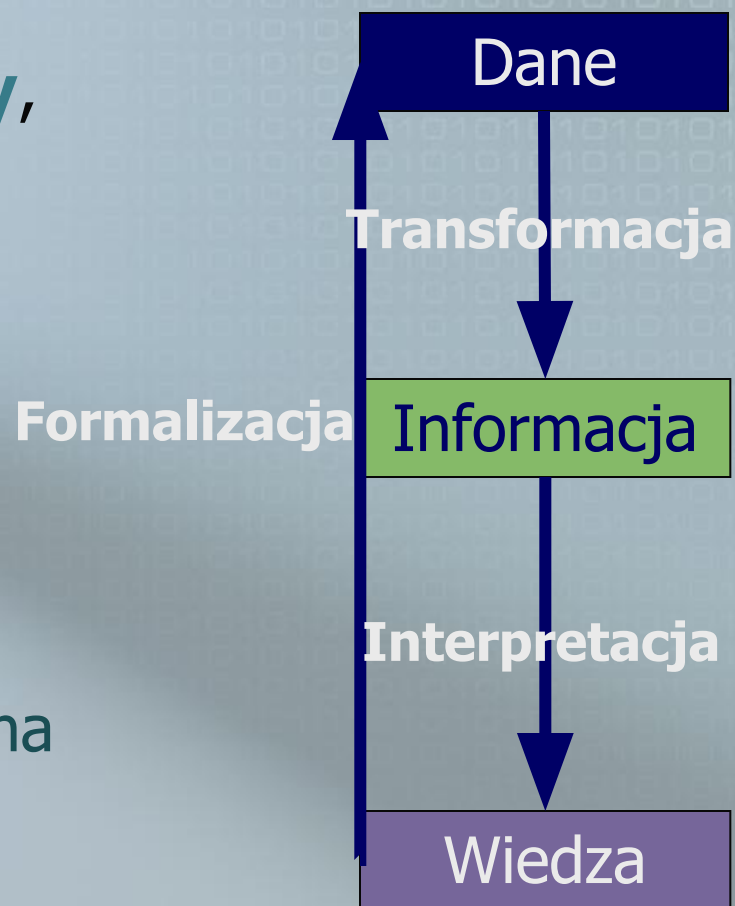
1. Informacja to określony ilościowo zbiór danych istotnych dla odbiorcy z punktu widzenia podwyższania jego zasobu (ilości) wiedzy, który jest pozyskiwany przez niego w określonej jednostce czasu.
2. W zależności od zapotrzebowania, określony zbiór danych będzie informacją o odpowiednio bogatej zawartości i odpowiednio pełnym zakresie.

Oba aspekty są w równym stopniu zdeterminowane przez **czynnik ludzki**.

Okreźny obieg wiedzy

Przyjmując wstępne założenie, że **dane to fakty**, w **okreźnym obiegu wiedzy** dokonano następującej interpretacji pojęć:

- informacja to dane poddane procesowi transformacji,
- wiedza to zinterpretowana i zintegrowana z wiedzą istniejącą informacja.



Pojęcie **dane** można w konsekwencji zinterpretować jako **wiedzę** poddaną procesowi **formalizacji**.

5-C filters – przekształcanie danych w informacje

Pięć filtrów (ang. 5-C filters) przekształcenia danych w informacje wg Davenport'a i Prusak'a:

- **kontekstualizacja** (ang. Contextualization) – wiemy dla jakich zastosowań dane są zbierane;
- **kategoryzacja** (ang. Categorization) – znamy kategorie, w oparciu o które dokonywana jest analiza lub kluczowe komponenty danych;
- **kalkulacja** (ang. Calculate) – dane mogą być analizowane w oparciu o metody matematyczne lub statystyczne;
- **korekcja** (ang. Correction) – z danych są usuwane błędy;
- **kondensacja** (ang. Condensation) – dane mogą być łączone do bardziej zwartej postaci.

