

۴

# معیارهای اندازهگیری و

# سنجش نرم افزار

درس مهندسی نرم افزار ۲

# مفاهیم کلیدی

- اندازهگیری
- معیار، شاخص و اندازه
- نقش معیارهای سنجش
- معیارهای فرآیند
- معیارهای پروژه
- ارزش تابعی
- معیارهای کیفیت نرم افزار
- فرآیند جمع آوری معیارها

# اندازه‌گیری

- اندازه‌گیری پایه مهندسی است و مهندسی نرم‌افزار نیز از این قاعده مستثنی نیست

## *Lord Kelvin* ■

- وقتی که بتوانید آن چه را که از آن سخن می‌گویید **اندازه‌گیری** کرده و آن را در  **قالب اعداد** بیان کنید، چیزی در مورد آن می‌دانید، ولی وقتی نتوانید آن را اندازه‌گیری کنید و نتوانید آن را در قالب اعداد بیان کنید، آگاهی شما ضعیف بوده و رضایت‌بخش نیست؛ در واقع ممکن است این امر نشاند هنده سرآغاز آگاهی شما از موضوع باشد، ولی در ذهن شما نقصان وجود دارد، تا به سطح دانش مورد نظر بررسید

در طول سال‌های گذشته جامعه مهندسی نرم‌افزار گفته *Kelvin* را مورد  
بحث قرار داده و شروع به پذیرش آن نمودند

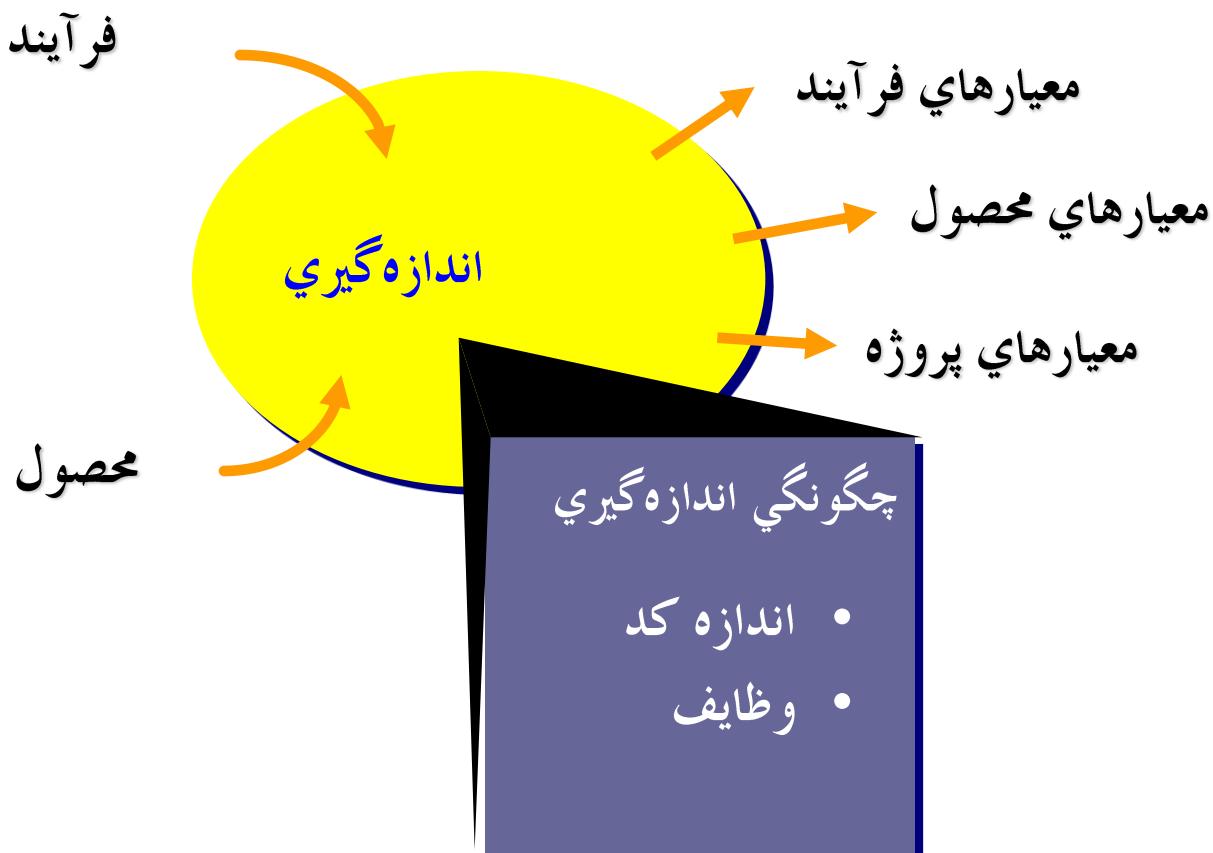
# اندازه‌گیری (ادامه)

