

# Informatyka 1 (EZ1E2008)

Politechnika Białostocka - Wydział Elektryczny  
Elektrotechnika, semestr II, studia niestacjonarne I stopnia  
Rok akademicki 2020/2021

## Wykład nr 8 (14.05.2021)

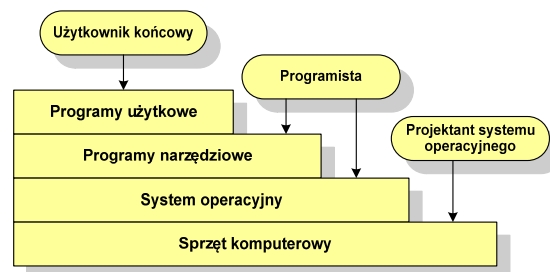
dr inż. Jarosław Forenc

## Plan wykładu nr 8

- System operacyjny
  - definicje systemu operacyjnego
- Zarządzanie procesami
  - definicja procesu, dwu- i pięciostanowy model procesu
- Zarządzanie dyskowymi operacjami we-wy
  - metody przydziału pamięci dyskowej
  - struktura dysku twardego (MBR, GPT)
  - systemy plików (FAT12, FAT16, FAT32, exFAT, NTFS)

## System operacyjny - definicja

- **System operacyjny** - jest to program sterujący wykonywaniem aplikacji i działający jako interfejs pomiędzy aplikacjami (użytkownikiem) a sprzętem komputerowym
- **użytkownik końcowy** nie jest zainteresowany sprzętem, interesują go tylko **aplikacje** (programy użytkowe)
- aplikacje są tworzone przez **programistów** za pomocą języków programowania



## System operacyjny - definicja

- System operacyjny - **administrator zasobów** - zarządza i przydziela zasoby systemu komputerowego oraz steruje wykonaniem programu
- **zasób systemu** - każdy element systemu, który może być przydzielony innej części systemu lub oprogramowaniu aplikacyjnemu
- do zasobów systemu zalicza się:
  - czas procesora
  - pamięć operacyjną
  - urządzenia zewnętrzne

## Zarządzanie procesami

- Głównym zadaniem systemu operacyjnego jest **zarządzanie procesami**
- Definicja procesu:
  - **proces** - program w trakcie wykonania
  - **proces** - ciąg wykonań instrukcji wyznaczanych kolejnymi wartościami licznika rozkazów wynikających z wykonywanej procedury (programu)
  - **proces** - jednostka, którą można przypisać procesorowi i wykonać
- Proces składa się z kilku elementów:
  - **kod programu**
  - **dane potrzebne programowi** (zmienne, przestrzeń robocza, bufory)
  - **kontekst wykonywanego programu** (stan procesu) - dane wewnętrzne, dzięki którym system operacyjny może nadzorować proces i nim sterować

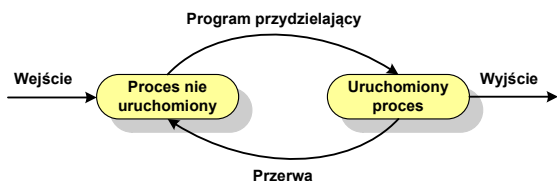
## Blok kontrolny procesu

- struktura danych tworzona i zarządzana przez system operacyjny, a opisująca właściwości procesu
- **identyfikator** - unikatowy numer skojarzony z procesem, dzięki któremu można odróżnić go od innych procesów
- **stan procesu**: nowy, gotowy, uruchomiony, zablokowany, anulowany
- **priorytet** - niski, normalny, wysoki, czasu rzeczywistego
- **licznik programu** - adres kolejnego rozkazu w programie, który ma zostać wykonany
- **wskaźniki pamięci** - wskaźniki do kodu programu, danych skojarzonych z procesem, dodatkowych bloków pamięci
- **dane kontekstowe** - dane znajdujące się w rejestrach procesora, gdy proces jest wykonywany
- **informacje na temat stanu żądań we-wy** - informacje na temat urządzeń we-wy przypisanych do tego procesu