4-C filters – przekształcanie informacji w wiedzę

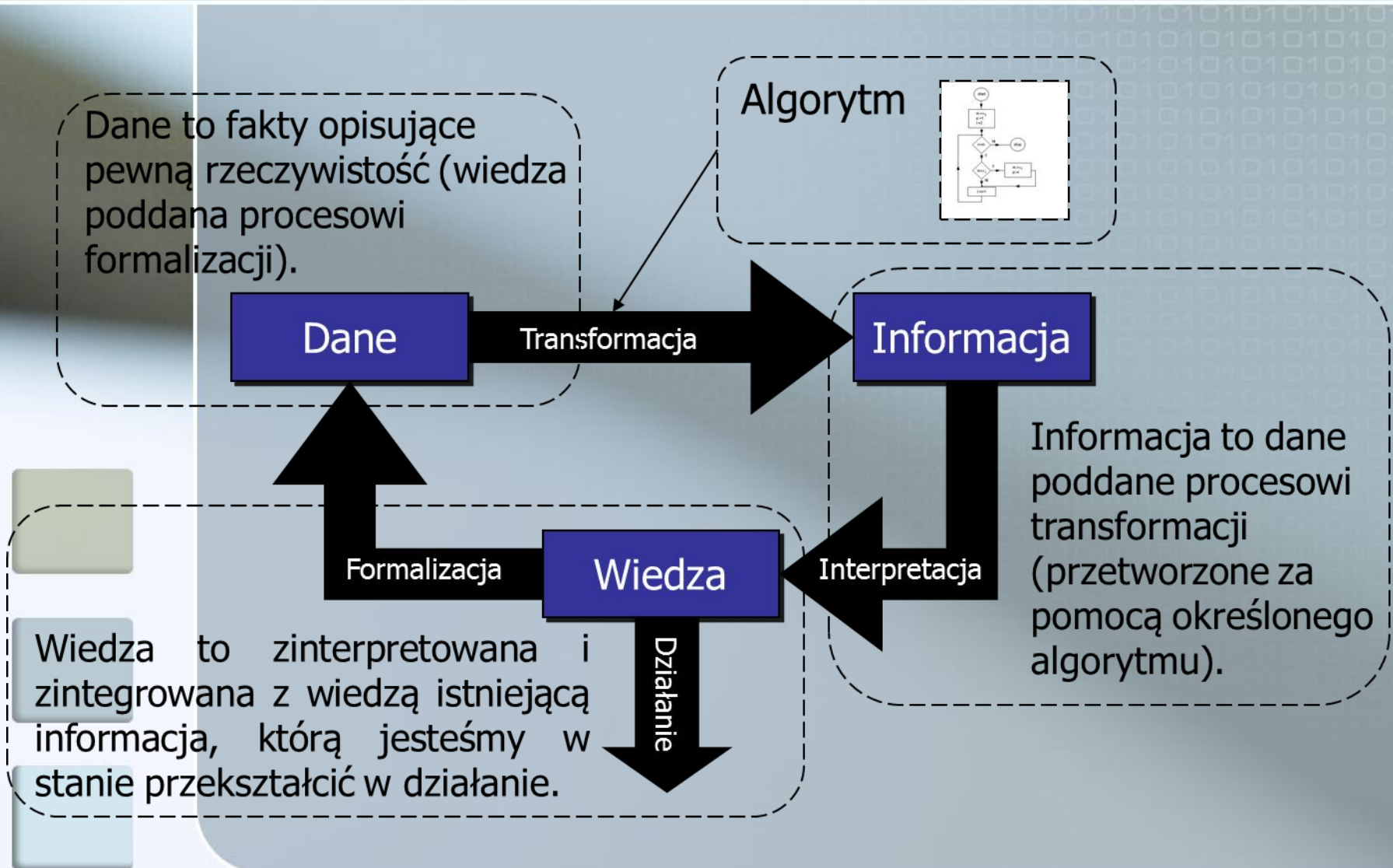
4 filtry (ang. 4-C filters) przekształcenia informacji w wiedzę wg Davenport'a i Prusak'a:

- **porównanie** (ang. Comparison) – jak informacja na temat danej sytuacji może być porównana do innych sytuacji, które znamy?
- **konsekwencje** (ang. Consequances) – jakie implikacje ma dana informacja dla podejmowanych decyzji i działań?
- **powiązanie** (ang. Connections) – jak dany fragment wiedzy odnosi się do innych?
- **konwersacja** (ang. Conversation) – co inni ludzie myślą o danej informacji?

