

# Systemy Operacyjne (1)

Marcin Gogolewski  
marcing@wmi.amu.edu.pl

Uniwersytet im. Adama Mickiewicza w Poznaniu

Poznań, 21 października 2018

# Zasady zaliczenia i egzamin

Dyżury Wtorki: 12:00-13:00, piątki: 11:30-12:30 (?)

Kontakt e-mail: [marcing@wmi.amu.edu.pl](mailto:marcing@wmi.amu.edu.pl),  
tel.: (061)8295400

Pokój B0-15, Collegium Mathematicum,  
ul. Umultowska 87

Zaliczenie quizy (40pkt z najlepszych 5), projekt (10pkt)  
każdy element należy oddzielnie zaliczyć na min. 50%

Egzamin test z całości materiału z wykładu

Strona WWW [https://git.wmi.amu.edu.pl/marcing/  
systemy-operacyjne-niestacjonarne-2018/  
wiki](https://git.wmi.amu.edu.pl/marcing/systemy-operacyjne-niestacjonarne-2018/wiki)

# Bibliografia

- *A. Silberschatz, P.B. Galvin, G. Gagne*, Podstawy systemów operacyjnych, Wydanie VII, WNT, 2006
- *A. S. Tanenbaum, H. Bos*, **Systemy operacyjne**, Wydanie IV, Helion 2015
- *M. Bach*, Budowa systemu operacyjnego UNIX, WNT 1995
- *D. Comer*, Sieci komputerowe TCP/IP, tom 1, WNT 1999
- *W. Stallings* Organizacja i architektura systemu komputerowego, WNT, 2004.
- *W. Stallings*: Systemy operacyjne. Struktura i zasady budowy, PWN, 2006
- *D. P. Bovet, M. Cesati*, Linux Kernel, od portów wejścia/wyjścia do zarządzania procesami, O'Reilly, 2001

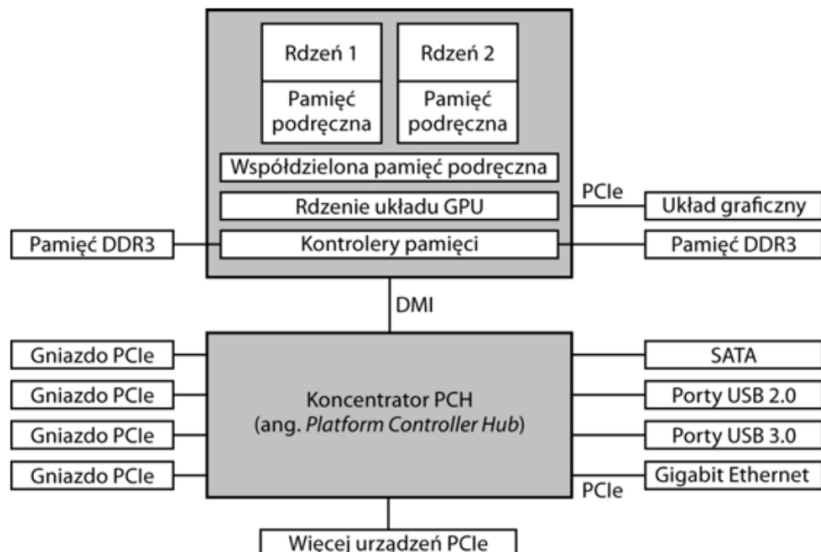
## Bibliografia (2)

- *U. Vahalia*: Jądro systemu Unix, Nowe horyzonty, WNT, 2001.
- *B. Goodheart, J. Cox*, Sekrety magicznego ogrodu, UNIX System V Wersja 4 od środka, WNT 2001
- *K. Haviland, D. Gray, B. Salma*, UNIX programowanie systemowe, Wyd RM 1999
- *J. S. Gray*, Komunikacja między procesami w Unixie, Wyd. RM, 1998
- *M.K. Johnson, E. W. Tron*, Oprogramowanie użytkowe w systemie Linux, WNT 2000

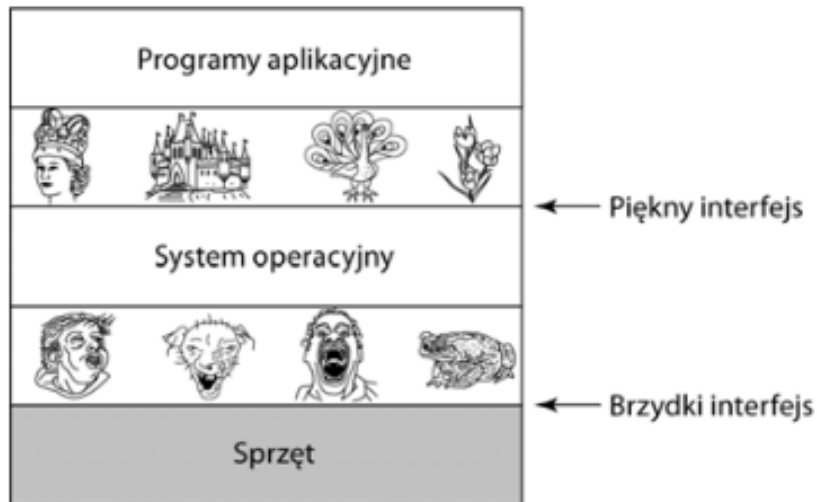
# Z czego składa się komputer?

