

مبانی مهندسی نرم افزار

فصل دوم

فرآیند

مقدمه

□ یک بنای ساختمانی چیست؟

□ یک بنای ساختمان محصولی از یک یا چند بنا است که شامل مصالح ساختمانی چون سیمان، گچ و.. است.

□ نرم افزار کامپیوتری چیست؟

□ نرم افزار محصول فرآیند مهندسی نرم افزار است که توسط مهندسان نرم افزار طراحی و ایجاد می شود.

□ مجموعه برنامه هایی که کنترل و هماهنگی فعالیت های سخت افزار و هدایت پردازش داده ها را برعهده دارد، نرم افزار نامیده می شود.

مقدمه

❑ نرم افزار عبارتست از:

۱. دستورات (برنامه های کامپیوتری) که در صورت اجرا شدن باعث انجام وظایف خواسته شده و کارایی مورد نظر می شوند.
۲. ساختمان داده هایی که امکان پردازش کافی اطلاعات را مهیا می سازند.
۳. مستنداتی که توصیف کننده عملکرد و چگونگی استفاده از برنامه ها می باشند. (یعنی دو مورد فوق را تشریح می کنند).

❑ نرم افزار کامپیوتری را چه کسانی استفاده می کنند؟

❑ هر فردی ممکن است به صورت مستقیم یا غیرمستقیم از آن استفاده کند.

❑ علت اهمیت نرم افزار؟

❑ تاثیر گسترده بر تمام جنبه های زندگی

❑ مراحل ایجاد نرم افزار چگونه است؟

❑ با بکارگیری فرآیندی خاص و طی کردن یک روند مشخص

نقش های دوگانه نرم افزار

- نقش اول: نرم افزار یک محصول است.
- دارای پتانسیل محاسباتی است.
- تولید، مدیریت، تغییر، نمایش و انتقال اطلاعات را انجام میدهد.
- نقش دوم: نرم افزار وسیله ای برای تحویل محصول است.
- از **functionality** سیستم پشتیبانی می کند یا به طور مستقیم آن را فراهم می کند.
- سایر برنامه ها را کنترل می کند (مثل سیستم عامل ها)
- بر روی ارتباطات تاثیر می گذارد (مثل نرم افزارهای شبکه)
- کمک به ساخت سایر برنامه ها می کند (مثل ابزارهای نرم افزاری)

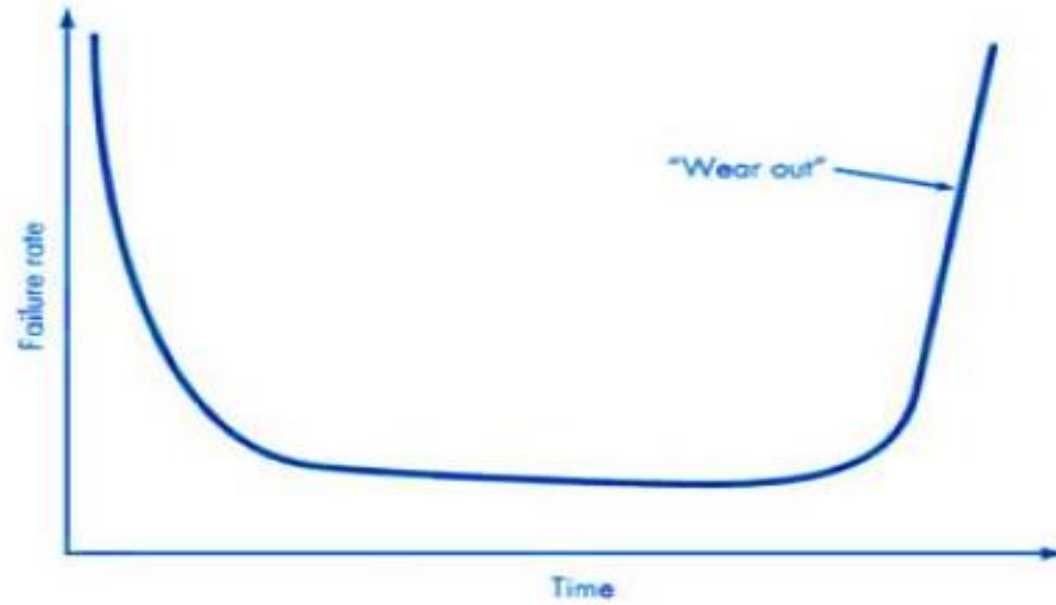
یک نرم افزار وقتی موفق است که:

- نیازهای کاربران را برطرف کند.
- بدون مشکل کار کند.
- اصلاح و تغییر بر روی آن ساده باشد.
- استفاده از آن آسان باشد.

خصوصیت نرم افزار

- ❑ نرم افزار یک عنصر **منطقی** است و نه یک جزء فیزیکی.
- ❑ بهترین راه شناخت مشخصات نرم افزار، **مقایسه آن با یک جزء دیگر** از محصولات ساخت انسان است (مثلا مقایسه با سخت افزار)
- ❑ بنابراین نرم افزار دارای مشخصاتی است که تا حد قابل ملاحظه ای با مشخصات سخت افزاری متفاوت است. تفاوت‌های عملیاتی:
- ❑ نرم افزار توسعه داده می شود (develop) در صورتیکه سخت افزار ساخته می شود (create). نرم افزار توسعه می یابد یا طراحی می شود؛ اما به مفهوم کلاسیک ساخته نمی شود.
- ❑ نرم افزار فرسوده نمی شود، اما کیفیت خود را از دست می دهد به عبارتی دیگر نرم افزار دور ریختنی نیست.
- ❑ گرچه صنعت به سمت مونتاژ اجزاء حرکت می کند، اما نرم افزار همچنان سفارشی ساخته می شود.
- ❑ روش ها و ابزارهای آزمایش نرم افزار از روش ها و ابزارهای سخت افزار متفاوت است.
- ❑ نرم افزار پیچیده است به این دلیل که بیشتر سفارشی ساخته می شود.

