فصل اول

- **Software & Software Engineering**

Slide Set to accompany

Software Engineering: A Practitioner's Approach, 7/e

by Roger S. Pressman

Slides copyright © 1996, 2001, 2005, 2009 by Roger S. Pressman

For non-profit educational use only

May be reproduced ONLY for student use at the university level when used in conjunction with *Software Engineering: A Practitioner's Approach, 7/e.* Any other reproduction or use is prohibited without the express written permission of the author.

All copyright information MUST appear if these slides are posted on a website for student use.

معرفی نرم افزار

(۱)دستورات (برنامه های کامپیوتری) که در صورت اجرا شدن باعث انجام عمل و کارآیی خواسته شده می شوند،(۲) ساختمان داده ها یی که باعث می شوند برنامه ها به طور مناسبی اطلاعات را دستکاری کنند ، و (۳) مستنداتی که توصیف کننده عملکرد و نحوه استفاده از برنامه ها هستند .

These slides are designed to accompany *Software Engineering: A Practitioner's Approach, 7/e* (McGraw-Hill 2009). Slides copyright 2009 by Roger Pressman.

نقش دوگانه نرم افزار

- نرم افزار یک محصول است.
  - تبدیل کننده اطلاعات – تولید ، مدیریت ، دریافت ، تغییر یا ارسال اطلاعات
  - ارائه پتانسیل محاسباتی سخت افزار و شبکه ها
- نرم افزار وسیله ای برای تحویل یک محصول (اطلاعات)می باشد.
  - پشتیبان یا تامین کننده قابلیت های سیستم
  - کنترل کننده برنامه های دیگر (سیستم عامل)
  - موثر بر ارتباطات (نرم افزار شبکه)
  - پشتیبانی از ایجاد نرم افزارهای دیگر (ابزارهای نرم افزار)

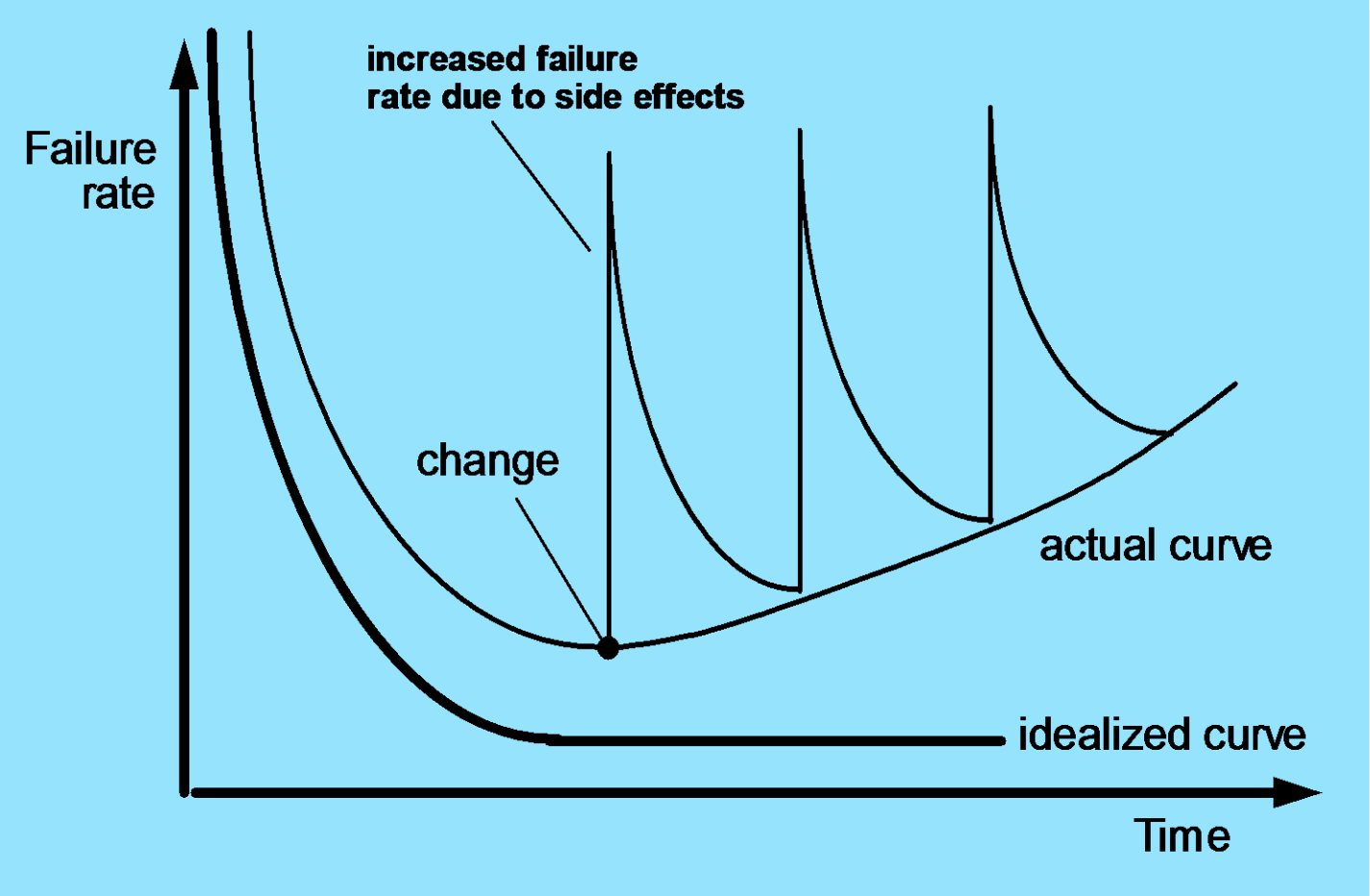
These slides are designed to accompany *Software Engineering: A Practitioner's Approach, 7/e* (McGraw-Hill 2009). Slides copyright 2009 by Roger Pressman.