## ■ چهار دلیل برای اندازه‌گیری

- **توصیف:** برای درک فرآیند، محصول، منابع و محیط
- **ارزیابی:** برای تعیین وضعیت مطابق با برنامه
- **پیش‌بینی:** ابتدا پیش‌بینی می‌کنیم و سپس برنامه‌ریزی می‌کنیم
- **بهبود:** کیفیت و کارایی فرآیند



# اندازه‌گیری (ادامه)



# مفاهیم اندازه‌گیری

## ■ اندازه (*Measure*)

- نمایش مقداری از میزان، ابعاد، ظرفیت ویژگی نرم‌افزار

## ■ اندازه‌گیری (*Measurement*)

- فعالیت تعیین اندازه

## ■ معیار (*Metric*)

- اندازه مقداری درجه یک سیستم، مولفه یا فرآیند

## ■ شاخص (*Indicator*)

- مجموعه‌ای از معیارهای سنجش برای تصحیح عملکرد پروژه

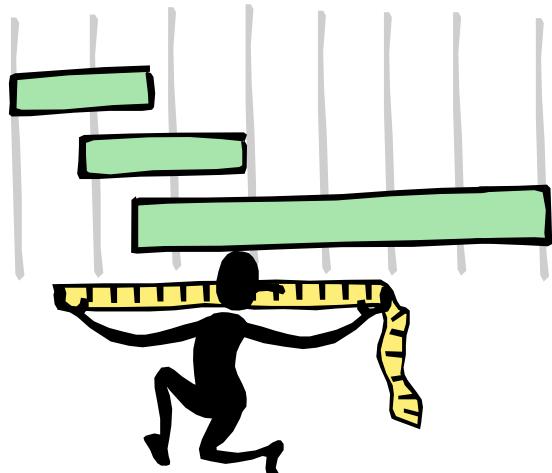
# نقش معیارهای سنجش

- اغلب معیارهای سنجش هم در حوزه پروژه و هم در فرآیند نرمافزار بکار می‌روند
- معیارهای سنجش فرآیند در طول پروژه‌های مختلف و در طی یک دوره زمانی طولانی گردآوری می‌شوند
- هدف از تهیه معیارهای سنجش فراهم آوردن **شاخصهایی** است که منجر به بهبود فرآیند نرمافزار در بلند مدت شود

# نقش معیارهای سنجش (ادامه)

## ■ شاخصهای فرآیند

- به سازمان توسعه‌دهنده نرم‌افزار کمک می‌کند تا درجه تاثیر فرآیند جاری سنجیده شود
- به مدیران و توسعه‌دهنگان کمک می‌کند تا بفهمند چه روشی اجرایی است و چه روشی اجرایی نیست



# نقش معیارهای سنجش (ادامه)

■ شاخصهای پروژه به مدیر پروژه کمک می‌کنند تا

- وضعیت یک پروژه در حال جریان را ارزیابی نماید
- ریسک‌های بالقوه را پیگیری کند
- زمینه‌های مشکل را پیش از آنکه «حاد» شوند، مشخص نماید
- جریان کار یا فعالیت‌ها را تعديل نماید
- قابلیت گروه پروژه را در مورد کنترل کیفیت محصولات کاری مهندسی نرم افزار مورد ارزیابی قرار دهد