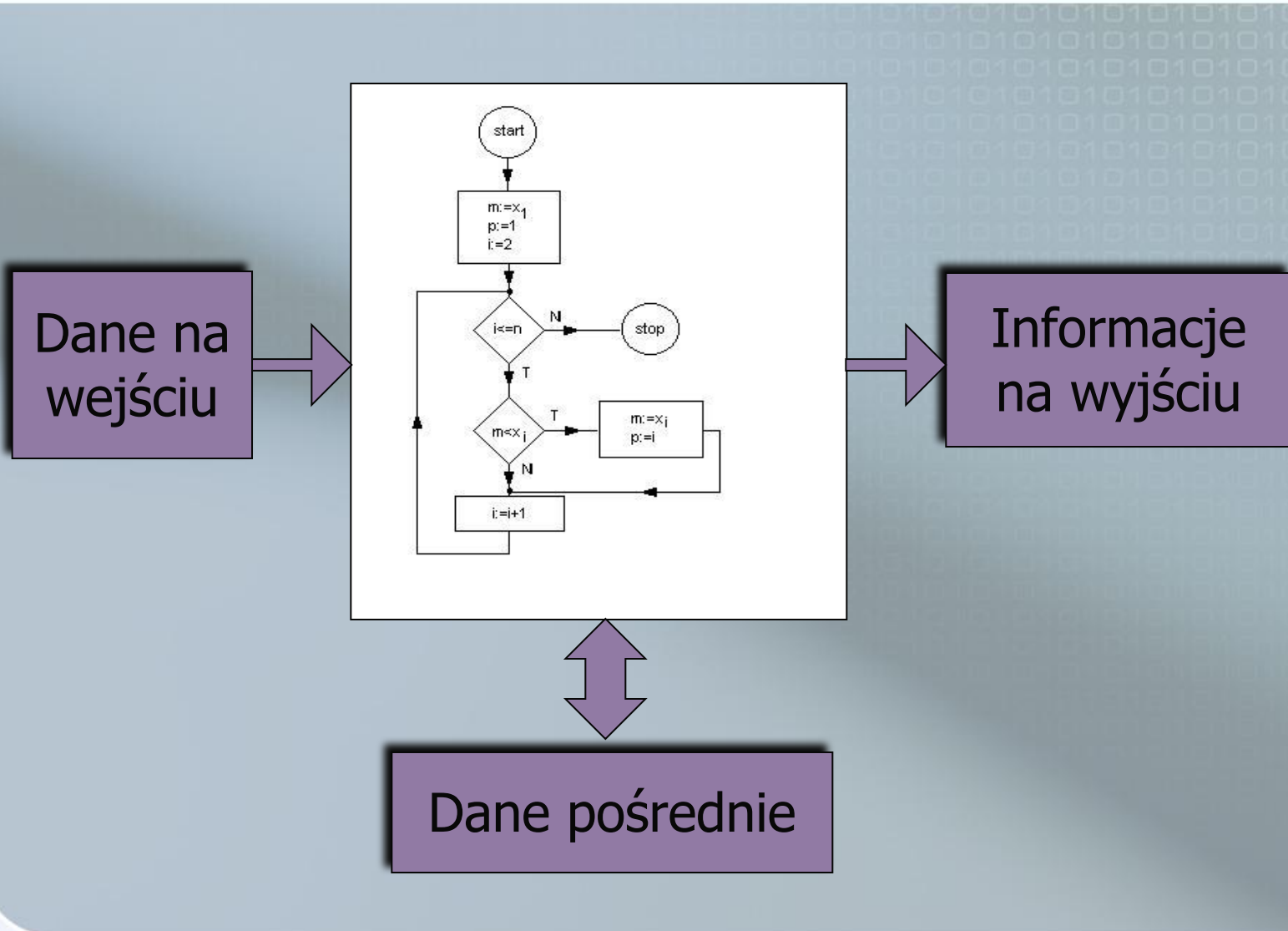
Piramida wiedzy



Dane – informacja – wiedza



Dane w systemie inform.