معرفی نرم افزار

- نرم افزار توسعه داده می شود یا مهندسی می شود و به شکل معمول ساخته نمی شود .
- نرم افزار فرسوده نمی شود .
- اگرچه صنعت به سمت نصب قطعات پیش ساخته می رود ، اکثر نرم افزار ها به شکل متداول ایجاد می شوند .

These slides are designed to accompany *Software Engineering: A Practitioner's Approach, 7/e* (McGraw-Hill 2009). Slides copyright 2009 by Roger Pressman.

شکست نرم افزار



These slides are designed to accompany *Software Engineering: A Practitioner's Approach, 7/e* (McGraw-Hill 2009). Slides copyright 2009 by Roger Pressman.

کاربرد های نرم افزار

- نرم افزار سیستمی
- نرم افزار کاربردی
- نرم افزار جاسازی شده
- نرم افزار نرم علمی/ مهندسی
- نرم افزار خط تولید
- برنامه های کاربردی وب
- نرم افزار هوش مصنوعی

These slides are designed to accompany *Software Engineering: A Practitioner's Approach, 7/e* (McGraw-Hill 2009). Slides copyright 2009 by Roger Pressman.

نرم افزار - طبقه بندی های جدید

- Open world computing—pervasive, distributed computing
- Ubiquitous computing—wireless networks
- Netsourcing—the Web as a computing engine
- Open source—”free” source code open to the computing community (a blessing, but also a potential curse!)
- Also … (see Chapter 31)
  - Data mining
  - Grid computing
  - Cognitive machines
  - Software for nanotechnologies

These slides are designed to accompany *Software Engineering: A Practitioner’s Approach, 7/e* (McGraw-Hill 2009). Slides copyright 2009 by Roger Pressman.