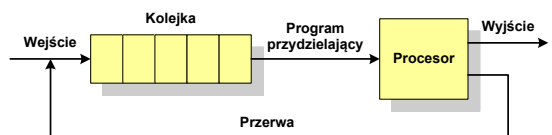
Identyfikator
Stan
Priorytet
Licznik programu
Wskaźniki pamięci
Dane kontekstowe
Informacje na temat stanu żądań we/wy
Informacje ewidencyjne
...

## Dwustanowy model procesu

- najprostszy model polega na tym, że w dowolnej chwili proces jest wykonywany przez procesor (**uruchomiony**) lub nie (**nie uruchomiony**)



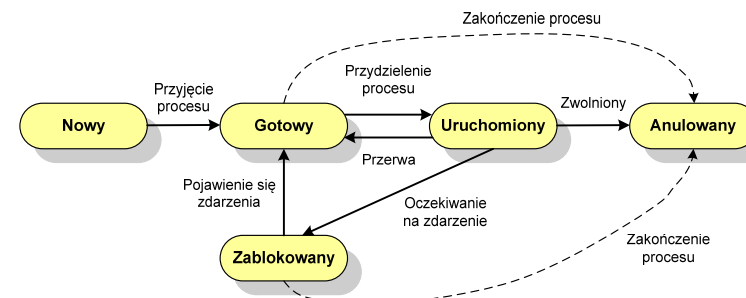
- procesy, które nie są uruchomione czekają w kolejce na wykonanie



- wadą tego modelu jest sytuacja, gdy kolejny proces pobierany do wykonania z kolejki jest **zablokowany**, gdyż oczekuje na **zakończenie operacji we-wy**

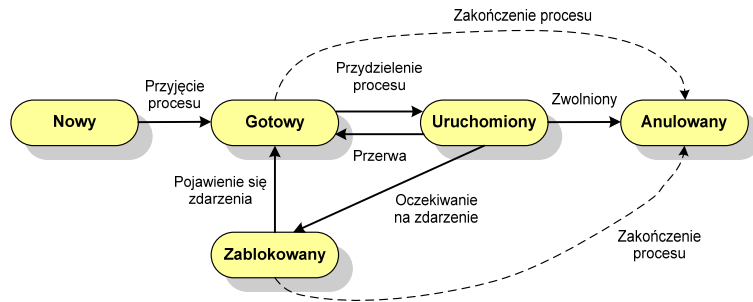
## Pięciostanowy model procesu

- rozwiązaniem powyższego problemu jest podział procesów nieuruchomionych na **gotowe do wykonania** i **zablokowane**



- pięciostanowy model procesu wymaga zastosowania minimum dwóch kolejek: dla procesów **gotowych do wykonania** i **zablokowanych**

## Pięciostanowy model procesu



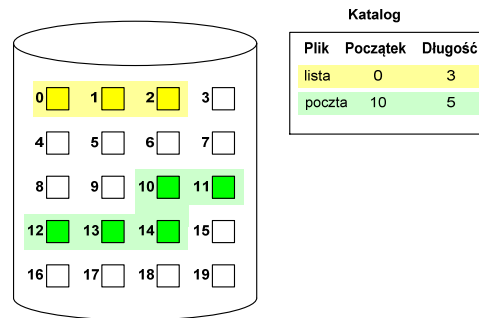
- **uruchomiony** - proces aktualnie wykonywany
- **gotowy** - proces gotowy do wykonania przy najbliższej możliwej okazji
- **zablokowany** - proces oczekujący na zakończenie operacji we-wy
- **nowy** - proces, który właśnie został utworzony (ma utworzony blok kontrolny procesu, nie został jeszcze załadowany do pamięci), ale nie został jeszcze przyjęty do grupy procesów oczekujących na wykonanie
- **anulowany** - proces, który został wstrzymany lub anulowany z jakiegoś powodu

