مبانی کامپیوتر و برنامه‌نویسی ـ زمستان ۹۷
تدریس توسط: حسین جوهری
دانشکده ریاضی ـ دانشگاه خواجه نصیرالدین طوسی

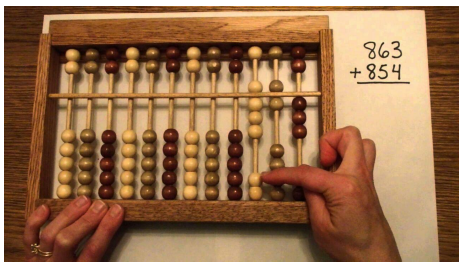
خلاصه هفته اول

مفاهیم: **مقدمات، آشنایی با سیستمهای کامپیوتری**

۱ کاربرد کامپیوترها و اهمیت علوم کامپیوتر

۱.۱ تسریع محاسبات

مکانیزه کردن محاسبات به هدف سرعت بخشیدن به آن انگیزه اصلی گسترش و ساخت کامپیوترها بوده است. از گذشته‌های دور ایده ساخت وسیله‌ای برای سرعت بخشیدن به انجام محاسبات وقتگیر وجود داشته است و ابزارهایی برای این منظور هم ساخته شده است. چرتکه که قدمت آن به حدود سه هزار سال قبل از میلاد میرسد نمونه‌ای خیلی ابتدایی و ساده از یک ابزار مکانیکی جهت تسریع محاسبات از قبیل جمع و تفریق می‌باشد.



شکل ۱: چرتکه

از نمونه‌های اخیرتر می‌توان به ماشین حساب نِپِر [۱] ، ماشین حساب پاسکال [۲] ، ماشین تفاضلی چارلز بابج [۳] و نمونه‌های پیشرفته ماشین حسابهای مکانیکی رومیزی در ابتدای قرن بیستم اشاره کرد. در این ابزارها ورودی عددی عموما با استفاده از کشیدن یک سری دسته، یا پایین آوردن یک سری مهره، به دستگاه داده میشد. در قرن نوزدهم ابزارهای مکانیکی حجیمی با قابلیتهای بیشتر ساخته شد که قادر به گرفتن ورودی از طریق خواندن کارت پانچ بودند. اعداد و ارقام مورد نیاز برای محاسبه با ایجاد یک سری سوراخ در کارتهای پانچ به دستگاه کارتخوان داده میشد که واحد محاسبه را تغذیه میکرد. با پیشرفت تکنولوژی در عرصه الکترونیک و ساخت ترانزیستور، کامپیوترهای دیجیتال خیلی سریعی ساخته شد که سرعت و توانایی محاسباتی به مراتب بالاتری نسبت به نمونه‌های مکانیکی و قبلی خود پیدا کردند. برای مثال، ابرکامپیوترهای امروزی قدرت انجام چندین هزار پتافلاپس (Peta FLOPS) در ثانیه را دارند. هر پتافلاپس معادل 10^{15} عملیات محاسبه‌ای در ثانیه می‌باشد.

---

[۱] John Napier دانشمند فرانسوی قرن شانزدهم، مکشف عدد نپر e و لگاریتم
[۲] Blaise Pascal فیلسوف و ریاضیدان فرانسوی قرن هفدهم
[۳] Charles Babbage ریاضیدان و مبدع بریتانیایی قرن نوزدهم

شکل ۲: یک ماشین حساب مکانیکی دستی

۲.۱ برنامه ریزی و انجام محاسبات مختلف

علاوه بر تسریع محاسبات، ایده ساخته وسیله‌ای که بتواند انواع مختلف برنامه محاسباتی را دریافت و آن را روی اعداد انجام دهد از انگیزه‌های اصلی گسترش کامپیوترها بوده است. ماشین حسابهای اولیه تنها توانایی انجام چند عمل اصلی مثل جمع و تفریق و ضرب را داشتند اما کم کم به توانایی محاسباتی این ابزارها اضافه شد. ماشین تفاضلی چارلز بابج علاوه بر انجام عملیات اصلی توانایی تخمین مقدار چند جمله‌ای ها را هم داشت. طرح اولین نمونه یک ماشین محاسباتی همه منظوره (با معماری نزدیک به کامپیوترهای امروزی) توسط خود چارلز بابج داده شد که البته ساخت آن در زمان خود او امکانپذیر نبود اما بعدها نمونه‌هایی از آن ساخته شد. این دستگاه توانایی ذخیره ۱۰۰۰ عدد چهل رقمی را داشت، حافظه‌ای نزدیک به ۱۷ کیلوبایت. به اضافه یک واحد محاسباتی، مثل نمونه‌ای از پردازشگرهای امروزی، که توانایی انجام چهار عمل اصلی به اضافه رادیکال و غیره را هم داشت. این واحد محاسباتی توانایی با زبانی شبیه زبان اسمبلی امروزی قابل برنامه‌ریزی بود. برای نمونه ماشین بابج توانایی برنامه‌ریزی با استفاده از حلقه‌های محاسباتی و عملگرهای شرطی را دارا بوده است. اِیدا لاولس Ada Lovelace همکار بابج برنامه‌هایی را برای این ماشین طراحی کرده بود که دست نوشته‌های آن اکنون موجود است. به همین خاطر به ایدا لاولس عنوان اولین برنامه نویس تاریخ داده شده است. زبان برنامه نوسی ایدا هم به افتخار او به این نام نامگذاری شده است.

همزمان و مخصوصا در اواخر قرن نوزدهم و اوایل قرن بیستم ریاضیدانان هم علاقه‌مند به پایه‌ریزی تئوری‌ای برای مکانیزه کردن محاسبات بودند. آلن تورینگ Alan Turing ریاضیدان بریتانیایی مدلی تئوریک از یک ماشین محاسبات ساده ارائه کرد که در آن برنامه و داده‌های ورودی روی یک نوار نوشته می‌شود و سپس ماشین با خواندن نوار، برنامه مورد نظر را اجرا و خروجی را روی نوار می‌نویسد. تورینگ ادعا کرد که این مدل ساده، فراگیرترین مدل برای انجام محاسبات مکانیکی است بدین معنی که هر ابزار فیزیکی که مسئله‌ای را حل کند، ماشین تورینگ هم توانایی حل این مسئله را دارد. این ایده به عنوان تز تورینگ معروف است. مدلهای کامپیوتر امروزی در واقع یک نوع پیاده‌سازی از مدل انتزاعی تورینگ هستند. کامپیوترهای امروزی علاوه براجرای انواع و اقسام محاسبات، کاربردها و استفاده‌های فوق العاده و کاملا غیر قابل پیش بینی پیدا کرده‌اند. کاربردهای چندرسانه‌ای، بازیهای کامپیوتری، شبکه اینترنت و توسعه شبکه‌های اجتماعی مجازی همه با وجود کامپیوترها امکان پذیر شده است.