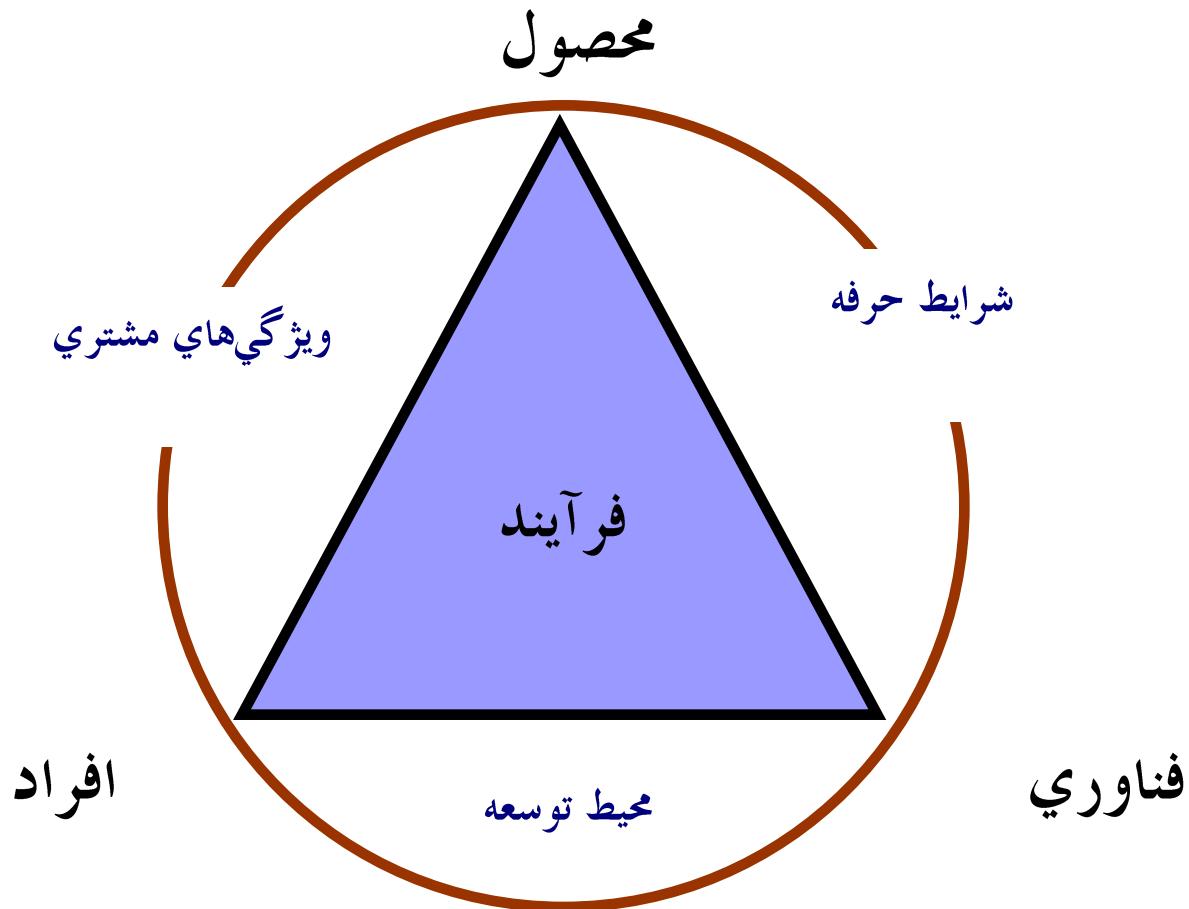
# معیارهای فرآیند

■ تنها راه بهبود هر فرآیند

- اندازهگیری ویژگی‌های خاص فرآیند
- تهیه مجموعه‌ای از معیارهای سنجش معنی‌دار مبتنی بر این صفات
- استفاده از معیارها جهت استخراج شاخص‌هایی که منجر به استقرار یک استراتژی برای بهبود فرآیند شود

■ در واقع، فرآیند تنها یکی از «عوامل قابل کنترل در بهبود کیفیت نرم‌افزار و کارایی سازمانی است»

# معیارهای فرآیند (ادامه)



دیگر معیارهای بخوبی کیفیت نرم افزار و کارایی سازمانی  
دست عزیرانه محبی در دری  
edu.mahdiegar@yahoo.com

# معیارهای فرآیند (ادامه)

- تاثیرگذاری فرآیند نرم افزار غیر مستقیم اندازه‌گیری می‌شود
  - مجموعه‌ای از معیارها، براساس خروجی‌هایی که می‌توان از فرآیند بدست آورده، ایجاد می‌شوند
  - نمونه‌ای از خروجی‌ها
    - اندازه‌گیری تعداد خطاهای قبلاً از تحويل نرم افزار به مشتری
    - نقص‌های گزارش شده توسط کاربر
    - محصولات کاری تحويل شده (بهره‌وری)
    - نیروی انسانی بکار گرفته شده
    - زمان صرف شده
    - مطابقت زمانبندی

معیارهای فرآیند می‌توانند با اندازه‌گیری خصوصیات برخی  
فعالیت‌های خاص مهندسی نرم‌افزار حاصل شوند

# معیارهای فرآیند (ادامه)

- دو نوع استفاده از داده‌های مختلف فرآیند (Grady)
  - معیارهای خصوصی (*Private Metrics*) برای یافتن شاخص‌های افزایش کارایی فرد
    - نرخ خرابی (توسط فرد)
    - نرخ خرابی (توسط واحد)
    - خطاهای پیدا شده در طول توسعه نرم‌افزار
  - معیارهای عمومی (*Public Metrics*) برای یافتن شاخص‌های افزایش کارایی گروه
    - خرابی یا نقص‌های گزارش شده در مورد توابع و عملکردهای نرم‌افزار
    - خطاهای مشاهده شده در حین بازبینی‌های فنی رسمی
    - تعداد خطوط برنامه