Typy danych

Typy danych

Typ danych określa zbiór wartości do jakiego należy określona dana oraz definiuje operacje, jakie mogą być na niej wykonywane.

Typy danych można podzielić na dwie zasadnicze grupy:

- **typy standardowe (podstawowe)**, zdefiniowane standardowo w określonym języku programowania;
- **typy niestandardowe** (zwane czasem **typami użytkownika**), definiowane przez programistę w zależności od struktury analizowanego problemu.

Podstawowe operacje na danych

Nazwa	Działanie	Przykład
<i>przypisanie</i>	Nadanie wartości obiektowi danych	X:=10 X:=Y
<i>porównanie</i>	Porównanie wartości dwóch obiektów danych	X=10, X==10, X!=10 X=Y, X==Y, X!=y
<i>konwersja</i>	Zmiana typu danych	Bin(x)

Standardowe typy danych

Standardowe (podstawowe) typy danych dzielą się na dwie zasadnicze grupy: **alfanumeryczne (tekstowe)** oraz **liczbowe**. Dodatkowo często wyodrębnianym typem danych jest typ **logiczny**, jednakże w praktyce ten typ danych może być reprezentowany przez liczby binarne.

Standardowe typy danych:

Tekstowe:

Znak

łańcuch znaków

Liczbowe:

Liczba całkowita

Liczba rzeczywista

Liczba binarna

Standardowe typy danych

Nazwa	Zakres	Rozmiar	Typ
bool	False lub True	1 B	binarny
char	Jeden znak	1 B	tekstowy
integer	-32768 do 32767	2 B	całkowity
unsigned int	0 to 65535	2 B	całkowity
long int	-2147483648 do 2147483647	4 B	całkowity
single	$1.5 \cdot 10^{-45}$ do $3.4 \cdot 10^{38}$	4 B	rzeczywisty
string(n)	Łańcuch n-znaków	n B	tekstowy

Standardowe typy danych w Ms Access

NAZWA TYPU	OPIS
TEKST	Typ znakowy o max liczbie n znaków; n - max 255
MEMO (NOTA)	Typ znakowy, max 65 535 znaków
LICZBA	<i>Bajt</i> – liczby naturalne od 0 do 255 <i>Liczba całkowita</i> – liczby całkowite od -32 768 do 32 767 <i>Liczba całkowita długa</i> – liczby całkowite od -2 147 483 648 do 2 147 647 <i>Pojedyncza precyzja</i> – liczby rzeczywiste od -3.402823E38 do 3.402823E38 <i>Podwójna precyzja</i> – liczby rzeczywiste od -1.7976931349E308 do 1.7976931349E308
DATA/GODZINA	Data i czas
WALUTOWY	Typ liczbowy, 2 miejsca po przecinku
AUTONUMER	Typ liczbowy; kolejne lub losowe liczby automatycznie generowane przez SZBD MS Access
TAK/NIE	Typ logiczny; przyjmujący tylko dwie wartości PRAWDA lub FAŁSZ
OLE OBJECT	Obiekt utworzony w dowolnej aplikacji posługującej się techniką OLE (np. obrazy, wykresy)
HIPERŁĄCZE	Typ znakowy tworzących adres hiperłącza do strony WWW, pliku lub miejsca w pliku

Standardowe typy danych liczbowych w Ms Access

Ustawienie w polu Rozmiar pola	Zakres	Miejsca dziesiętne	Rozmiar w pamięci
Bajt	0 do 255	Brak	1 bajt
Liczba całkowita	-32 768 do 32 767	Brak	2 bajty
Liczba całk. długa	-2 147 483 648 do 2 147 483 647	Brak	4 bajty
Double	-1.797×10^{308} do 1.797×10^{308}	15	8 bajtów
Pojedyncza precyzja	-3.4×10^{38} do 3.4×10^{38}	7	4 bajty
Identyfikator replikacji	Nie dotyczy	Nie dotyczy	16 bajtów
Ułamki dziesiętne	Dokładność 1 – 28	15	8 bajtów

Niestandardowe typy danych

Niestandardowe typy danych definiowane są przez **programistę** w zależności od potrzeb i specyfiki problemu. Programista samodzielnie definiuje **nazwę typu danych** oraz **zbiór wartości**, jakie dana tego typu może przyjąć.