۳.۱ کاربردهای مدرن

امروزه کاربرد کامپیوترها از انجام محاسبات عددی به مراتب فراتر رفته است. در زیر به چند کاربرد متفاوت و مدرن کامپیوترها اشاره می‌شود.

سیستمهای مدیریت اطلاعات

کامپیوترها به ابزارهایی اساسی در سیستمهای مدیریت اطلاعات تبدیل شده‌اند که کار دریافت، ذخیره‌سازی و بروزسازی اطلاعات را برعهده دارند. برای مثال امروزه هر دانشگاهی یک سیستم مدیریت اطلاعات دارد که کار ثبت نام، انتخاب دروس و ثبت نمرات را انجام می‌دهد. امروزه اداره بانکها، بیمارستانها، ارگانها و خیلی از شرکتها و بنگاههای کسب وکار بدون کامپیوتر غیرقابل تصور است.

سیستمهای چندرسانه‌ای

نشریات و کتب الکترونیک و بازیهای کامپیوتری از کاربردهای مدرن کامپیوتر هستند که هر روز دچار تحول میشوند. بالاخص با توسعه شبکه اینترنت، هر کامپیوتر به یک نقطه ارتباطی در دنیایی وسیع تبدیل شده است که ارتباطات چندرسانه‌ای را فراهم می‌کند.

شبکه‌های مجازی و اینترنت

کامپیوترها امکان دسترسی سریع به اطلاعات را در هر جای دنیا که باشد فراهم می‌سازند. توسعه ایمیل، موتورهای جستجو مانند گوگل و یاهو، شبکه‌های اجتماعی مجازی مثل فیسبوک و تویتر، ساختار ارتباطی جوامع را کاملا دگرگون کرده است.

هوش مصنوعی و یادگیری ماشین

در چند دهه اخیر، کامپیوترها در شبیه‌سازی توانایی های انسانی پیشرفتهای قابل ملاحظه‌ای بدست آورده‌اند. برای مثال امروزه سیستمهای تشخیص گفتار با درصد خطای بسیار ناچیز قادر به شناسایی و تبدیل گفتار انسانی به متن هستند. نرم افزارهای داده‌کاوی با پیداکردن الگوهای آماری در داده‌های بزرگ به مدیران و صاحبان کسب و کار در برنامه‌ریزی یاری می‌رسانند. رباتهای هوشمند نقش مهمی در کارخانجات و پروسه ساخت و تولید پیدا کرده‌اند.

۴.۱ فرصتها، چالشها و تبعات

"شاید پنج کامپیوتر در دنیا قابل فروش باشد." توماس واتسون. مدیر شرکت IBM سال ۱۹۴۳

برخلاف پیش بینی توماس واتسون، امروزه کامپیوترها و وسایل ارتباطی هوشمند در سطح وسیعی بکاربرده می‌شوند. طبق آمار وبسایت internetworldstats.com در سال ۲۰۱۷ نصف مردم دنیا کاربر اینترنت شده‌اند. برای خیلی از مردم کامپیوتر به یک وسیله ضروری برای کار و زندگی تبدیل شده است. اما استفاده از ابزارهای رایانشی و ارتباطی هوشمند، چالشهایی را نیز با خود به همراه آورده است. در اینجا به چند نمونه از این چالشها اشاره می‌کنیم.

تاثیر شبکه‌های مجازی بر ساختار اجتماعی و ارتباطات

شبکه‌های مجازی مثل فیسبوک و تلگرام باعث تشکیل گروهها و تسریع اطلاع رسانی و دسترسی بیشتر شده است. اما در عین حال درصد بالایی از اعضا و کاربران این شبکه‌ها به جای ارتباط مستقیم با خانواده، مردم و محیط اطراف خود سهم قابل توجهی از وقت خود را در محیطهای مجازی می‌گذرانند. منتقدان این شبکه‌ها معتقدند که شبکه‌های مجازی باعث اتلاف وقت و پایین آمدن کیفیت ارتباطات انسانی شده است.

امنیت داده و حریم خصوصی

افزایش سطح دسترسی و امکان ارتباط در سطح وسیع لازمه اصلی‌اش ایجاد بستری مطمئن و قابل اعتماد است که حریم خصوصی و داده‌های افراد در آن محافظت شوند. رمزنگاری یک زمینه تحقیقی وسیع است که کاربرد زیادی پیدا کرده است و هدفش ایجاد متدها و پروتکهای ارتباطی ایمن برای نقل و انتقال داده است. با وجود روشهای پیشرفته و ایمن برای حفاظت از داده‌ها در مقابل تهدیداتی مثل استراق سمع، سرقت هویت و حملات شبکه‌ای و نرم افزاری مختلف، بدلیل سهل انگاری کاربران، اشتباهات انسانی و ضعف سیستمها، سالانه خسارات زیادی به سازمانها و افراد وارد می‌شود.

کالای دیجیتال و مسئله حق مولف

کامپیوترها و شبکه اینترنت امکان تولید و نشر انواع و اقسام کالای دیجیتال از قبیل نشریات الکترونیک، موسیقی، فیلم و ویدیو و نرم افزار را فراهم کرده است و بازار این محصولات را گسترده‌تر کرده است. از طرفی دیگر امکان کپی این محصولات چالشهای جدیدی را برای حفاظت از حقوق مولف و ناشر ایجاد کرده است که قبلا وجود نداشت. علاوه بر این شرکتهای بزرگ نرم افزاری مثل آمازون، مایکروسافت و گوگل عملا بازار خیلی از سرویسها و کالاهای نرم افزاری را بصورت انحصاری در اختیار خود گرفته‌اند بطوری که امکان رقابت برای شرکتهای کوچکتر وجود ندارد.