# معیارهای فرآیند (ادامه)

- تقسیم معیارها به صورت خصوصی و عمومی مشکل ایجاد می‌کند!
- Grady رسومی را برای استفاده از معیارهای سنجش نرم‌افزار پیشنهاد می‌کند که هم برای مدیران و هم توسعه‌دهندگان مناسب است
- در موقع تفسیر داده‌های سنجش توجه به حساسیت متدال و سازمانی داشته باشد
- هرگز از معیارهای سنجش برای تهدید افراد یا گروه‌ها استفاده نکنید
- هرگز اجازه ندهید یکی از معیارها جایگزین تمامی معیارهای دیگر گردد
- و ...

# معیارهای فرآیند (ادامه)

- هرچه سازمان بیشتر از معیارهای فرآیند استفاده نماید و شاخصهای ساده را ایجاد نماید، میتواند معیارهای دشوارتری به نام «بهدود آماری فرآیند نرمافزاری» را اجرا نماید

## *Statistical Software Process Improvement* •

- در واقع، SSPI از تحلیل شکستهای نرمافزار برای جمعآوری اطلاعات درباره خطاهای و نقصهای (*Defects*) در زمان استفاده یا توسعه یک برنامه کاربردی، سیستم یا محصول بهره میبرد

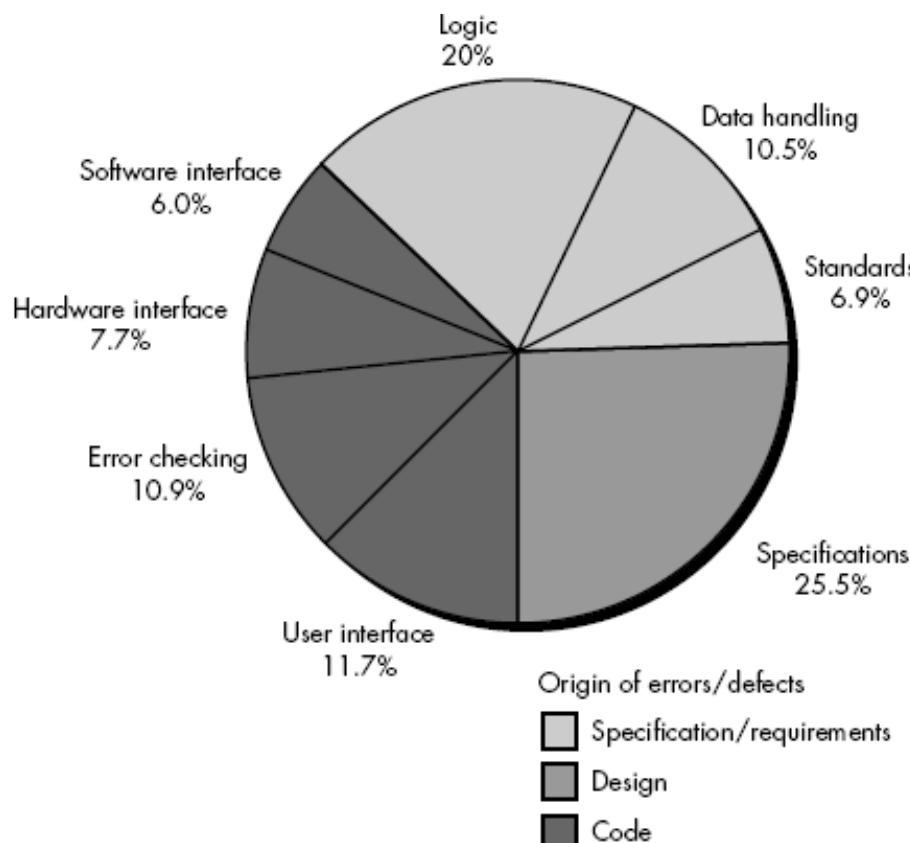
# معیارهای فرآیند (ادامه)