## Zarządzanie dyskowymi operacjami we-wy

- Metody przydziału pamięci dyskowej (teoria)
  - alokacja ciągła
  - alokacja listowa
  - alokacja indeksowa
- Struktura dysku twardego
  - MBR (BIOS)
  - GPT (UEFI)
- Systemy plików (praktyka)
  - FAT (FAT12, FAT16, FAT32, exFAT)
  - NTFS
  - ext2

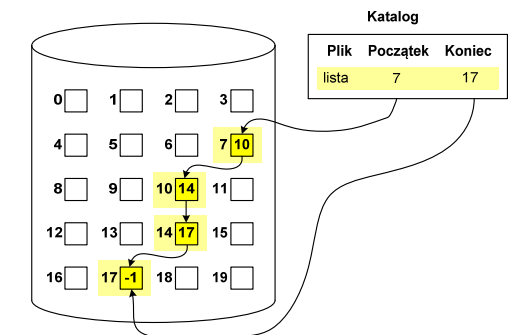
## Przydział pamięci dyskowej - alokacja ciągła

- każdy plik zajmuje ciąg kolejnych bloków na dysku
- plik zdefiniowany jest przez adres pierwszego bloku i ilość kolejnych zajmowanych bloków
- zalety: małe opóźnienia w transmisji danych, łatwy dostęp do dysku
- wady: trudność w znalezieniu miejsca na nowy plik



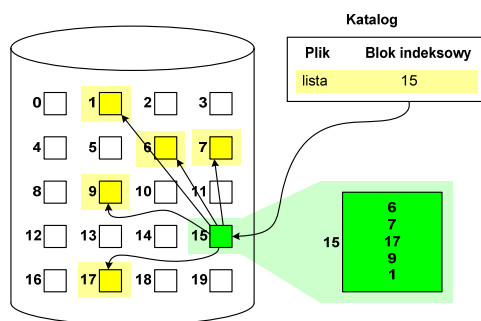
## Przydział pamięci dyskowej - alokacja listowa

- każdy plik jest listą powiązanych ze sobą bloków dyskowych, które mogą znajdować się w dowolnym miejscu na dysku
- w katalogu dla każdego pliku zapisany jest wskaźnik do pierwszego i ostatniego bloku pliku
- każdy blok zawiera wskaźnik do następnego bloku



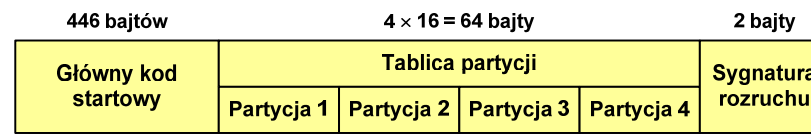
## Przydział pamięci dyskowej - alokacja indeksowa

- każdy plik ma własny blok indeksowy, będący tablicą adresów bloków dyskowych
- w katalogu zapisany jest dla każdego pliku adres bloku indeksowego



## Struktura dysku twardego - MBR

- MBR (Master Boot Record)** - główny rekord ładujący (1983, PC DOS 2.0)
- struktura danych opisująca podział dysku na partycje
- pierwszy sektor logiczny dysku (CHS → 0,0,1), zajmuje 512 bajtów