تکنولوژی و مصرف گرایی

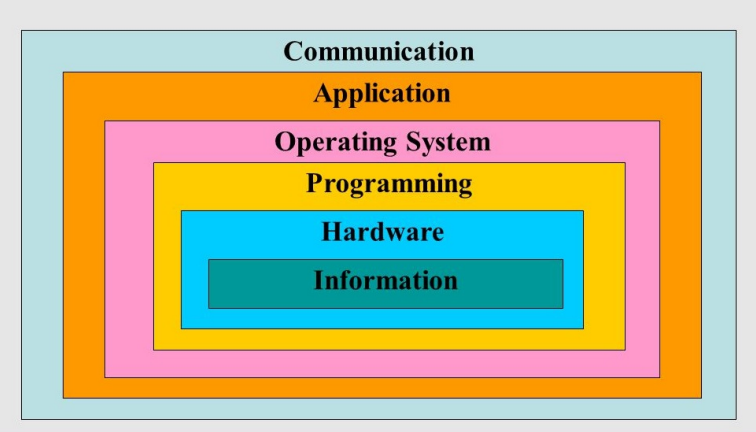
نظام سرمایه‌داری و تکنولوژی ارتباط تنگاتنگ و مستقیمی با هم دارند. نظام سرمایه‌داری برای تولید و فروش بیشتر نیازمند معرفی نیازهای جدید و معرفی تنکولوژی‌های جدید است و این خود به بازار تکنولوژی و علوم مهندسی مربوطه رونق بسیاری بخشیده است. خیلی از تکنولوژیهای موجود، بالاخص در زمینه استفاده شخصی، صرفا برای تولید و سوددهی بیشتر عرضه می‌شوند بدون اینکه یک نیاز واقعی را برآورده کنند و یا کیفیت زندگی را بهبود ببخشند.

۲ سیستم کامپیوتری، مفاهیم اولیه

کامپیوتر فقط یک ابزار است درحالیکه یک سیستم کامپیوتری مجموعه‌ای پویا از سخت افزار و نرم افزار و داده‌ها است که درتعامل با یکدیگر کار خاصی را انجام می‌دهند. در این میان، سخت افزار مجموعه‌ای از قطعات و ابزار است که قسمتهای مختلف یک کامپیوتر را از لحاظ فیزیکی تشکیل می‌دهند مانند مادربورد، پردازنده، تراشه‌های حافظه، سیمها، هارد دیسک، صفحه کلید، مونیتور و غیره. نرم افزار کامپیوتر مجموعه‌ای از برنامه‌هاست از قبیل سیستم عامل، مرورگر وب، برنامه‌های کاربردی و غیره که با استفاده از کامپیوتر اجرا می‌شوند. در قلب یک سیستم کامپیوتری اطلاعاتی است که پردازش و مدیریت می‌شوند. بدون داده و اطلاعات، سخت افزار و نرم افزار استفاده‌ای ندارد.

۱.۲ لایه‌های یک سیستم کامپیوتری

یک سیستم کامپیوتری مثل یک پیاز از چند لایه تشکیل شده است. هر لایه نقش خاصی را در طرح کلی سیستم بر عهده دارد. این لایه‌ها که در شکل زیر نشان داده شده‌اند، سطوح مختلفی از انتزاع abstraction را فراهم می‌آورند. این بدین معنی است که لایه‌های بالاتر نیاز به دانستن جزئیات لایه‌های پایینتر ندارند و مکانیزم درونی هر لایه مستقل از لایه‌های دیگر است.



شکل ۳: لایه‌های یک سیستم کامپیوتری

در بالاترین لایه انتزاع برنامه‌های کاربردی و ارتباطی قرار دارند مانند ویرایشگر متن، بازیهای کامپیوتری، مرورگر وب و غیره. این لایه با کاربر سیستم در تعامل است، داده‌ها و فرامین را گرفته و طبق دستورالعمل برنامه عمل می‌کند. در لایه پایینتر سیستم عامل قرار دارد. سیستم عامل وظیفه مدیریت برنامه‌های کاربردی را برعهده دارد. معمولا یک سیستم کامپیوتری در زمان واحد چندین برنامه کاربردی با هم اجرا می‌کند. سیستم عامل وظیفه تخصیص منابع پردازشی و حافظه را برعهده دارد و برنامه‌ها را برای اجرا زمانبندی می‌کند. در لایه پایینتر برنامه‌های کامپیوتری قرار دارند که بصورت کد ماشین در آمده تا اجرای آن برای کامپیوتر قابل فهم باشد. کد ماشین مجموعه‌ای از دستورالعملهای ریز است که در حافظه اصلی قرار می‌گیرد و توسط پردازنده کامپیوتر بصورت پشت سر هم اجرا می‌شوند. در لایه پایین تر سخت افزار سیستم قرار دارد که مجموعه‌ای از ابزارهای فیزیکی برای اجرای کد ماشین و نقل و انتقال و پردازش داده‌های برنامه‌های کاربردی می‌باشد. در قلب یک سیستم کامپیوتری داده‌ها و اطلاعات قرار دارد. اطلاعات بصورت داده‌های بیتی صفر و یک در آمده و با استفاده از سخت افزار مربوطه در حافظه اصلی قرار داده می‌شوند و یا در میان قسمتهای مختلف ورودی/خروجی و پردازشی در حال جابجایی هستند.

لایه اطلاعات Information

لایه اطلاعات یک لایه مفهومی است که مرتبط با طرز نمایش اطلاعات در کامپیوتر در پایین ترین سطح ممکن است، در جایی که با سخت افزار ارتباط پیدا می‌کند. اطلاعات در کامپیوتر در پایینترین سطح خود بصورت اعداد و بالاخص اعداد باینری ۰ و ۱ نمایش داده می‌شوند. حتی حروف نیز بصورت اعداد کدگذاری شده و ذخیره می‌شوند. در اینجا به اجمال اعداد و شیوه نمایش آنها در کامپیوتر را مورد بررسی قرار می‌دهیم.