Przykładowe typy użytkownika:

- **type** kształt = (*prostokąt, kwadrat, elipsa, okrąg*)
- **type** płeć = (*mężczyzna, kobieta*)
- **type** dzień tygodnia = (*poniedziałek, wtorek, środa, czwartek, piątek, sobota, niedziela*)
- **type** wzrost = (*niski, średni, wysoki*)

Specjalnym typem danych są **typy okrojone**, w których programista definiuje pewien zakres wartości dostępnych dla określonego typu danych, np.:

- **type** rok = *1900 .. 2100* (typ nadrzędny: *integer*)



Struktury danych

Struktury danych

Struktura danych to sposób reprezentacji danych w określonym języku programowania, służący do zapamiętania ich we właściwy sposób w pamięci komputera. Struktury te **organizują dane** tak, aby procesor był w stanie przeprowadzić na nich poszczególne operacje.

Najprostszą strukturą danych jest **zmienna** (*choć wielu autorów uważa, że zmienna powinna być utożsamiana z prostymi typami danych*), pozwalająca zapamiętać tylko jedną wartość w danej jednostce czasu.

Obiekty danych mogą być łączone w **struktury złożone** o dowolnym stopniu zagnieżdżenia (*często dopiero te właśnie konglomeraty danych są uznawane za struktury danych w pełnym tego słowa znaczeniu*).

zmienna

element

struktura złożona

element1

element2

..

elementN

Podstawowe struktury danych

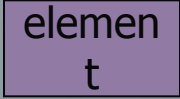
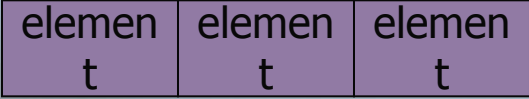
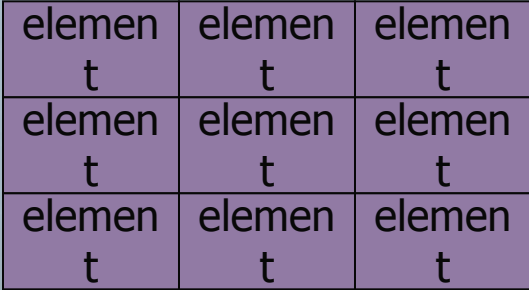
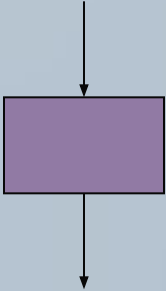
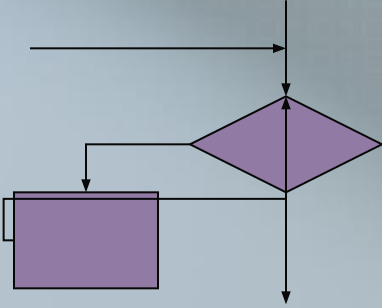
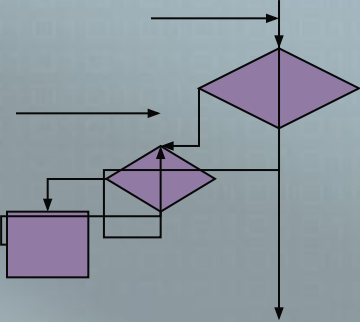
Do podstawowych złożonych struktur danych zalicza się:

- **wektory** (inaczej: **listy**), stanowiące **jednowymiarowe tablice**, zawierające zwykle dane tego samego typu;
- **macierze** (inaczej: **tablice dwuwymiarowe**), stanowiące konglomeraty danych zwykle tego samego typu, uporządkowane w kolumnach i wierszach;
- **rekordy**, pozwalające budować konglomeraty składające się z danych różnych typów, w tym innych typów złożonych.

