

Podstawowe pojęcia dotyczące
komputera i podzespoły komputera.
Urządzenia peryferyjne i media.

Pojęcie i budowa systemu komputerowego.

System komputerowy

(ang. computer system)

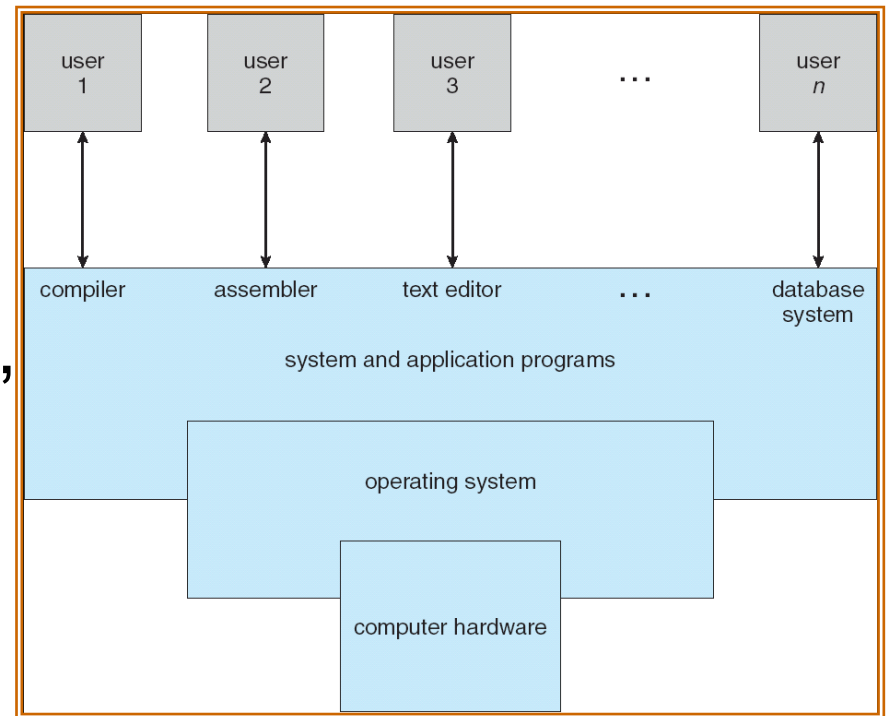
– układ współdziałania sprzętu komputerowego oraz oprogramowania, działających również w ramach sieci komputerowej.

Jednostka centralna (inaczej jednostka systemowa lub komputer) – zasadnicza część zestawu komputerowego zawierająca najważniejsze elementy składowe zestawu komputera, zawarte we wspólnej obudowie.