عدد در واقع عضوی کوچک از یک سیستم مفهومی در ریاضیات است (مثل گروه و میدان) که قواعد و قوانین بر روی آن اعمال می‌شود. برای مثال، قاعده ترتیب، اعداد یک سیستم را مرتب می‌کند و یا اعمال جمع و ضرب و تفریق روابطی را میان اعداد یک سیستم نشان می‌دهد. مجموعه اعداد طبیعی $N = \{0, 1, 2, 3, \}$ با عملگرهای جمع و تفریق و ضرب یک سیستم معمولی از اعداد است که ما بصورت روزمره با آن سروکار داریم. اعداد گویا، اعشاری و اعداد حقیقی هم با عملگرهای جمع و تفریق و ضرب و تقسیم از سیستم‌های معمول برای اعداد هستند. ابتدا به طرز نمایش اعداد طبیعی می‌پردازیم.

اعداد طبیعی بطور معمول در مبنای 10 نمایش داده می‌شوند به این معنا که در برای نشان دادن ارقام آنها از ده رقم $\{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ استفاده می‌شود. مقدار یک عدد از روی ارقام و مکان ارقام تعیین می‌شود. به شیوه نمایش اعداد بر مبنای 10 دهدهی یا دسیمال decimal گفته می‌شود. برای مثال عدد 1529

معادل $1 \times 10^3 + 5 \times 10^2 + 2 \times 10^1 + 9 \times 10^0$ می‌باشد. بطور کلی برای نمایش عدد n در مبنای b از b نشانه $\{0, 1, 2, 3, .., b-1\}$ استفاده می‌شود و معادل دهدهی آن بصورت زیر محاسبه می‌شود.

$$n = (a_{k-1}a_{k-2}\dots a_1a_0)_b = a_{k-1} \times b^{k-1} + a_{k-2} \times b^{k-2} + \dots + a_1 \times b^1 + a_0 \times b^0$$

در کامپیوتر اعداد در مبنای ۲ یا اصطلاحا بصورت باینری نمایش داده می‌شوند. برای مثال عدد باینری 100101 معادل $1 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$ در مبنای ۱۰ می‌باشد.

به یک رقم باینری یک بیت bit گفته می‌شود. به یک عدد باینری متشکل از ۸ رقم یک بایت Byte گفته می‌شود. Word مجموعه‌ای از بایتهاست که یک واحد ذخیره‌سازی اطلاعات در کامپیوتر می‌باشد. به تعداد بیتها در یک word اصطلاحا طول کلمه word length کامپیوتر گفته می‌شود.

علاوه بر مبنای دو، مبنای ۸ و مبنای ۱۶ هم نقش اساسی در محاسبات و ذخیره‌سازی کامپیوتری دارند. اعداد در مبنای 8 از هشت رقم $\{0, 1, 2, 3, 4, 5, 6, 7\}$ تشکیل شده اند در حالیکه برای نمایش اعداد در مبنای 16 علاوه بر اعداد از حروف هم استفاده می‌شود.

0,1,2,...,9,A,B,C,D,E,F

در اینجا حرف A معادل عدد 10 می‌باشد. B معادل با 11 است و به همین شکل تا حرف F که معادل 15 می‌باشد. برای مثال عدد $E24A$ در مبنای شانزده معادل $14 \times 16^3 + 2 \times 16^2 + 4 \times 16^1 + 10 \times 16^0$ در مبنای ده می‌باشد.

حروف یا بطور کلی کاراکترها در کامپیوتر بصورت کدهای عددی نمایش داده می‌شوند. از قدیمی‌ترین کدها برای نمایش حروف کد اسکی ASCII می‌باشد که برای نمایش حروف انگلیسی و یک سری نشانه‌های اضافی به‌کار می‌رود. کد اسکی برای هر کاراکتر ۸ بیت در نظر می‌گیرد و به این ترتیب می‌تواند تا ۲۵۶ کاراکتر مختلف را نمایش دهد. این کد هر کاراکتر را با دو عدد دهدهی نمایش می‌دهد. عدد سمت چپ و عدد سمت راست. این اعداد البته بصورت باینری ذخیره می‌شوند. جدول زیر کد اسکی کاراکترهای مختلف را نشان می‌دهد.

Left Digit(s) \ *Right Digit*	*ASCII* 0	1	2	3	4	5	6	7	8	9
0	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT
1	LF	VT	FF	CR	SO	SI	DLE	DC1	DC2	DC3
2	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS
3	RS	US	□	!	“	#	$	%	&	’
4	(	)	*	+	,	-	.	/	0	1
5	2	3	4	5	6	7	8	9	:	;
6	<	=	>	?	@	A	B	C	D	E
7	F	G	H	I	J	K	L	M	N	O
8	P	Q	R	S	T	U	V	W	X	Y
9	Z	[	\	]	^	_	`	a	b	c
10	d	e	f	g	h	i	j	k	l	m
11	n	o	p	q	r	s	t	u	v	w
12	x	y	z	{	|	}	~	DEL		

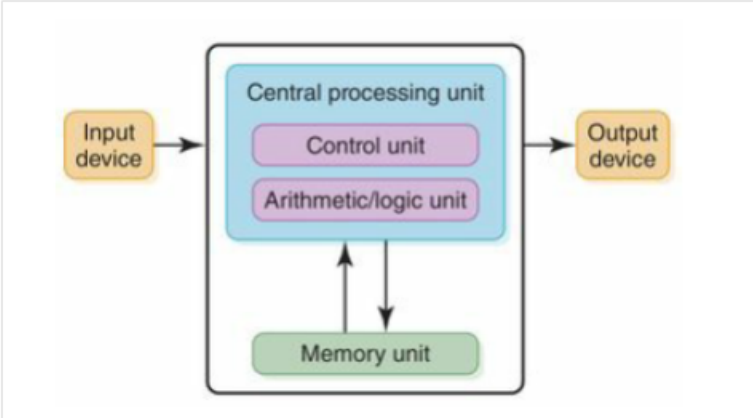
برای پوشش دادن زبانها و کاراکترهای مختلف دیگر، کدهای دیگری طراحی شده است. فراگیرترین کد که امروز به استانداردی جهانی تبدیل شده است کد Unicode است که برای هر کاراکتر از ۱۶ بیت استفاده می‌کند. به این ترتیب Unicode می‌تواند تا 2^{16} کاراکتر مختلف را پوشش دهد. جدول زیر چند نمونه از کد Unicode را نشان می‌دهد که در مبنای ۱۶ نوشته شده است.