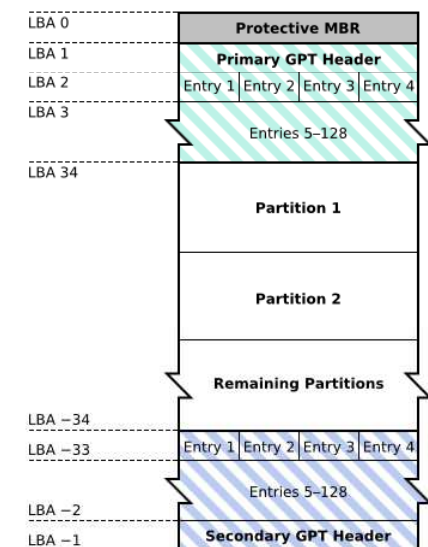
- główny kod startowy (Master Boot Code, bootloader)** - program odszukujący i ładujący do pamięci zawartość pierwszego sektora aktywnej partycji
- tablica partycji** - cztery 16-bajtowe rekordy opisujące partycje na dysku
  - zawartość i organizacja tablicy jest niezależna od systemu operacyjnego
  - maksymalny rozmiar partycji na dysku to **2 TB** ( $2^{32} \times 512$  bajtów)
- sygnatura rozruchu (boot signature)** - znacznik końca MBR (**0x55AA**)

## Struktura dysku twardego - GPT

- GPT (GUID Partition Table)** - standard zapisu informacji o partycjach na dysku twardym
- GUID (Globally Unique Identifier)** - 128-bitowa liczba stosowana do identyfikowania informacji w systemach komputerowych
- GPT to część standardu **UEFI (Unified Extensible Firmware Interface)**, który zastąpił BIOS w komputerach PC (interfejs graficzny, obsługa myszki)
- opracowanie: IBM/Microsoft, 2010 rok
- maksymalny rozmiar dysku to **9,4 ZB** ( $2^{64}$  sektorów × 512 bajtów)
- możliwość utworzenia do 128 partycji podstawowych

## Struktura dysku twardego - GPT (struktura)

- Protective MBR** - pozostawiony dla bezpieczeństwa
- GPT Header (512 bajtów):**
  - liczba pozycji w tablicy partycji
  - rozmiar pozycji w tablicy partycji
  - położenie zapasowej kopii GPT
  - unikatowy identyfikator dysku
  - sumy kontrolne
- Entry x (128 bajtów):**
  - typ partycji
  - unikatowy identyfikator
  - początkowy i końcowy numer LBA
  - atrybuty
  - nazwa



## System plików FAT (File Allocation Table)

- opracowany na przełomie lat 70. i 80. dla systemu MS-DOS
- występuje w czterech wersjach: FAT12, FAT16, FAT32 i exFAT (FAT64)
- numer występujący po słowie FAT oznacza liczbę bitów przeznaczonych do kodowania (numeracji) **jednostek alokacji pliku** (JAP), tzw. **klastrów** (ang. cluster) w tablicy alokacji plików
  - 12 bitów w systemie FAT12
  - 16 bitów w systemie FAT16
  - 32 bity w systemie FAT32 (praktycznie 28)
  - 64 bity w systemie exFAT (FAT64)
- ogólna struktura dysku logicznego / dyskietki w systemie FAT:

Rekord ładujący + sektory zarezerwowane	Tablica rozmieszczenia plików - FAT	Kopia FAT	Katalog główny (FAT12 i FAT16)	Miejsce na pliki i podkatalogi
---	-------------------------------------	-----------	--------------------------------	--------------------------------

## FAT12

- tablica rozmieszczenia plików FAT** tworzy swego rodzaju „mapę” plików zapisanych na dysku
- za tablicą FAT znajduje się jej kopia, która nie jest wykorzystywana

Rekord ładujący + sektory zarezerwowane	Tablica rozmieszczenia plików - FAT	Kopia FAT	Katalog główny (FAT12 i FAT16)	Miejsce na pliki i podkatalogi
---	-------------------------------------	-----------	--------------------------------	--------------------------------

- za kopią tablicy FAT znajduje się **katalog główny** zajmujący określoną dla danego typu dysku liczbę sektorów

Rekord ładujący + sektory zarezerwowane	Tablica rozmieszczenia plików - FAT	Kopia FAT	Katalog główny (FAT12 i FAT16)	Miejsce na pliki i podkatalogi
---	-------------------------------------	-----------	--------------------------------	--------------------------------