Warstwy systemu komputerowego

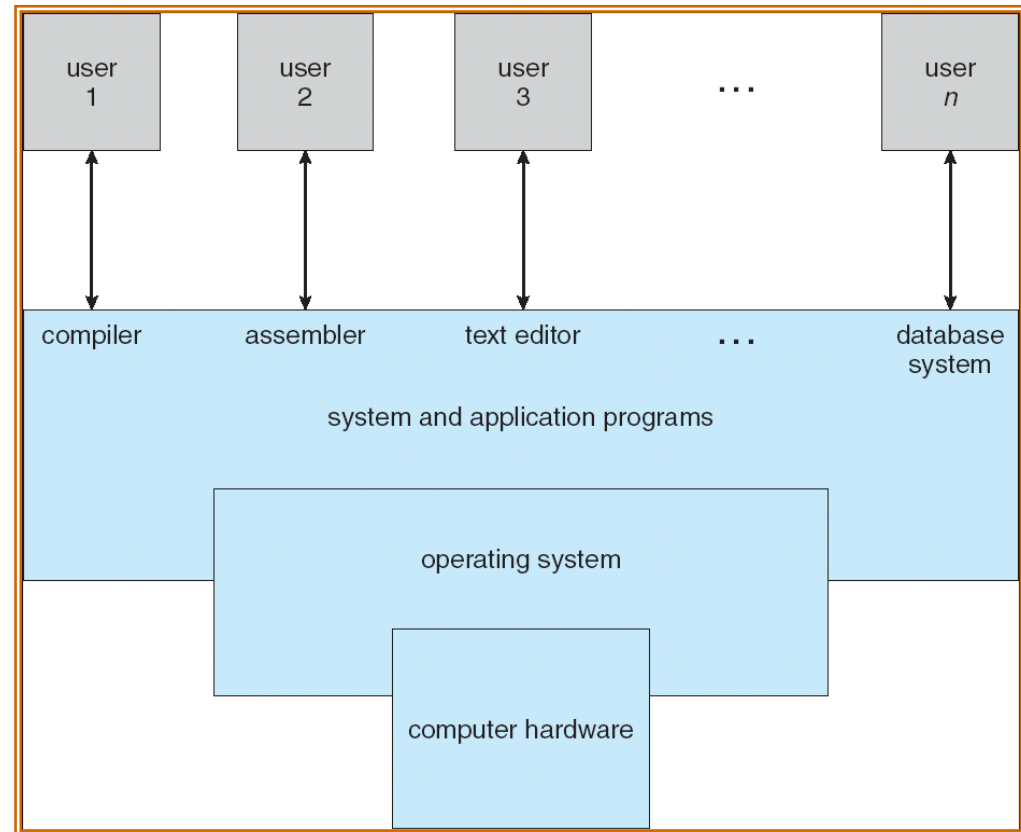
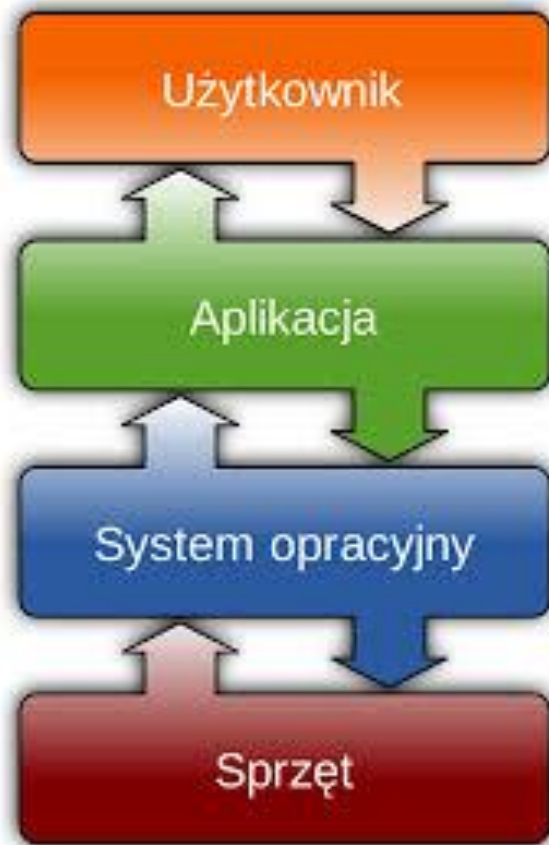
Struktura systemu komputerowego składa się z pięciu warstw:

- sprzętowa,
- system operacyjny,
- programy narzędziowe,
- programy użytkowe,
- użytkownicy.



System komputerowy działa bez udziału człowieka.

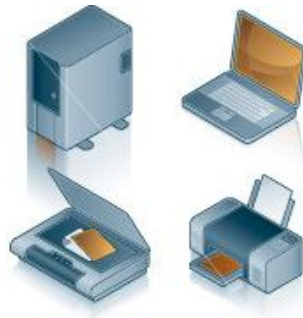
Warstwy systemu komputerowego



Aplikacje = programy narzędziowe i programy użytkowe

Zadania warstw systemu komputerowego

- Sprzęt (hardware) – zapewnia podstawowe możliwości obliczeniowe (procesor, pamięć, urządzenia wejścia/wyjścia) – podstawowe zasoby systemu komputerowego.



- Urządzenie we/wy, (I/O) – urządzenie do komunikacji systemu komputerowego z jego użytkownikiem lub innym systemem przetwarzania danych.

Zadania warstw systemu komputerowego

- urządzenia wejścia to: klawiatura, mysz, skaner, mikrofon, kamera internetowa, ekran dotykowy, czytnik linii papilarnych, odbiornik GPS, urządzenia IoT (np. inteligentne czujniki, smartfony). >



- urządzenia wyjścia to: monitor, drukarka, głośniki, słuchawki, brzęczyk, ploter, projektor . <



- urządzenia wejścia i wyjścia to: karta sieciowa, **modem**, ekran dotykowy, moduł Bluetooth, **moduł IrDA**, urządzenia USB-C, moduły Bluetooth 5.x, Wi-Fi, Thunderbolt, nowoczesne kamery, mikrofony konferencyjne, urządzenia IoT urządzenie USB oraz wszelkie inne nośniki danych z możliwością zapisu i odczytu. =



Technologie jak IrDA czy modem są przestarzałe.

Zadania warstw systemu komputerowego

- Wszystkie urządzenia wejścia-wyjścia są wyposażone w interfejs, który pozwala na komunikację z systemem.