Code (Hex)	Character	Source
0041	A	English (Latin)
042F	Я	Russian (Cyrillic)
OE09	ก	Thai
13EA	Ꮺ	Cherokee
211E	℞	Letterlike Symbols
21CC	⇌	Arrows
282F	⠯	Braille
345F	㑟	Chinese/Japanese/ Korean (Common)

لایه سخت افزار Hardware

لایه سخت افزار یا لایه فیزیکی، مجموعه‌ای از قطعات و دستگاههایی است که بخشهای مختلف یک سیستم کامپیوتری را از لحاظ فیزیکی تشکیل می‌دهد. در کامپیوترهای امروزی، در سطح خیلی پایین، سخت افزار از مدارات، سویچها و گیتهای الکترونیکی که برای پردازش سیگنال استفاده می‌شود تشکیل شده است. بعضی از این مدارات الکترونیکی عملگرهای ساده منطقی مثل AND و OR را پیاده‌سازی می‌کنند و در سطوح بالاتر اعمال اصلی محاسباتی مثل جمع و ضرب و امکان نگهداری داده در حافظه را فراهم می‌آورند.

در سطح بالاتر، قطعات سخت افزاری در یک معماری منسجم با هم در ارتباط هستند و هر کدام وظایف و نقشی را در سیستم بر عهده دارد. برای مثال در معماری وان نیومن von Neumann که اساس معماری کامپیوترهای امروزی است، داده و دستورالعملها هر دو در حافظه نگهداری می‌شوند و واحد محاسباتی که دستورالعملها را اجرا می‌کند یک واحد مجزا را به خود اختصاص داده است. شکل زیر ساختار کلی معماری وان نیومن را نشان می‌دهد.

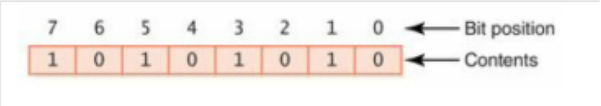


شکل ۴: در معماری وان نیومن سخت افزار کامپیوتر از چهار واحد اصلی تشکیل شده است: واحد پردازش مرکزی، واحد حافظه، واحد ورودی، واحد خروجی

۱. **واحد حافظه** کار نگهداری داده‌ها و برنامه‌ها را بر عهده دارد. حافظه اصلی بصورت ماتریسی از خانه‌های کوچک است که هر کدام واحدی از حافظه است. هر خانه حافظه از یک تعداد بایت تشکیل شده است. برای مثال یک حافظه $4GB$ از 4×2^{30} بایت تشکیل شده است. هر بایت از حافظه می‌تواند آدرسی منحصر بفرد داشته باشد. وقتی کامپیوتر می‌خواهد به محتوای یک بایت از حافظه دسترسی پیدا کند باید آدرس آن محل از حافظه را بداند. در شکل زیر حافظه در دو ستون نشان داده شده است. ستون سمت چپ محتوای حافظه را نشان می‌دهد در حالیکه ستون سمت راست آدرس هر خانه از حافظه را نشان می‌دهد. توجه کنید که ستون آدرس در حافظه ذخیره نمی‌شود و فقط برای فهم مطلب نمایش داده شده است. شماره بیتها در حافظه از سمت راست شروع می‌شود. در هر خانه از

Address	Contents
00000000	11100011
00000001	10101001
. . .	. . .
11111100	00000000
11111101	11111111
11111110	10101010
11111111	00110011

حافظه، بیت انتهای سمت راست بیت شماره صفر است و بیت سمت چپ آن بیت شماره یک و به این ترتیب آخرین بیت، بیت انتهای سمت چپ است.



حافظه اصلی

دو نوع حافظه اصلی وجود دارد:

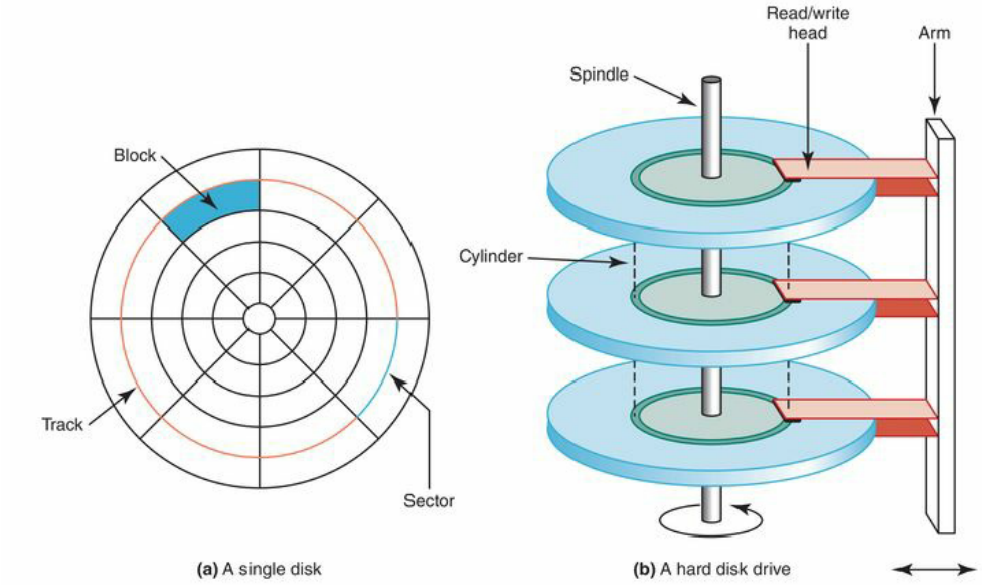
حافظه RAM که مخفف Random Access Memory می‌باشد و موقع روشن بودن کامپیوتر داده‌ها و برنامه‌ها را در خود نگه میدارد. در واقع RAM حافظه کاری کامپیوتر است. حافظه RAM قابل تغییر و پاک شدن است و محتوای هر خانه آن بطور مستقیم قابل دسترسی است. وقتی کامپیوتر خاموش می‌شود محتوای حافظه RAM بطوری کلی از دست می‌رود و با هر روشن شدن محتوای جدیدی پیدا می‌کند.