## ■ چگونگی تحلیل شکست نرم افزار

- ۱- تمام خطاها و نقص‌ها براساس منشا آنها دسته‌بندی می‌شوند (نظیر نقص در تعیین مشخصات، نقص در منطق و عدم پیروی از استانداردها)
- ۲- هزینه اصلاح هر خطا و نقص ثبت می‌گردد
- ۳- تعداد خطاها و نقص‌های موجود در هر دسته شمارش و به ترتیب نزولی مرتب می‌گردند
- ۴- کل هزینه خطاها و نقص‌های هر گروه محاسبه می‌گردد
- ۵- داده‌های حاصله تحلیل می‌شوند تا گروه دارای بیشترین هزینه برای سازمان تعیین شود
- ۶- برنامه‌ای برای اصلاح فرآیند، به منظور حذف (یا کاهش تواتر رخداد) آن دسته از خطاها و نقص‌ها که پر هزینه‌ترند، تدوین می‌گردد

# معیارهای فرآیند (ادامه)

■ با ادامه گام‌های اول و دوم می‌توان توزیع ساده‌ای از نوافص  
ایجاد نمود



# معیارهای سنجش پروژه

- در مقابل معیارهای سنجش فرآیند که برای مقاصد راهبردی بکار می‌رond، معیارهای سنجش پروژه فنی هستند
- برای تطبیق جریان کار پروژه و فعالیت‌های فنی بکار می‌رond
- کاربرد معیارهای سنجش پروژه
  - کاهش زمان توسعه با انجام تطبیق‌ها لازم برای کاهش تاخیرها و کاهش مشکلات و خطرات
  - ارزیابی کیفیت محصول براساس روند جاری فعالیت‌ها و تغییر فعالیت‌ها برای افزایش کیفیت

# معیارهای سنجش پروژه (ادامه)

## ■ مدلی برای اندازه‌گیری پروژه

- ورودی‌ها: اندازه‌گیری منابع مورد نیاز برای انجام کار
- خروجی‌ها: اندازه‌گیری تحویل دادنی‌ها یا محصولات کاری ایجاد شده در طول فرآیند مهندسی
- نتایج: اندازه‌گیری شاخص‌های موثر بر تحویل دادنی‌ها

# معیارهای سنجش پروژه (ادامه)

- در دنیای واقعی اندازه‌گیری به دو دسته عمده تقسیم می‌شود
  - اندازه‌گیری مستقیم
    - برای فرآیند دربرگیرنده سنجش هزینه و کار است
    - برای محصول
      - میزان خطوط برنامه
      - زمان اجرا
      - میزان حافظه
      - خطای گزارش شده در زمان معین
  - اندازه‌گیری غیرمستقیم
    - کیفیت نرم‌افزار
      - کارایی
      - قابلیت اصلاح و ...

# نرمال‌سازی معیارها



■ در صورتی که معیارها نرمال شود، می‌توان معیارهای نرم‌افزاری ایجاد نمود که قابل مقایسه برای سازمان‌های متفاوت هستند

- دو روش نرمال‌سازی *Function-Oriented* و *Size-Oriented*

# معیارهای سنجش مبتنی بر اندازه

- با نرمال‌سازی اندازه‌های مقداری کیفیت و یا بهره وری با استفاده از «اندازه» نرم‌افزار تولید شده، حاصل می‌شوند
- نمونه معیارهای سنجش مبتنی بر اندازه
  - میزان خطأ در هر KLOC (هزار خط کد)
  - نقص پیدا شده در KLOC
  - هزینه بهازای هر KLOC
  - میزان مستندات به ازای هر KLOC
  - خطاهای ازای هر شخص در ماه
  - و ....

اعتقاد بر این است که این معیارها، معیارهای وابسته به زبان برنامه‌نویسی هستند

# معیارهای سنجش مبتنی بر اندازه (ادامه)

Project	LOC	Effort	\$(000)	Pp. doc.	Errors	Defects	People
alpha	12,100	24	168	365	134	29	3
beta	27,200	62	440	1224	321	86	5
gamma	20,200	43	314	1050	256	64	6
•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•

# معیارهای سنجش مبتنی بر عملکرد

- وظیفه‌مندی ارائه شده توسط نرم‌افزار به عنوان مقدار نرمال‌سازی بکار می‌رود
- وظیفه‌مندی را نمی‌توان به صورت مستقیم ارزیابی نمود
  - باید به صورت غیرمستقیم از معیارهای مستقیم استخراج شود
- این روش اولین بار توسط *Albrecht* با معرفی «ارزش تابعی» (function point) پیشنهاد شد