- Urządzenia peryferyjne mogą być podłączone przewodowo lub bezprzewodowo (np. USB-C, Thunderbolt, HDMI, DisplayPort, Bluetooth, Wi-Fi).



Zadania warstw systemu komputerowego

Oprogramowanie systemowe (System software) – kontroluje i koordynuje działanie zasobów sprzętowych przez różne programy użytkowe dla różnych użytkowników.

Przykłady współczesnego oprogramowania systemowego:

- Windows 11 / Windows Server 2022 – popularne w środowiskach biurowych i serwerowych.
- Linux (Ubuntu, Debian, Fedora, Red Hat Enterprise Linux) – szeroko stosowane w serwerach, chmurze i systemach wbudowanych.
- macOS – system dla komputerów Apple.
- Android / iOS – systemy operacyjne dla urządzeń mobilnych.



Zadania warstw systemu komputerowego

Sterowniki urządzeń (Device Drivers) to kluczowy element oprogramowania systemowego – działają jako pośrednik między systemem operacyjnym a sprzętem, umożliwiając prawidłową komunikację i wykorzystanie funkcji urządzeń.

Przykłady sterowników urządzeń:

1. Sterowniki GPU (karty graficzne)

NVIDIA GeForce Driver – obsługa kart graficznych NVIDIA, zapewnia akcelerację grafiki 3D, CUDA.

AMD Radeon Software – sterowniki dla kart AMD, obsługa DirectX, Vulkan.

Intel Graphics Driver – dla zintegrowanych układów graficznych Intel.

2. Sterowniki drukarek

HP Universal Print Driver – obsługa wielu modeli drukarek HP.

Canon UFR II / PCL – sterowniki dla drukarek Canon.

Epson ESC/P-R – dla drukarek Epson.

Zadania warstw systemu komputerowego

Sterowniki urządzeń (Device Drivers) to kluczowy element oprogramowania systemowego – działają jako pośrednik między systemem operacyjnym a sprzętem, umożliwiając prawidłową komunikację i wykorzystanie funkcji urządzeń.

Przykłady sterowników urządzeń:

3. Sterowniki kart sieciowych

Intel PROSet/Wireless Driver – dla kart Wi-Fi Intel.

Realtek Ethernet Controller Driver – dla kart przewodowych Realtek.

Broadcom NetXtreme – dla kart sieciowych w serwerach.

4. Sterowniki kontrolerów RAID

Intel Rapid Storage Technology (RST) – obsługa macierzy RAID na platformach Intel.

MegaRAID (Broadcom/LSI) – dla kontrolerów RAID w serwerach.

Adaptec RAID Controller Driver – dla kontrolerów Adaptec.

Zadania warstw systemu komputerowego

Programy narzędziowe wbudowane (Utilities) – udostępnia dogodne interfejsy użytkowe wspomagające zarządzanie zasobami sprzętowymi oraz usprawniające działanie systemu.

Klasyczne przykłady:

PowerShell – automatyzacja i zarządzanie systemem Windows.

bash / zsh – powłoki systemowe w Linux/Unix.

Disk Management, fsck, chkdsk – narzędzia do zarządzania dyskami i systemami plików.

Nowoczesne przykłady wbudowanych programów narzędziowych (Utilities), które wspomagają zarządzanie systemem i zasobami, to:

1. Windows Terminal

Umożliwia pracę z wieloma powłokami (PowerShell, CMD, WSL) w jednym oknie.

Obsługuje zakładki, motywy, integrację z Git Bash.

2. Windows Subsystem for Linux (WSL)

Pozwala uruchamiać dystrybucje Linux bezpośrednio w Windows.

Idealne dla administratorów i programistów DevOps.



Zadania warstw systemu komputerowego

Programy narzędziowe wbudowane (Utilities) – udostępnia dogodne interfejsy użytkowe wspomagające zarządzanie zasobami sprzętowymi oraz usprawniające działanie systemu.

Nowoczesne przykłady wbudowanych programów narzędziowych (Utilities), które wspomagają zarządzanie systemem i zasobami, to:

3. Podsystemy i narzędzia kontenerowe

Docker CLI – zarządzanie kontenerami i obrazami.

Podman – alternatywa dla Dockera w środowiskach Linux.

4. Narzędzia do zarządzania bezpieczeństwem

Windows Defender Security Center** – zaawansowane opcje ochrony.

Linux `auditd` – monitorowanie zdarzeń bezpieczeństwa.