حافظه ROM که مخفف Read Only Memory است. معمولا یک حافظه کوچک است که محتوای آن ثابت است و تغییر نمی‌کند. هر کامپیوتر معمولا یک قطعه حافظه ROM دارد که حاوی دستورالعملهایی برای شروع کار کامپیوتر است. تا زمانی که اختیار کامپیوتر به برنامه سیستم عامل سپرده نشود دستورالعملهای برنامه ROM اجرا می‌شود.

حافظه جانبی

علاوه بر حافظه اصلی، کامپیوترها معمولا مجهز به حداقل یک حافظه جانبی هستند تا داده‌ها و برنامه‌ها بصورت ماندگار در آنجا نگه‌داری شوند. حافظه‌های جانبی معمولا از لحاظ سرعت دسترسی به محتوا بسیار کندتر از حافظه اصلی هستند. معمولترین نوع حافظه جانبی، دیسکهای مغناطیسی یا Hard

disk ها هستند. یک دیسک مغناطیسی از چند لایه یا طبقه تشکیل شده است. هر لایه یک صفحه نازک دیسک به همراه یک قلم خواندن و نوشتن Read/Write head دارد. صفحه دیسک از مواد مغناطیسی درست شده است بطوری که توانایی نگه‌داری اطلاعات را برای دراز مدت دارا باشد بدون اینکه نیازی به اتصال به برق داشته باشد. سطح دیسک به چند نوار دایره‌ای هم مرکز تقسیم می‌شود. به هر نوار دایره‌ای یک شیار (track) گفته می‌شود. هر شیار به یک سری کمان تقسیم می‌شود که به آنها سکتور sector گفته می‌شود. متناظر با هر سکتور یک قطعه از دیسک وجود دارد که به آن بلاک block گفته می‌شود. داده‌ها در بلاکها ذخیره می‌شوند. اهرم خواندن و نوشتن با گرفتن آدرس بلاک دلخواه حرکت کرده و روی تراک مربوطه قرار می‌گیرد. دیسک با سرعت بالا میچرخد و بلاک مورد نظر از زیر قلم خواندن رد می‌شود و محتوای آن خوانده می‌شود. یک هارد دیسک ساده، بطور معمول ۷۲۰۰ بار در دقیقه می‌چرخد. هر چه سرعت چرخش بیشتر باشد، سرعت خواندن و نوشتن داده‌ها هم بیشتر خواهد بود. شکل زیر ساختار درونی یک دیسک سخت را نشان می‌دهد.



نوع دیگر حافظه‌های جانبی حافظه‌های نوری هستند مثل CD ROM و DVD ROM که از برجستگی و انعکاس نور برای خواندن و نوشتن حافظه استفاده می‌کنند. صفحات سی دی رام مثل هارد دیسک به یک سری سکتور تقسیم می‌شود اما بر خلاف هارد دیسک تنها یک شیار دارد که از مرکز شروع شده و بصورت مارپیچ کل صفحه را می‌پوشاند. اطلاعات سی‌دی رام به محض نوشتن دیگر قابل تغییر نیست. هر بیت بصورت یک برجستگی یا تورفتگی در سطح دیسک ایجاد می‌شود. دستگاه سی‌دی رام شعاع نوری بر سطح مربوطه می‌تاباند. زاویه انعکاس نور مشخص می‌کند که سطح مربوطه برجستگی یا تورفتگی بوده است و از این طریق مقدار بیتی آن محل مشخص می‌شود.

۲. **واحد پردازش مرکزی** یا CPU که خود متشکل از دو قسمت واحد کنترل و واحد محاسباتی و منطقی است. بخش واحد کنترل وظیفه مدیریت اجرای دستورالعملها را بر عهده دارد. این واحد بین بخشهای مختلف هماهنگی ایجاد می‌کند تا داده‌ها و دستورالعملهای برنامه برای اجرا از حافظه به پردازنده منتقل شوند و دستورات به ترتیب پشت سر هم اجرا شوند. واحد محاسباتی و منطقی مجموعه مداراتی است که اعمال اصلی مثل جمع و ضرب و تقسیم و عملگرهای منطقی را پیاده‌سازی می‌کند. این واحد دارای

یک سری رجیستر register است. هر رجیستر مثل یک خانه حافظه است که اعداد و داده‌های لازم برای اجرای دستورالعملها را بصورت موقت نگه‌داری می‌کند.

۳. **واحد ورودی** شامل ابزار و دستگاههایی برای گرفتن داده از کاربر و دنیای بیرون هستند مانند صفحه کلید، اسکنر، ماوس و غیره.

۴. **واحد خروجی** وظیفه‌اش نمایش اطلاعات و خروجی برنامه‌هاست. مثل مانیتور، اسپیکر و چاپگر.

لایه سیستم عامل Operating System

برنامه‌هایی که در کامپیوتر اجرا می‌شوند بطور کلی به دو دسته کاربردی و سیستمی تقسیم می‌شوند. برنامه‌های کاربردی برنامه‌هایی هستند که نیازهای ویژه ما را برآورده می‌کنند. برای مثال برنامه ویرایشگر متن، برای نوشتن و ویرایش متن بکار می‌رود. برنامه مرورگر وب برای اتصال به وبسایتها و دانلود کردن صفحات وب و دیگر منابع روی وب مورد استفاده قرار می‌گیرد. اما دسته‌ای دیگر از برنامه‌ها هستند که، در سطح پایینتر، یک سیستم کامپیوتری را مدیریت می‌کنند و زمینه را برای اجرای برنامه‌های کاربردی فراهم می‌آورند. به این نوع برنامه‌ها برنامه‌های سیستمی گفته می‌شود. برنامه‌های سیستمی در ارتباط مستقیم با سخت افزار هستند و مثل یک مترجم امکان ارتباط بین کاربر و سخت افزار را فراهم می‌کنند.