- procesor(y)
- pamięć
- zegar
- klawiatura/mysz
- monitor(y)
- karty sieciowe
- dyski
- drukarki
- ...

# Komputer



# Co to jest system operacyjny?



# Po co nam system operacyjny

## Funkcje systemu operacyjnego

- udostępnianie zasobów (w wygodnej formie) programom użytkowym
- zarządzanie zasobami



## Per Brinch Hansen – definicja SO (1973?)

System operacyjny jest zbiorem ręcznych i automatycznych procedur, które pozwalają grupie osób na efektywne współdzielenie urządzeń maszyny cyfrowej.

## Alan Shaw – definicja SO (2002?)

System operacyjny (nadzorczy, nadrzędny, sterujący) jest to zorganizowany zespół programów, które pośredniczą między sprzętem a użytkownikami, dostarczając użytkownikom zestawu środków ułatwiających projektowanie, kodowanie, uruchamianie i eksploatację programów oraz w tym samym czasie sterują przydziałem zasobów dla zapewnienia efektywnego działania.

## Abraham Silberschatz – definicja SO (2010?)

System operacyjny jest programem, który działa jako pośrednik między użytkownikiem komputera a sprzętem komputerowym. Zadaniem systemu operacyjnego jest tworzenie środowiska, w którym użytkownik może wykonywać programy w sposób wygodny i wydajny

# Andrew Tanenbaum – definicja SO

System operacyjny jest warstwą oprogramowania operującą bezpośrednio na sprzęcie, której celem jest zarządzanie zasobami systemu komputerowego i stworzenie użytkownikowi środowiska łatwiejszego do zrozumienia i wykorzystania.

# Krótką historia komputerów

## Pierwsze znane pomysły komputerów cyfrowych

Charles Babbage (1792-1871) budował „silnik analityczny”. Nie udało się, ale prawdopodobnie rozumiał ideę oprogramowania (zatrudnił nawet Adę Lovelace jako programistkę; notabene: nazwa języka Ada z lat 70-tych jest właśnie na jej cześć).

Komputer ten nie miał oczywiście systemu operacyjnego (skoro nigdy nie zaczął poprawnie działać, to nie pojawiła się taka idea).

## Krótką historia komputerów (2)

### Pierwsza generacja 1945-1955

Lampy elektronowe (niektóre ponad 20tyś lamp), programowalne z początku przez *plugboard* potem karty perforowane.

### Druga generacja 1955-1965

Tranzystory, systemy wsadowe, programowalne w *Fortranie*, nazywane często *mainframe* np. IBM7094 (wraz z systemami pomocniczymi IBM1401). Karty zawierały dyrektywy (typu \$JOB, \$FORTRAN, \$LOAD, \$RUN, \$END), dane lub kod programu.

## Krótką historia komputerów (3)

### Trzecia generacja 1965-1980

Układy scalone, ale systemy nadal dość duże i drogie. Przykładowy system to System/360, pojawiła się wieloprogramowość, wykorzystano *spooling* (ang. *Simultaneous Peripheral Operation On Line*), czyli przechowywanie wielu zadań w pamięci (kolejno przydzielano im procesor). Pojawił się nawet pomysł *CTSS* (ang. *Compatible Time-Sharing System*) i powstał system *MULTICS*. O ile komercyjnie nie był sukcesem, to stanowi podwalinę większości współczesnych systemów operacyjnych.

## Krótką historia komputerów (4)

### Narodziny Uniksa

Ken Thompson z Bell Labs (wcześniej pracujący nad MULTICSem) dostał do dyspozycji komputer PDP-7 i... napisał wersję „jednoużytkownikową” MULTICSa, którą nazwał *Unix*. Do prac przyłączył się Dennis Ritchie, poza rozwojem systemu operacyjnego. Stworzono język programowania B (nie miał on jednak np. struktur). W pierwszej połowie lat 70tych (1973) powstał język C wraz z kompilatorem i napisanym w tym języku jądrem systemu Unix.



## Krótką historia komputerów (5)

### Czwarta generacja od 1980

Komputery osobiste. W 1974 powstał komputery oparte na procesorze Intel 8080. System do nich nazywał się *CP/M (Control Program for Microcomputers)*. Prawa do systemu miała firma *Digital Research*.