# ارزش تابعی

- ارزش تابعی با استفاده از یک رابطه تجربی مبتنی بر معیارهای قابل اندازه‌گیری (مستقیم) دامنه اطلاعات نرم‌افزار و ارزیابی‌های مربوط به پیچیدگی نرم‌افزار حاصل می‌شود
- معیارهای قابل اندازه‌گیری
  - تعداد ورودی کاربران (Number of user inputs)
  - تعداد خروجی‌های کاربر (Number of user outputs)
  - تعداد درخواست‌های کاربران (Number of user inquiries)
  - تعداد فایل‌ها (Number of files)
  - تعداد واسطه‌های خارجی (Number of external interfaces)

# ارزش تابعی (ادامه)

## ■ تعداد ورودی کاربران

- هر **ورودی کاربر** که **داده‌هایی** با کاربرد متمایز را برای نرم‌افزار فراهم می‌آورد

## • تعداد خروجی‌های کاربر

- هر **خروجی کاربر** که **اطلاعاتی** کاربردی را برای **کاربر** فراهم می‌کند
- گزارشات، خروجی صفحه نمایش، پیام‌های خطأ و ... به عنوان خروجی محسوب می‌شوند
- اقلام داده‌ای مجزا که در یک گزارش قرار دارند، مجزا محاسبه نمی‌شوند

# ارزش تابعی (ادامه)

## ■ تعداد درخواست‌های کاربران

- درخواست به عنوان **ورودی لحظه‌ای** (on-line) تعریف می‌شود که حاصل آن **تولید پاسخ(هایی)** توسط نرم‌افزار به شکل خروجی لحظه‌ای است
- هر درخواست، مجزا شمارش می‌شود

## ■ تعداد فایل‌ها

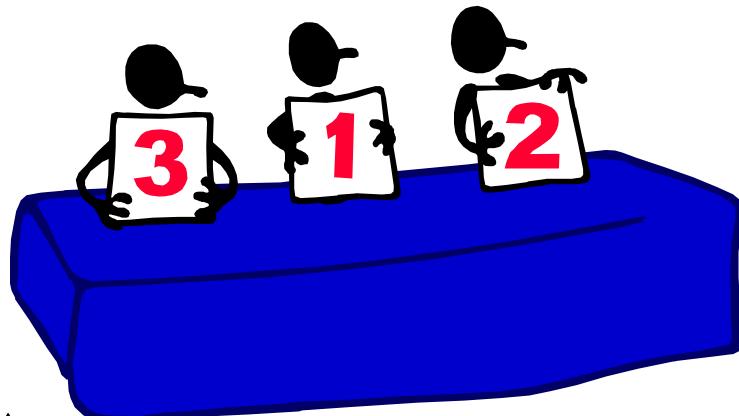
- هر فایل اصلی منطقی شمارش می‌شود
- فایل منطقی یک **گروه‌بندی منطقی** از داده‌هاست که می‌تواند **بخشی** از یک پایگاه داده بزرگ یا یک فایل باشد

# ارزش تابعی (ادامه)

## ■ تعداد واسطه‌ای خارجی

- تمام واسطه‌ای قابل خواندن توسط دستگاه (نظیر فایل‌های داده‌ای روی ابزار ذخیره‌سازی) که برای **ارسال اطلاعات** به یک سیستم دیگر مورد استفاده هستند، شمارش می‌شوند