یکی از مهمترین برنامه‌های سیستمی، برنامه سیستم عامل است. سیستم عامل وظیفه مدیریت منابع سخت افزاری را برعهده دارد (مثل پردازنده، حافظه اصلی و جانبی و ورودی و خروجی) و برنامه‌های مختلف را برای اجرا زمان بندی می‌کند. معمولا در هر سیستم کامپیوتری در یک زمان چند برنامه کاربردی در حال اجرا هستند. سیستم عامل باید منابع کامپیوتری را که محدود هستند میان این برنامه‌ها بطرز مطلوبی تقسیم کند و اجرای آنها را زمانبندی کند. سیستم عامل برای هر برنامه مکانی در حافظه در نظر می‌گیرد و اجرای آن را همگام با اجرای برنامه‌های دیگر مدیریت می‌کند. برای مثال ممکن است چند برنامه همزمان نیاز به چاپ مطلبی در خروجی داشته باشند. در این موقعیت، سیستم عامل صفی از برنامه‌ها برای چاپ خروجی ایجاد می‌کند و به ترتیب خاصی کنترل خروجی را در اختیار این برنامه‌ها قرار می‌دهد.

از سیستم عاملهای فراگیر می‌توان به Microsoft Windows، نسخه‌های مختلف لینوکس Linux مانند Ubuntu و Fedora و سیستم‌عاملهای موبایل مانند Android و iOS اشاره کرد.

لایه برنامه‌های کاربردی و ارتباطی

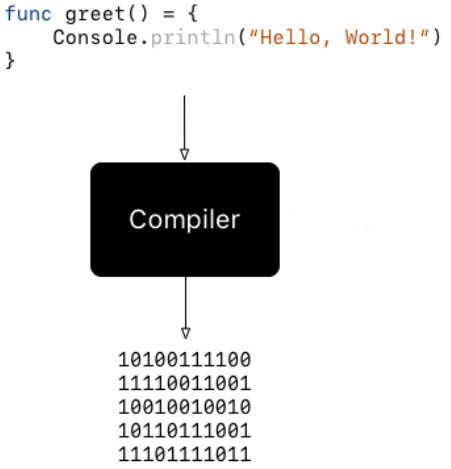
این لایه نزدیکترین لایه به کاربر سیستم است. در این لایه برنامه‌های کاربردی از قبیل ویرایشگر متن، برنامه‌های ویرایش تصویر، پخش چند رسانه‌ای، مرورگر وب و بازیهای کامپیوتری قرار دارند.

لایه برنامه‌نویسی Programming

یک لایه مفهومی است که مرتبط با برنامه‌هایی است که سخت افزار کامپیوتر آنها را اجرا می‌کند. هر برنامه کامپیوتری زنجیره‌ای از دستورالعملهاست که توسط کامپیوتر اجرا می‌شود و با زبان خاصی بیان شده است. زبانی که برای توصیف دستورالعملها به کار می‌رود می‌تواند یک زبان سطح بالا باشد (یعنی با جملاتی بیان شود که به زبان و شهود انسانی نزدیک است) و یا اینکه زبانی سطح پایین باشد که نزدیک به زبان ماشین است. زبان ماشین یا کد ماشین machine code مجموعه‌ای از دستورالعملها و کدهای باینری است که مستقیما توسط پردازنده و سخت افزار کامپیوتر قابل اجراست. همانطور که هر قفل کلید مختص به خود را دارد، هر نوع پردازنده هم زبان ماشین مختص به خود را دارد.

برای اینکه برنامه‌ای که با زبانی سطح بالا نوشته شده است توسط کامپیوتر اجرا شود باید آن برنامه به زبان ماشین تبدیل شود و یا اصطلاحا به زبان ماشین ترجمه شود. به عمل تبدیل برنامه از زبان سطح بالا به زبان

سطح پایین کامپایل compile یا ترجمه گفته می‌شود. به نرم افزاری که عمل ترجمه را انجام می‌دهد کامپایلر گفته می‌شود.

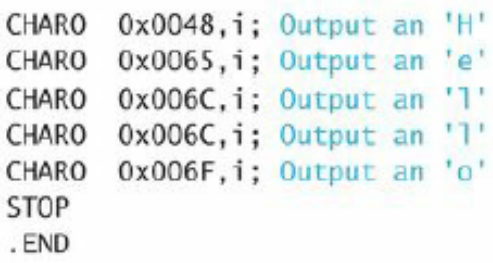


زبانهای برنامه‌نویسی از لحاظ نزدیکی به زبان انسان و زبان ماشین در طیفی از انتزاع قرار می‌گیرند. همانطور که گفته شد، در پایین ترین سطح زبان ماشین قرار دارد که مجموعه ای از دستورالعملهای ریز است که با یک سری کد باینری نمایش داده می‌شوند. هر نوع پردازنده زبان مختص به خود را دارد. در شکل زیر نمونه‌ای از یک سری دستورالعمل به زبان کد ماشین Pep/8 نشان داده شده است. در ستون سمت چپ مفهوم هر دستور نوشته شده است. در ستون وسطی کد دستورالعمل نشان داده شده است و در ستون سمت راست معدل کد دستورالعمل در مبنای ۱۶ (مبنای هگزادسیمال) بیان شده است.

Action	Binary Instruction	Hex Instruction
Write 'H'	**01010000** 0000000001001000	50 0048
Write 'e'	**01010000** 0000000001100101	50 0065
Write 'l'	**01010000** 0000000001101100	50 006C
Write 'l'	**01010000** 0000000001101100	50 006C
Write 'o'	**01010000** 0000000001101111	50 006F
Stop	**00000000**	00

در سطح بالاتر، زبانهایی مثل اسمبلی Assembly قرار دارد که به جای کد باینری دستورالعمل از کلمات قابل فهم استفاده می‌شود تا برای برنامه‌نوسی قابل فهمتر باشد. کد اسمبلی بسیار نزدیک به زبان ماشین است و معمولا هر نوع پردازنده زبان اسمبلی مختص به خود را دارد. به برنامه‌ای که زبان اسمبلی را به زبان ماشین ترجمه می‌کند اسمبلر Assembler گفته می‌شود. شکل زیر نمونه‌ای از کد اسمبلی را نمایش می‌دهد که معادل کد ماشینی است که در شکل قبلی نمایش داده شده بود.

در سطوح بالاتر، زبانهای برنامه نویسی مثل Python , Java , Basic, C++ قرار دارند. برنامه‌هایی که به این زبانها نوشته می‌شود به زبان انسانی نزدیکتر هستند و برنامه‌نویسی با آنها بسیار سریعتر و راحتتر از نوشتن کد اسمبلی است. ویژگی مهم این زبانها این است که وابسته به یک پردازنده خاص نیستند و برنامه‌ای



شکل ۵: یک نمونه از کد اسمبلی

که با این زبانها نوشته می‌شود در صورتی که توسط کامپایلر مناسب ترجمه شود در هر نوع پردازنده‌ای قابل اجراست. این یک مزیت بسیار مهم است چون در این صورت برنامه‌نویس نیازی به در نظر گرفتن نوع و جزئیات پردازنده و سخت افزار مقصد ندارد و می‌تواند بر روش حل مسئله و مسائل سطح بالاتر تمرکز کند.