5. Narzędzia do zarządzania pamięcią masową

Storage Spaces (Windows) – tworzenie macierzy dyskowych.

ZFS utilities (Linux) – zarządzanie systemem plików ZFS.



Zadania warstw systemu komputerowego

Programy narzędziowe wbudowane (Utilities) – udostępnia dogodne interfejsy użytkowe wspomagające zarządzanie zasobami sprzętowymi oraz usprawniające działanie systemu.

Nowoczesne przykłady wbudowanych programów narzędziowych (Utilities), które wspomagają zarządzanie systemem i zasobami, to:

6. Narzędzia do monitorowania i diagnostyki

Windows Performance Monitor – analiza wydajności systemu.

htop (Linux) – interaktywne monitorowanie procesów.

nmon – zaawansowane statystyki systemowe.

7. Narzędzia do automatyzacji i zarządzania konfiguracją

Ansible CLI – automatyzacja konfiguracji w Linux.

Chocolatey (Windows) – zarządzanie pakietami.



Zadania warstw systemu komputerowego

Oprogramowanie narzędziowe (Software tools) – udostępnia dogodne interfejsy użytkowe wspomagające zarządzanie zasobami sprzętowymi oraz usprawniające działanie systemu.

Klasyczne przykłady:

Edytory tekstu – np. Notepad, Vim.

Kompilatory – GCC, Turbo Pascal.

Debuggery – GDB.

Systemy kontroli wersji – CVS, Subversion.

Narzędzia do archiwizacji – WinRAR, tar.

Nowoczesne przykłady:

Wirtualizacja – VMware, Hyper-V.

Konteneryzacja – Docker, Kubernetes.

Narzędzia DevOps – Jenkins, Ansible, Terraform.

Systemy CI/CD – GitLab CI, GitHub Actions.

Monitorowanie i zarządzanie – Prometheus, Grafana.



Zadania warstw systemu komputerowego

Oprogramowanie użytkowe (Utility software) – określa sposoby użycia zasobów systemowych do rozwiązywania problemów obliczeniowych zadanych przez użytkownika.

Klasyczne przykłady:

- Edytory tekstu – WordPerfect, Microsoft Word (starsze wersje).
- Arkusze kalkulacyjne – Lotus 1-2-3, Excel.
- Programy graficzne – CorelDRAW, Paint.
- Systemy baz danych – dBase, MS Access.
- Programy do tworzenia prezentacji – Harvard Graphics.

Nowoczesne przykłady:

- Chmura obliczeniowa – Microsoft Azure, AWS, Google Cloud.
- Systemy baz danych – MySQL, PostgreSQL, MongoDB.
- Aplikacje biurowe online – Microsoft 365, Google Workspace.
- Narzędzia do analizy danych – Power BI, Tableau.
- Aplikacje komunikacyjne – Teams, Slack.



Oprogramowanie do wirtualizacji

Hyper-V, VMware ESXi, VirtualBox, KVM

– umożliwiają uruchamianie wielu systemów operacyjnych na jednym sprzęcie.

Przykłady i ich charakterystyka:

Hyper-V (Microsoft)

- Typ: Hypervisor typu 1 (bezpośrednio na sprzęcie) w Windows Server, ale w Windows 10/11 działa jako typ 2.
- Zastosowanie: środowiska testowe, serwery wirtualne, integracja z Azure.

VMware ESXi

- Typ: Hypervisor typu 1.
- Zastosowanie: profesjonalne środowiska serwerowe, wirtualizacja w centrach danych.

VirtualBox (Oracle)

- Typ: Hypervisor typu 2 (instalowany na systemie operacyjnym).
- Zastosowanie: nauka, testy, środowiska developerskie.

KVM (Kernel-based Virtual Machine)

- Typ: Hypervisor typu 1 w jądrze Linux.
- Zastosowanie: serwery Linux, chmura, integracja z OpenStack.

Zarządzania bezpieczeństwem

1. Kontrola dostępu w systemach Linux

- SELinux (Security-Enhanced Linux)

Mechanizm kontroli dostępu typu MAC (Mandatory Access Control), pozwala definiować polityki bezpieczeństwa dla procesów i plików.

- AppArmor

Alternatywa dla SELinux, prostsza konfiguracja, profilowanie aplikacji.

- Szyfrowanie

Narzędzia takie jak LUKS, cryptsetup – szyfrowanie dysków i partycji.

2. Ochrona przed zagrożeniami i kontrola ruchu sieciowego

- Windows Defender

Wbudowany antywirus i system ochrony w Windows.

- FirewallD (Linux)

Dynamiczna konfiguracja zapory sieciowej, zarządzanie strefami bezpieczeństwa.