Struktura	Przykład
<i>wektor</i>	type lista = array [1..10] of integer ;
<i>macierz</i>	type tablica = array [1..10,1..5] of integer ;
<i>rekord</i>	type osoba = record nazwisko, imie array [1..30] of char ; telefon: integer ;

Pewną specyficzną strukturą danych jest **plik**.

Podstawowe struktury danych, a struktury sterujące

	Zmienna	Wektor	Macierz
Struktura danych			
Struktura sterująca	<p><i>instrukcja</i></p> 	<p><i>pętla</i></p> 	<p><i>pętla zagnieżdżona</i></p> 



Inne struktury danych

Kolejka i kolejkowanie

Kolejka (ang. *queue*) to liniowa struktura danych, pozwalająca na przechowywanie zmieniających się w czasie danych i ich przetwarzanie w wybranej kolejności.

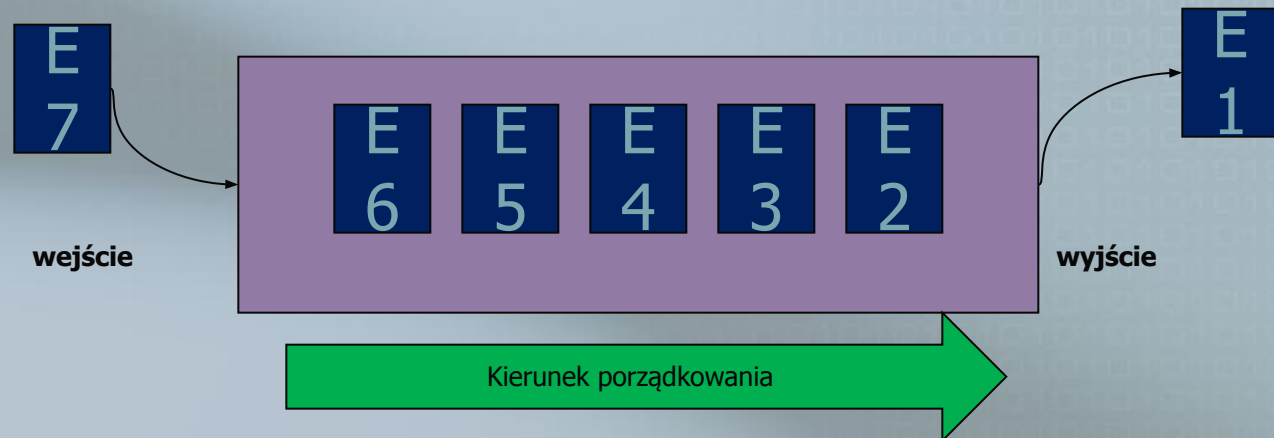
W systemach komputerowych stosuje się dwie podstawowe kolejki:

- kolejkę **FIFO**;
- kolejkę **LIFO**, nazywaną inaczej **stosem**.

Kolejkowanie (ang. *queuing*) to ogół działań wykonywanych przez system operacyjny lub oprogramowanie użytkowe związanych z obsługą kolejek.

Kolejki FIFO

Kolejka FIFO (ang. *First In – First Out*) jest specjalnym rodzajem listy, udostępniającej na wyjściu jako pierwszy do pobrania ten element, który został wcześniej jako pierwszy na liście umieszczony.