## Krótką historia komputerów (6)

### MS-DOS

W latach 80tych firma IBM stworzyła IBM-PC i szukała oprogramowania. Skontaktowali się m.in. z Billem Gatesem (autorem kodu kompilatora BASIC), który poradził IBM, by skontaktował się z Digital Research. Na skutek błędnej decyzji transakcja nie doszła do skutku, a Bill Gates zakupił prawa do systemu DOS (Z Seattle Computer Products), dostosował go (zatrudnił jednego z programistów do nowopowstałej firmy Microsoft) i zaczął sprzedawać MS-DOS producentom sprzętu komputerowego (a właściciel Digital Research zmarł w tym czasie w niewyjaśnionych okolicznościach).

## Krótką historia komputerów (5)

### Piąta generacja od 1990

Pierwszy telefon komórkowy (1946) ważył ok. 40kg, od 1990 powstaje coraz więcej urządzeń przenośnych (nieco lżejszych).

## Inne (znane) systemy operacyjne

OS/2 IBM i Microsoft, od 1987

Windows NT pierwszy nie bazujący na DOS system Microsoftu (bazujący na OS/2) od 1993

Linux premiera jądra systemu 17 września 1991 (od początku 32 bitowy)

GNU/Linux system operacyjny oparty o jądro Linuksa i narzędzia GNU (praktycznie każda dystrybucja, najstarsza z obecnie istniejących to *Slackware Linux* (połowa 1993))

Android Gnu/Linux + Android (nie zawiera kodu GNU) = masa obsługiwanych architektur

# Uruchamianie komputera

- CPU wykonuje program (BIOS) spod ustalonego adresu
- BIOS wykrywa i inicjalizuje sprzęt
- BIOS poszukuje nośnika rozruchowego SO
- po załadowaniu SO może odczytać konfigurację z BIOSu

# Powłoka systemu

## Przykłady

CMD systemy Windows, OS/2

Bash systemy „uniksowe”

...

# Plik wsadowy

Lista poleceń powłoki zapisana w pliku (w bardziej zaawansowanej wersji możliwe instrukcje warunkowe, pętle, funkcje – skrypt).

## Uwaga

W skrypcie można co prawda definiować zmienne lub funkcje, jednak (inaczej niż np. w językach obiektowych) kolejność definicji **ma znaczenie**.

# Podstawowe polecenia CMD

Wszystkich poleceń jest około 300.  
Niektóre funkcjonalności się powtarzają!

**Wielkość liter nie ma znaczenia, w odróżnieniu od systemów uniksowych.**



# Podstawowe polecenia CMD (1)

`attrib` ustawia atrybuty pliku (np. RO)

`chdir/cd` zmiana bieżącego katalogu

`certutil` m.in. funkcje haszujące (nie dostępne w DOS)

`choice` do wykorzystania w skryptach (np. wybór opcji)

`cls` *clear screen* – czyszczenie ekranu

`comp` porównywanie plików (nie w DOS)

`copy` kopiowanie pliku

`date` bieżąca data (lub ustawienie daty)

`del` usuwanie pliku (także *deltree* dla struktury podkatalogów)

`dir` zawartość katalogu (ciekawe opcje)

# Podstawowe polecenia CMD (1)

doskey program do tworzenia makr (jak *alias* w Bash)

echo wypisywanie tekstu/zmiennych

edit/edlin edytory tekstu (historyczne)

exit zakończenie sesji powłoki

fdisk program do podziału dysku na partycje

for tworzenie „pętli” po listach plików

ftp klient ftp (nie zawsze dostępny)

goto w skryptach skok do etykiety (np. *tutaj:*)

hostname nazwa komputera

if instrukcja warunkowa

## Podstawowe polecenia CMD (2)

`ipconfig` konfiguracja sieci (nie w DOS)

`label` do wyświetlenia/zmiany etykiety dysku

`mkdir/md` tworzenie katalogu

`more` wyświetlanie tekstu po stronie

`msd` Microsoft Diagnostics

`msg` program do wysyłania komunikatów

`net` program do konfiguracji sieci (np. montowanie udziału Windows)

`nslookup` zapytanie do serwera DNS

`ping`

`rmdir/rd` usuwanie katalogu

## Podstawowe polecenia CMD (3)

recover odzyskiwanie (?) danych z dysku

reg manipulacja rejestrem

rem do komentarzy w skryptach

n/rename zmiana nazwy pliku

replace podmiana pliku

reset sesji/terminala

runas uruchamianie programu jako inny użytkownik

set ustawianie zmiennych

shift przesuwanie listy zmiennych (parametrów skryptu)

shutdown

## Podstawowe polecenia CMD (4)

`verify` test poprawności zapisu danych

`vol` tzw. volume label

`waitfor` oczekiwanie na zdarzenie

i inne, np. `sort`, `systeminfo`, `taskkil`, `tasklist`, `telnet`, `time`, `tree`, `type`.