- iptables (Linux)

Klasyczne narzędzie do filtrowania pakietów i konfiguracji zapory.

Systemy plików

Zarządzanie strukturą danych na dyskach. Przykłady systemów plików i ich charakterystyka:

1. NTFS (New Technology File System)

Platforma: Windows

Cechy: obsługa dużych plików, uprawnienia ACL, szyfrowanie (EFS), kompresja, dziennikowanie.

2. ext4 (Fourth Extended File System)

Platforma: Linux

Cechy: obsługa dużych woluminów, journaling, szybkie odzyskiwanie po awarii, kompatybilność z ext2/ext3.

3. APFS (Apple File System)

Platforma: macOS, iOS

Cechy: szyfrowanie natywne, snapshoty, optymalizacja dla SSD, dynamiczne alokowanie przestrzeni.

4. ZFS (Zettabyte File System)

Platforma: Unix/Linux (np. FreeBSD, Ubuntu)

Cechy: integracja z RAID, snapshoty, deduplikacja, wysoka odporność na błędy, skalowalność.

Zadania warstw systemu komputerowego

Kategorie użytkowników systemów informatycznych

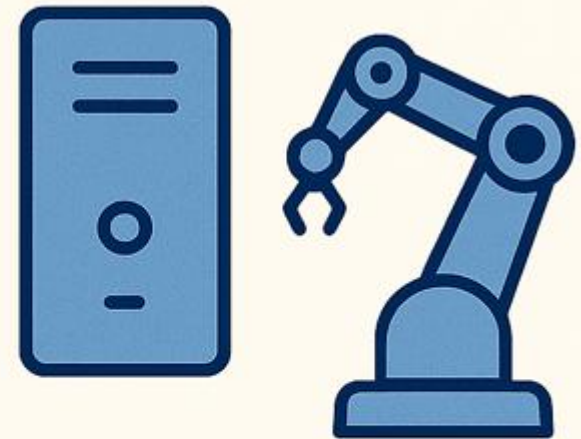
1. Użytkownicy końcowi – osoby korzystające z aplikacji (np. pakiet biurowy, CRM).
2. Administratorzy – zarządzają konfiguracją i bezpieczeństwem.
3. Maszyny i urządzenia – np. sterowniki PLC, IoT.
4. Systemy i komputery – komunikacja w sieciach, integracja usług.



Ludzie



Administratorzy



Maszyny
i komputery

Integracja z chmurą

Azure, AWS, Google Cloud

- Warstwa systemowa umożliwia bezpośrednie połączenie z usługami chmurowymi (np. backup, synchronizacja, zarządzanie VM).
- Wirtualizacja + kontenery (Docker, Kubernetes) pozwalają na przenoszenie aplikacji między lokalnym środowiskiem a chmurą.
- Narzędzia typu Azure Arc czy AWS Systems Manager integrują zarządzanie lokalnymi serwerami z chmurą.

Automatyzacja

PowerShell

- Skrypty do zarządzania systemem Windows, integracja z Azure.
- Ansible
- Automatyzacja konfiguracji w Linux i Windows, zarządzanie wieloma hostami.

Terraform

- Infrastruktura jako kod (IaC) – definiowanie całych środowisk w chmurze.
- CI/CD narzędzia (np. GitHub Actions, Jenkins)
- Automatyzacja wdrożeń i testów.

Pytania

1. Co to jest system komputerowy?
2. Jakie elementy wchodzi w skład jednostki centralnej?
3. Wymień pięć warstw systemu komputerowego.
4. Jakie zadania realizuje warstwa sprzętowa?
5. Podaj przykłady nowoczesnych urządzeń wejścia, wyjścia oraz wejścia-wyjścia.
6. Co to są urządzenia peryferyjne?
7. Czym różni się oprogramowanie narzędziowe od użytkowego?
8. Podaj przykłady oprogramowania użytkowego.
9. Jakie są przykłady wirtualizacji i konteneryzacji oraz ich zastosowania?
10. Podaj przykłady platform chmurowych i ich funkcje.
11. Jakie narzędzia wspierają automatyzację i DevOps?
12. Jakie mechanizmy bezpieczeństwa stosuje się w systemach Linux i Windows?
13. Jakie są przykłady systemów plików i ich cechy?
14. Kim są użytkownicy systemu komputerowego?
15. Dlaczego interfejs urządzeń wejścia-wyjścia jest istotny?
16. Jakie są przykłady komunikacji przewodowej i bezprzewodowej w systemie komputerowym?