## FAT12

- system plików FAT12 przeznaczony jest dla nośników o małej pojemności
- obsługuje  $2^{12} = 4096$  jednostek alokacji, max. rozmiar partycji to 16 MB
- rekord ładujący** zajmuje pierwszy sektor dyskietki lub dysku logicznego

Rekord ładujący + sektory zarezerwowane	Tablica rozmieszczenia plików - FAT	Kopia FAT	Katalog główny (FAT12 i FAT16)	Miejsce na pliki i podkatalogi
---	-------------------------------------	-----------	--------------------------------	--------------------------------

- rekord ładujący zawiera następujące dane:
  - instrukcja skoku do początku programu ładującego (3 bajty)
  - nazwa wersji systemu operacyjnego (8 bajtów)
  - struktura BPB (ang. BIOS Parametr Block) - blok parametrów BIOS (25 bajtów)
  - rozszerzony BPB (ang. Extended BPB, 26 bajtów)
  - wykonywalny kod startowy uruchamiający system operacyjny (448 bajtów)
  - znacznik końca sektora - 55AAH (2 bajty)

## FAT12

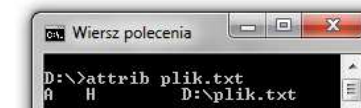
- katalog główny zawiera 32-bajtowe pola mogące opisywać pliki, podkatalogi lub etykiety dysku

Zawartość pola:

Atrybuty pliku:

Bajty	Rozmiar	Zawartość
00H-07H	8	Nazwa pliku w kodach ASCII
08H-0AH	3	Rozszerzenie nazwy pliku
0BH	1	Atrybuty pliku
0CH-15H	10	Zarezerwowane
16H-17H	2	Czas utworzenia lub aktualizacji pliku
18H-19H	2	Data utworzenia lub aktualizacji pliku
1AH-1BH	2	Numer pierwszej JAP
1CH-1DH	2	Mniej znaczące słowo rozmiaru pliku
1EH-1FH	2	Bardziej znaczące słowo rozmiaru pliku

Bit	Znaczenie
0	Plik tylko do odczytu (read only)
1	Plik ukryty (hidden)
2	Plik systemowy (system)
3	Etykieta dysku (volume label)
4	Podkatalog
5	Plik archiwalny (archive)
6,7	Nie wykorzystywane





## FAT32


- po raz pierwszy wprowadzony w systemie Windows 95 OSR2
- ogólna struktura systemu FAT32 jest taka sama jak w FAT12/FAT16 - nie ma tylko miejsca przeznaczonego na katalog główny
- w systemie FAT32 katalog główny może znajdować się w dowolnym miejscu na dysku i może zawierać maksymalnie 65 532 pliki i katalogi

Rekord ładujący + sektory zarezerwowane	Tablica rozmieszczenia plików - FAT	Kopia FAT	Miejsce na pliki i katalogi
---	-------------------------------------	-----------	-----------------------------

- do adresowania JAP stosuje się, obcięty o 4 najstarsze bity, adres 32-bitowy i dlatego dysk z FAT32 może zawierać maksymalnie  $2^{28}$  JAP
- w systemie FAT32 można formatować tylko dyski, nie można natomiast zainstalować go na dyskietkach

## FAT32 - długie nazwy plików

- Nazwa pliku: **Systemy Operacyjne - praca domowa.txt**

długa nazwa pliku 

0000	43 20 00 64 00 6F 00 6D-00 6F 00 0F 00 CF 77 00	C .d.o.m.o....w.
0010	61 00 2E 00 74 00 78 00-74 00 00 00 00 FF FF	a...t.x.t.....
0020	02 63 00 79 00 6A 00 6E-00 65 00 0F 00 CF 20 00	.c.y.j.n.e....
0030	2D 00 20 00 70 00 72 00-61 00 00 00 63 00 61 00	-.p.r.a...c.a.
0040	01 53 00 79 00 73 00 74-00 65 00 0F 00 CF 6D 00	.S.y.s.t.e....m.
0050	79 00 20 00 4F 00 70 00-65 00 00 00 72 00 61 00	y. .O.p.e...r.a.
0060	53 59 53 54 45 4D 7E 31-54 58 54 20 00 4B 03 80	SYSTEM~1TXT .K..
0070	67 32 67 32 00 00 08 80-67 32 02 00 06 00 00 00	g2g2....g2.....