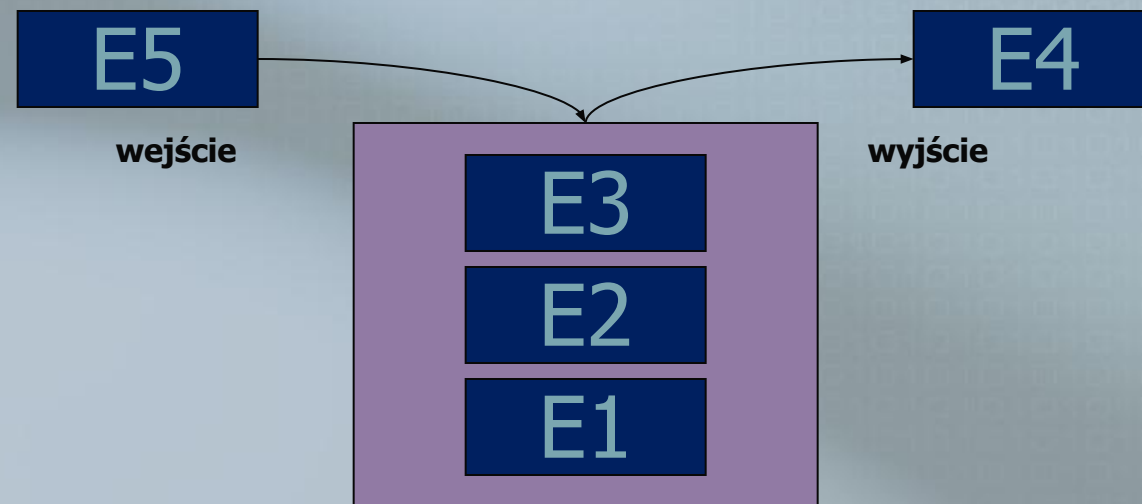


Specjalnym rodzajem kolejki FIFO jest **kolejka priorytetowa**, w której każda ze znajdujących się w niej danych dodatkowo ma przypisany **priorytet** modyfikujący kolejność późniejszego wykonania.

Struktury te wykorzystywane są między innymi przez **systemy operacyjne do buforowania (szeregowania) zadań**.

Kolejki LIFO – stos

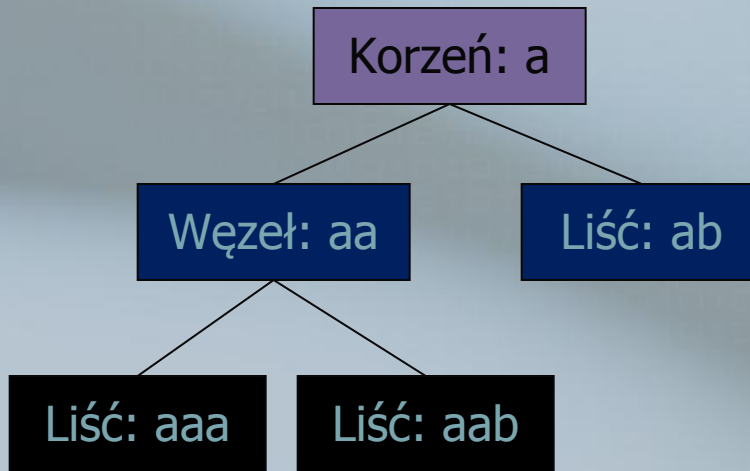
Kolejka LIFO (ang. *Last In – First Out*) nazywana zwykle **stosem** jest specjalnym rodzajem listy, udostępniającej na wyjściu jako pierwszy do pobrania ten element, który został na liście umieszczony jako ostatni.



Są to struktury bardzo często wykorzystywane. Stos umożliwia między innymi **zapamiętanie procesorowi miejsca powrotu w przypadku wykonywania podprogramu** (np. przy algorytmach rekurencyjnych).

Drzewa

Drzewa to specjalne **struktury hierarchiczne**, w których jeden element stanowi tzw. **korzeń**, natomiast pozostałe elementy są jego elementami **potomnymi**.



Tabelaryczna reprezentacja drzewa

<i>Kod</i>	<i>Wartość</i>	<i>Rodzic</i>
0	a	-
1	aa	0
2	ab	0
3	aaa	1
4	aab	1

Można wskazać wiele zastosowań drzew, np. **drzewa gry** stosowane w systemach sztucznej inteligencji lub **drzewa binarne** (w których każdy rodzic może posiadać co najwyżej dwoje potomków), stosowane w sortowaniu. Struktury te były również stosowane w **bazach danych**.

Do przemyślenia w domu

- Pojęcie bazy danych.
- Modele danych:
 - pojęcie modelu danych,
 - generacje modeli danych.
- Relacyjny model danych:
 - pojęcie relacji,
 - encje,
 - związki encji.
- Projektowanie baz danych.
- Hurtownie danych.





Dziękuję za uwagę

Zapraszam w przyszłym tygodniu