- پس از شمارش این پارامترها، مقداری به عنوان پیچیدگی به هر یک منصب می‌گردد



# محاسبه ارزش تابعی

## فاکتورهای وزن دهنده

از ساده تا پیچیده

تعداد

معیارهای اندازه گیری

$$\boxed{\quad} = 6 \quad 4 \quad 3 \quad \times \quad \boxed{\quad}$$

تعداد ورودی های کاربران

$$\boxed{\quad} = 7 \quad 5 \quad 4 \quad \times \quad \boxed{\quad}$$

تعداد خروجی های کاربر

$$\boxed{\quad} = 6 \quad 4 \quad 3 \quad \times \quad \boxed{\quad}$$

تعداد پرس و جوهای کاربران

$$\boxed{\quad} = 15 \quad 10 \quad 7 \quad \times \quad \boxed{\quad}$$

تعداد فایل ها

$$\boxed{\quad} = 10 \quad 7 \quad 5 \quad \times \quad \boxed{\quad}$$

تعداد واسطه های خارجی



مجموع معیارها

# محاسبه ارزش تابعی (ادامه)

هر یک از پارامترهای ورودی را بشمارید

به هر شمارش یک مقدار وزنی نسبت دهید و «مجموع معیارها» را محاسبه کنید

مقادیر فاکتورهای خارجی  $Fi$  را محاسبه کنید

ارزش تابعی برابر است با

$$FP = C \times [0.65 + 0.01 \times \sum_{i=1}^{14} (F_i)]$$

# محاسبه ارزش تابعی (ادامه)

- مقادیر  $F_i$ ‌ها با پاسخ (مقدار ۰ - ۵) به سوالات ذیل به دست می‌آید
- ۱- آیا سیستم نیاز به داشتن نسخه پشتیبان و بازیابی دارد؟
- ۲- آیا ارتباطات داده‌ای مورد نیاز است؟
- ۳- آیا توابع پردازشی توزیع شده وجود دارد؟
- ۴- آیا کارایی عامل حساسی است؟
- ۵- آیا سیستم در محیط عملیاتی موجود و دشوار اجرا خواهد شد؟
- ۶- آیا سیستم نیاز به داده ورودی لحظه‌ای دارد؟
- ۷- آیا داده ورودی لحظه‌ای نیاز دارد تا تراکنش ورودی بر روی صفات چندگانه یا توسط عملیات چندگانه ساخته شود؟

# محاسبه ارزش تابعی (ادامه)

## ■ مقادیر (ادامه)

- ٨ - آیا فایل اصلی لحظه‌ای بهنگام می‌شود؟
- ٩ - آیا ورودی‌ها، خروجی‌ها، فایل‌ها و درخواست‌ها پیچیده هستند؟
- ١٠ - آیا پردازش داخلی پیچیده است؟
- ١١ - آیا کد بگونه‌ای طراحی می‌شود که قابل استفاده مجدد باشد؟
- ١٢ - آیا تغییرات و نصب در طراحی در نظر گرفته شده است؟
- ١٣ - آیا سیستم طوری طراحی شده است که بتوان آن را چند بار در سازمان‌های متفاوت نصب کرد؟
- ٤ - آیا نرم افزار طوری طراحی شده است که امکان تغییرات را بدهد و کاربر به سادگی از آن استفاده نماید؟

# نرمال‌سازی ارزش تابعی

■ پس از محاسبه مقدار ارزش تابعی، همانند روش LOC نرمال می‌شود

- نسبت خطاهای بر FP
- نسبت خرابی بر FP
- نسبت هزینه بر FP
- نسبت تعداد صفحات مستندات بر FP
- نسبت FP بر نفر - ماه



# دلیل انتخاب ارزش تابعی

- عدم وابستگی به زبان برنامه‌نویسی
- از معیارهای ساده استفاده می‌کند که می‌تواند قبل از انجام فرآیند توسعه نرم‌افزار تعیین شوند
- پیاده‌سازی‌های ابتکاری که سبب کاهش LOC می‌شوند منجر بر جریمه نمی‌شوند
- ارزیابی میزان تاثیر مولفه‌های قابل استفاده مجدد را ممکن می‌سازد

# معیارهای کیفیت نرم افزار

## ■ عوامل موثر بر کیفیت

- عملیات محصول (استفاده از آن)
- تغییرات محصول (تغییر آن)
- انتقال محصول (اصلاح آن برای کار در محیطی متفاوت)



# معیارهای کیفیت نرم افزار (ادامه)

## ■ صحت (*Correctness*)

- برنامه باید صحیح عمل کند و گرنه برای کاربر ارزشی نخواهد داشت
- درستی عملکرد درجه‌ای است که در آن نرم افزار عملکرد مورد نیاز خود را بروز می‌دهد
- از معمول‌ترین معیارهایی درستی عملکرد می‌توان به تعداد خرابی در هر KLOC اشاره نمود
- خرابی (شکست) بعنوان عدم تطابق مشاهده شده با نیازها تعریف می‌گردد

# معیارهای کیفیت نرم افزار (ادامه)

## ■ قابلیت نگهداری (*Maintainability*)

- نگهداری امکانی است که توسط آن در صورت بروز خطا می‌توان یک برنامه را اصلاح نمود و یا در صورت تغییر محیط آن را با محیط تطبیق داد، و یا در صورتی که مشتری خواهان تغییری در نیازهای نرم افزار باشد امکاناتی به آن اضافه کرد

## ■ جامعیت (*Integrity*)

- قابلیت یک سیستم را در مقایله با دستکاری‌های تصادفی و یا عمدی در راستای ایمنی آن اندازه گیری می‌کند