نمودار زوال پذیری (نرخ شکست) در سخت افزار



نمودار زوال پذیری (نرخ شکست) در سخت افزار

□ شکل موجود در اسلاید قبل، شکست سخت افزار نسبت به زمان را نشان می دهد. این رابطه، اغلب منحنی وانی شکل نامیده می شود و نشان می دهد که سخت افزار در ابتدا میزان شکست بالایی دارد (این شکست ها اغلب مربوط به اشکالات ساخت و طراحی هستند)، در یک بازه زمانی، اشکالات برطرف شده و میزان شکست در یک میزان ثابت می ماند. با گذشت زمان، قطعات سخت افزاری تحت تاثیر گرد و غبار، نوسان، استفاده نادرست، تغییرات زیاد درجه حرارت، و بسیاری عوامل دیگر کیفیت خود را از دست می دهند. به بیان ساده، سخت افزار به سمت از رده خارج شدن پیش می رود.

نمودار زوال پذیری (نرخ شکست) در نرم افزار



نمودار زوال پذیری (نرخ شکست) در نرم افزار

- نرم افزار به تدریج دچار فرسودگی نمی گردد، بلکه فاسد می شود.
- در طول دوران زندگی، نرم افزار نیازمند تغییرات (نگهداری) است. تغییرات انجام می شود، و اشکالات جدیدی ممکن است بوجود آید، که باعث می شود به میزان شکست ثابت خود بازگردد، تغییر دیگری درخواست می شود، و باعث می شود منحنی مجدد پرش داشته باشد. این مسئله باعث می شود سطح میزان شکست به آرامی افزایش یابد، و کیفیت نرم افزار به علت تغییرات تنزل می یابد.

کاربردهای نرم افزار

- محتوای اطلاعاتی و هدف برنامه ماهیت و کاربرد نرم افزار را مشخص می کند.
- نرم افزار سیستمی برای سرویس دهی به سایر برنامه ها نوشته می شود.
- نرم افزار کاربردی نیازهای خاص مثل نیازهای تجاری را رفع می کند.
- نرم افزارهای مهندسی / علمی مثل نرم افزارهای CAD
- نرم افزارهای جاسازی شده مثل محصولات هوشمند
- نرم افزارهای خط تولید
- نرم افزاری وب
- نرم افزارهای هوش مصنوعی

کاربردهای نرم افزار

- نرم افزار چه در یک تلفن سلولی قرار داشته باشد یا درون یک کامپیوتر بزرگ، یک **انتقال دهنده** اطلاعات محسوب می گردد. یعنی اطلاعاتی را **تولید** می کند، **سازمان** می دهد، **تحویل** می گیرد، **نمایش** یا **انتقال** می دهد.
- اگر از آن به عنوان وسیله **انتقال دهنده** محصول استفاده شود، به صورت پایه و اساس کنترل کامپیوتر (سیستم عامل)، تبادل اطلاعات (شبکه ها) و ایجاد و کنترل برنامه های دیگر (ابزارهای نرم افزاری و محیط آن) استفاده خواهد شد.



نمایش یا انتقال + تحویل + سازماندهی + تولید → انتقال دهنده

طبقه بندی نرم افزار

□ نرم افزار ها را می توان براساس ماهیت کاربردی آن ها مطابق زیر طبقه بندی کرد:

□ نرم افزار های سیستمی

□ نرم افزار های بلادرنگ

□ نرم افزار های تجاری

□ نرم افزار های مهندسی و علمی

□ نرم افزار های جاسازی شده

□ نرم افزار های کامپیوترهای شخصی

□ نرم افزار های مبتنی بروب

□ نرم افزار های هوش مصنوعی

باورهای غلط مشتریان

□ همه شما به یک دستورالعمل نیاز دارید که به شما بگوید چگونه برنامه نویسی را شروع کنید.

□ نیازمندی ها تغییر می کنند و نرم افزار می تواند خودش را با انعطافی که دارد با آن ها وفق دهد.

باورهای غلط توسعه دهندگان نرم افزار

- ❑ وقتی که برنامه ما اجرا شد کار تمام است.
- ❑ تا زمانی که برنامه در حال اجرا است من هیچ نظری در مورد چگونگی عملکرد آن ندارم.
- ❑ تنها چیز قابل تحویل برنامه ای است که کار می کند.
- ❑ مهندسی نرم افزار باعث ایجاد مستندات غیرضروری و حجیم می شود که نتیجه آن افت سرعت ما است.

باورهای غلط مدیریتی

- ❑ کتاب استانداردها برای ساخت نرم افزار نیازهای اطلاعاتی توسعه دهندگان را برطرف می کند.
- ❑ اگر از روال عادی کار عقب افتادیم می توانیم افراد بیشتری را جذب کنیم.
- ❑ می توانیم تولید نرم افزار را به شرکت دیگری بسپاریم و خودمان استراحت کنیم.