skrótowa nazwa pliku 

## FAT32 - długie nazwy plików

- wprowadzone w systemie Windows 95
- informacje o nazwie pliku zapamiętywane są jako:
  - długa nazwa
  - skrótowa nazwa (tzw. alias długiej nazwy)
- długie nazwy plików** zapisywane są także w 32-bajtowych strukturach, przy czym jedna nazwa zajmuje kilka struktur (w jednej strukturze umieszczonych jest 13 kolejnych znaków w formacie Unicode)
- skrótowa nazwa pliku** przechowywana jest w identycznej, 32-bajtowej, strukturze jak w przypadku plików w starym formacie 8+3
  - rozszerzenie długiej nazwy staje się rozszerzeniem skróconej nazwy
  - pierwsze 6 znaków długiej nazwy staje się pierwszymi sześcioma znakami skróconej nazwy (niezgodzone znaki zamieniane są na znak podkreślenia, małe litery zamieniane są na wielkie litery)
  - pozostałe dwa znaki nazwy skróconej to ~1 lub jeśli plik o takiej nazwie istnieje ~2, itd.

## FAT - wady systemu plików FAT

- fragmentacja wewnętrzna** - nawet najmniejszy plik zajmuje całą JAP - gdy rozmiar klastra jest duży, a na dysku znajduje się dużo małych plików - pewna część miejsca jest tracona
- fragmentacja zewnętrzna** - silna fragmentacja plików pomiędzy wiele klastrów o bardzo różnym fizycznym położeniu na dysku (konieczność okresowej defragmentacji przy użyciu specjalnych narzędzi programowych)
- duże prawdopodobieństwo powstawania błędów zapisu, polegających na przypisaniu jednego klastra dwóm plikom (tzw. **crosslinks**), co kończy się utratą danych z jednego lub obu „skrzyżowanych” plików
- typowym błędem, pojawiającym się w systemie FAT, jest również pozostawianie tzw. **zagubionych klastrów (lost chains)**, tj. jednostek alokacji nie zawierających informacji, ale opisanych jako zajęte
- brak mechanizmów ochrony - praw dostępu

## exFAT (FAT64)

- po raz pierwszy pojawił się w listopadzie 2006 roku w Windows Embedded CE 6.0 i Windows Vista SP1
- obsługiwany także przez Windows 7/8/10, Windows Server 2003/2008, Windows XP SP2/SP3, Linux
- stworzony przez Microsoft na potrzeby pamięci Flash
- podstawowe cechy:
  - maksymalna wielkość pliku to  $2^{64} = 16$  EB
  - maksymalna wielkość klastra - do 32 MB
  - nieograniczona liczba plików w pojedynczym katalogu
  - prawa dostępu do plików i katalogów

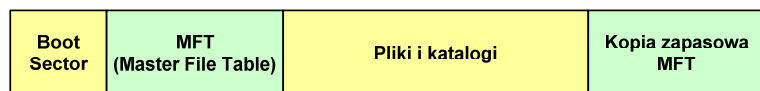
## NTFS (New Technology File System)

- wersja 1.0 (połowa 1993 r.) - Windows NT 3.1
- wersja 3.1 (NTFS 5.1) - Windows XP/Server 2003/Vista/7/8/10
- struktura wolumenu (dysku) NTFS:



- **Boot Sector** rozpoczyna się od zerowego sektora partycji, może zajmować 16 kolejnych sektorów, zawiera podobne dane jak w systemie FAT