# معیارهای کیفیت نرم افزار (ادامه)

## ■ قابلیت استفاده (Usability)

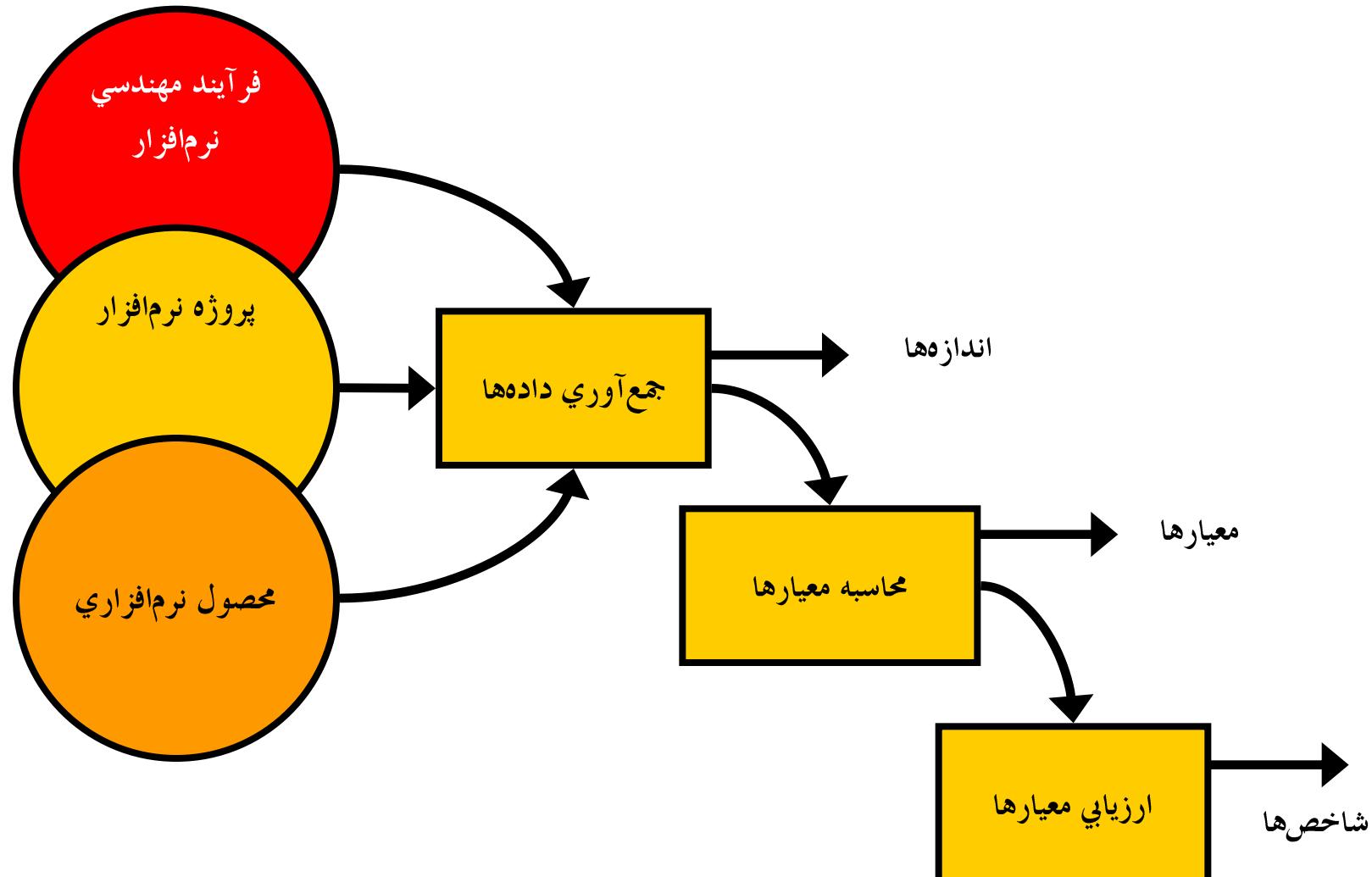
- برنامه باید *User Friendly* باشد
- از طریق ۴ ویژگی این کار اندازه گیری می گردد
  - ۱) مهارت فیزیکی یا ذهنی مورد نیاز برای یادگیری نرم افزار
  - ۲) مدت زمان مورد نیاز برای کسب کارایی بالا در استفاده از سیستم
  - ۳) خالص افزایش بهره وری در مقایسه با سیستم قبلی
  - ۴) ارزیابی شهودی (گاهی این معیار از طریق پرسشنامه از نظرسنجی کاربران نسبت به سیستم بدست می آید

# کارایی رفع خرابی (Defect Removal Efficiency)

- یکی از معیارهای کیفی که در هر دو سطح پروژه و فرآیند سودمند است
- معیار قابلیت پالایش (فیلتر کردن) فعالیت‌های تضمین کیفیت و کنترل می‌باشد

$$DRE = (\text{Errors}) / (\text{Errors} + \text{Defects})$$

# فرآیند جمع‌آوری معیارها



# پایان