زبانهای برنامه‌نویسی سطح بالاتری هم مثل SQL وجود دارند که درجه‌ای بالاتری از انتزاع را فراهم می‌آورند. برای مثال دستور زیر به این معنی است که همه اطلاعات مربوط به دانشجویان از پایگاه داده استخراج شده و در حافظه اصلی قرار گیرد. برای پیاده سازی این دستور در زبان C ممکن است به چند صفحه کد نیاز داشته باشیم.

```
Select * From Students
```

الگوریتم و شبه کد. الگوریتم یک مفهوم اساسی در علوم کامپیوتر است اما نمونه‌های آن تقریبا همه جا پیدا می‌شود. بطور کلی الگوریتم توصیف قدم به قدم انجام کاری است. برای مثال دستور تهیه یک غذای خاص، شیوه کار با یک ماشین لباس شویی، مراحل لازم برای پست کردن یک نامه و طرز انجام یک بازی همه نمونه‌هایی از الگوریتمها هستند که در زندگی روزمره یافت می‌شوند.

هر الگوریتم ترتیبی از مراحل را برای حل مسئله مورد نظر بیان می‌کند. الگوریتم ها معمولا یک سری ورودی دارند که داده‌های مورد نیاز برای حل مسئله هستند و یک سری خروجی تولید می‌کنند که نتیجه حل مسئله می‌باشد. برای مثال ورودی مسئله می‌تواند دو عدد باشد و خروجی آن بزرگترین مقسوم علیه مشترک این دو عدد باشد.

مراحل و گامهای الگوریتم معمول توسط زبانی سطح بالا و نزدیک به زبانهای انسانی بیان می‌شود (می‌توان از کلمات و جلمات فارسی هم در آن استفاده کرد). به این نوع بیان حل مسئله "شبه کد" گفته می‌شود چون چیزی شبیه کد برنامه نویسی است. بطور معمول برنامه‌نویس ها در مرحله اول حل مسئله را بصورت شبه کد می‌نویسند و سپس آن را تبدیل به کد زبان برنامه‌نویسی دلخواه خود می‌کنند.

برای مثال الگوریتم زیر یک عدد را دریافت کرده و مقدار قدرمطلق آن را در خروجی چاپ می‌کند.

```
1. Read X
2. A <-- X
3. If X > 0 go to step 5.
4. A <-- -X
5. Print A
6. End
```

با بیان رسمی‌تر، الگوریتم بیان دقیق و بدون ابهام یک لیست از قدمهای محاسباتی است که اجرای یک پروسه خاتمه پذیر را توصیف می‌کند.

نکاتی در مورد این تعریف:

- ترتیب اجرای قدمها از اهمیت فراوانی برخوردار است. هر الگوریتم باید ساختاری واضح و روشن برای ترتیب اجرا داشته باشد.
- قدمهای یک الگوریتم باید در زمانی متناهی قابل محاسبه باشند. به مثال زیر توجه کنید. "همه اعداد طبیعی را چاپ کن. " از آنجا که بی نهایت عدد طبیعی وجود دارد اجرای این کار ممکن نیست. اما قدمی مثل " بزرگترین عدد اول کوچکتر از ۲۱۰۰۰۰ را پیدا کن." گرچه اجرایش زمان زیادی را می‌طلبد قابل اجراست و مطمئنا در زمانی معین خاتمه می‌پذیرد.
- نکته مهم دیگر این است که توصیف الگوریتم باید بدون ابهام باشد. این به این معنی است که در هر مرحله از الگوریتم اطلاعات لازم برای اجرای قدم بعدی موجود باشد و اجرا بدون ابهام و نیاز به طراحی بیشتر، صرفا با پیروی از دستورات امکانپذیر باشد.

شکل زیر توصیفی سطح بالا از الگورتیم اقلیدس برای پیداکردن بزرگترین مقسوم علیه مشترک را بیان می‌کند. (این خود نشان می‌دهد که الگوریتمها خیلی قبل از عصر کامپیوتر در مباحث علمی و ریاضیاتی وجود داشته اند اما بطور دقیق به عنوان موضوعی مستقل مورد مطالعه نبوده‌اند. خود واژه الگوریتم algorithm در قرن بیستم مستعمل شده و به افتخار ریاضیدان مسلمان و ایرانی قرن هشتم، از اسم محمد بن موسی خوارزمی گرفته شده است.)

الگوریتم پیدا کردن بزرگترین مقسوم علیه مشترک دو عدد
ورودی: دو عدد طبیعی N و M

۱. (فرض کنید M عدد بزرگتر باشد.) اگر M عدد بزرگتر نبود جای M و N را عوض کن.

۲. M را بر N تقسیم کن و باقیمانده را در R قرار بده.

۳. اگر R صفر است، N را به عنوان جواب مسئله گزارش کن در غیر این صورت M را با N مقداردهی کن و مقدار N را R قرار بده و به قدم ۲ برگرد.

همین الگوریتم را می‌توان با توصیفی که به زبان ریاضیات نزدیکتر است بصورت زیر بیان کرد.

الگوریتم پیدا کردن بزرگترین مقسوم علیه مشترک دو عدد
ورودی: دو عدد طبیعی N و M

۱. اگر $M < N$ آنگاه $N \longleftrightarrow M$.

۲. $R \leftarrow (M \mod N)$

۳. اگر $R = 0$ آنگاه N را به عنوان جواب گزارش کن.
در غیر این صورت: $M \leftarrow N$ ، $N \leftarrow R$ و به قدم ۲ برگرد.

کد زیر یک نمونه کد پایتون است که الگوریتم اقلیدس را پیاده‌سازی کرده است.

```
def gcd(M, N):
   if (M < N):
      A = M
      M = N
      N = A
   while (N != 0):
      R = M % N
      M = N
      N = R
   return M
```