## NTFS



- **MFT (Master File Table)** - specjalny plik, niewidoczny dla użytkownika, zawiera wszystkie dane niezbędne do odczytania pliku z dysku, składa się z rekordów o stałej długości (1 kB - 4 kB)
- pierwsze 16 (NTFS 4) lub 26 (NTFS 5) rekordów jest zarezerwowane dla tzw. metaplików, np.
  - rekord nr: 0 plik: **\$Mft** (główna tablica plików)
  - rekord nr: 1 plik: **\$MftMirr** (główna tablica plików 2)
  - rekord nr: 5 plik: **\$** (indeks katalogu głównego)
- pozostała część pliku MFT przeznaczona jest na rekordy wszystkich plików i katalogów umieszczonych na dysku

## NTFS

- struktura wolumenu (dysku) NTFS:



- plik w NTFS to **zbiór atrybutów**
- wszystkie atrybuty mają dwie części składowe: **nagłówek** i **blok danych**
- **nagłówek** opisuje atrybut, np. liczbę bajtów zajmowanych przez atrybut, rozmiar bloku danych, położenie bloku danych, znacznik czasu
- **bloku danych** zawiera informacje zgodne z przeznaczeniem atrybutu



## NTFS - Pliki

- pliki w systemie NTFS są reprezentowane w MFT przez rekord zawierający atrybuty:

- `$Standard_Information`
- `$File_Name`
- `$Security_Descriptor`
- `$Data`

<code>\$Standard_Information</code> (Informacje standardowe)	<code>\$File_Name</code> (Nazwa pliku)	<code>\$Security_Descriptor</code> (Opis praw dostępu)	<code>\$Data</code> (Dane)
---	---	---	-------------------------------

- w przypadku małych plików wszystkie jego atrybuty zapisywane są bezpośrednio w MFT (atrybuty **rezydentne**)

## NTFS - Katalogi

- katalogi reprezentowane są przez rekordy zawierające trzy takie same atrybuty jak pliki:

- `$Standard_Information`
- `$File_Name`
- `$Security_Descriptor`

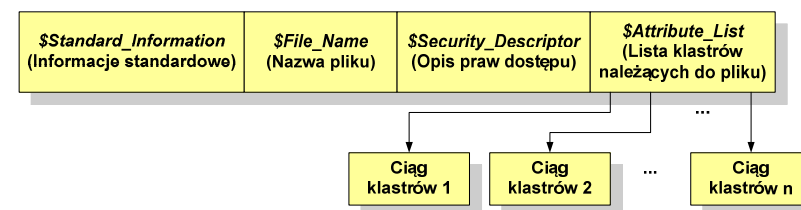
<code>\$Standard_Information</code> (Informacje standardowe)	<code>\$File_Name</code> (Nazwa pliku)	<code>\$Security_Descriptor</code> (Opis praw dostępu)	<code>\$Index_Root</code>	<code>\$Index_Allocation</code>	<code>\$Bitmap</code>
---	---	---	---------------------------	---------------------------------	-----------------------

- zamiast atrybutu `$Data` umieszczone są trzy atrybuty przeznaczone do tworzenia list, sortowania oraz lokalizowania plików i podkatalogów

- `$Index_Root`
- `$Index_Allocation`
- `$Bitmap`

## NTFS - Pliki

- jeśli atrybuty pliku są duże (najczęściej dotyczy to atrybutu `$Data`), to w rekordzie w MFT umieszczany jest tylko nagłówek atrybutu oraz wskaźnik do jego bloku danych, a sam blok danych przenoszony jest na dysk poza MFT (atrybuty **nierezydentne**)
- blok danych atrybutu nierezydentnego zapisywany jest w przyległych klastrach
- jeśli nie jest to możliwe, to dane zapisywane są w kilku ciągach jednostek alokacji i wtedy każdemu ciągowi odpowiada wskaźnik w rekordzie MFT



## Koniec wykładu nr 8

Dziękuję za uwagę!