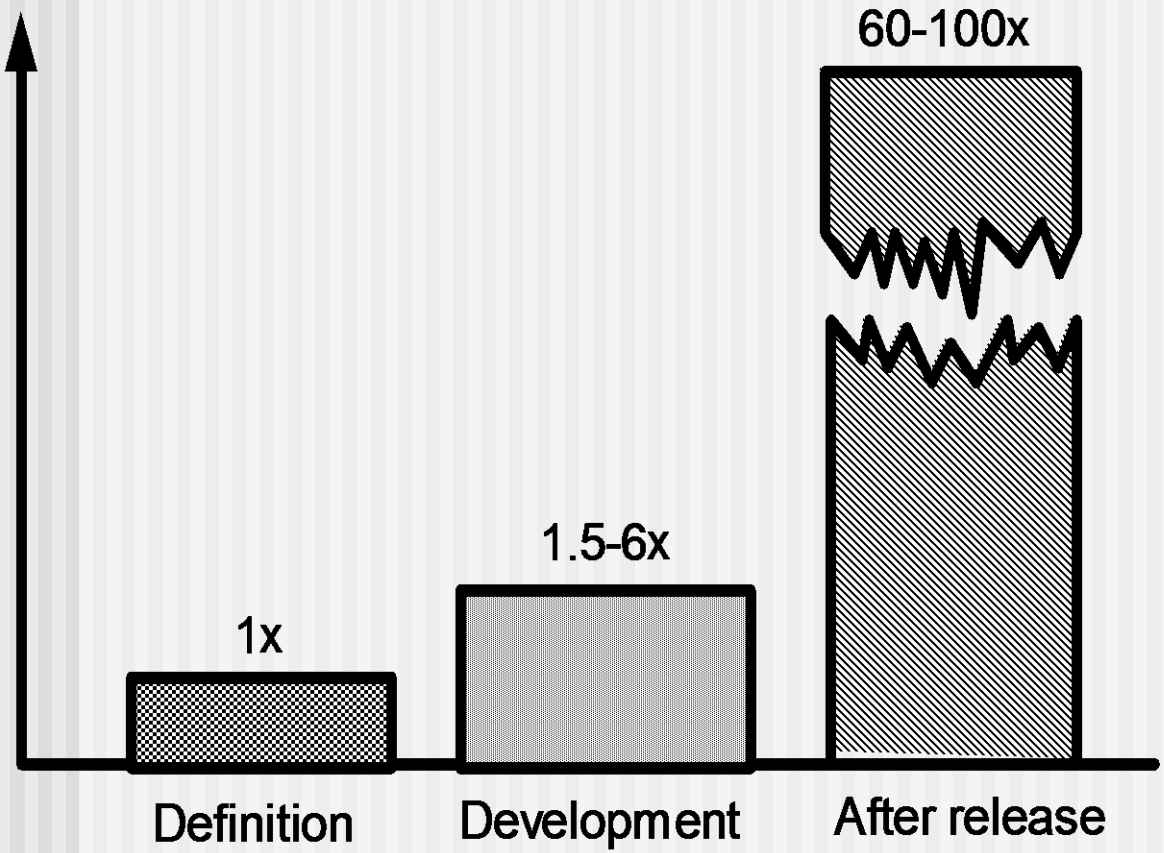
میراث نرم افزار

چرا باید دچار تغییر شود؟

- نرم افزار باید سازگار شود تا نیازهای محیط محاسباتی و تکنولوژی جدید را تامین کند .
- نرم افزار باید ارتقا یابد تا نیازهای شغلی جدید را پیاده سازی کند.
- نرم افزار باید توسعه یابد تا بتواند با سیستم ها یا پایگاه داده های مدرن دیگر ارتباط داشته باشد .
- معماری نرم افزار باید دوباره ایجاد شود تا در محیط شبکه قابل بکارگیری باشد.

These slides are designed to accompany *Software Engineering: A Practitioner's Approach, 7/e* (McGraw-Hill 2009). Slides copyright 2009 by Roger Pressman.

هزینه تغییرات



ویژگی های کاربرد های تحت وب

- توان شبکه ای. یک کاربرد تحت وب روی شبکه قرار می گیرد و باید نیازهای مجموعه ای از سرویس گیرندگان را فراهم نماید .
- همروندی. تعداد زیادی از کاربران ممکن است در یک زمان به یک کاربرد تحت وب دسترسی یابند .
- بار غیر قابل پیش بینی. تعداد کاربران کاربرد های تحت وب از یک روز به روز دیگر ممکن است تغییر کنند .
- کارآیی. اگر یک کاربر کاربرد تحت وب خیلی منتظر بماند (برای دسترسی به پردازش سمت سرویس دهنده ، برای قالب بندی و نمایش سمت سرویس گیرنده)ممکن است تصمیم بگیرد به جای دیگری برود .
- در دسترس بودن. اگرچه انتظار صد در صدی برای در دسترس بودن غیر معمول است ، کاربران کاربرد های مشهور تحت وب اغلب تقاضای دسترسی بر مبنای "24/7/365" را دارند .

These slides are designed to accompany *Software Engineering: A Practitioner's Approach, 7/e* (McGraw-Hill 2009). Slides copyright 2009 by Roger Pressman.

ویژگی های کاربرد های تحت وب (ادامه)

- داده محوری. عملکرد اصلی بسیاری از کاربرد های تحت وب ، استفاده از امکانات ابررسانه ای برای نمایش متن ، گرافیک ، صوت ، و ویدیو برای کاربر می باشد.
- حساسیت به محتوا. کیفیت و ظاهر محتوا از ویژگی های مهم کیفیت در هر کاربرد تحت وب هستند .
- تکامل مداوم. بر خلاف نرم افزار های کاربردی متداول که با یک سری نسخه های برنامه ریزی شده در طول زمان تکامل می یابند ، کاربرد های تحت وب به طور مداوم تکامل داده می شوند .
- بلافاصله بودن. یک نیاز اصلی برای نرم افزاری که سریع به بازار داده می شود ، ویژگی زمان بازار است که کاربرد های تحت وب اغلب زمان بازار در حد چند روز یا هفته دارند.
- حفاظت. چون کاربرد های تحت وب از طریق شبکه قابل دسترس هستند ، محدود نمودن دسترسی مجموعه ای از کاربران اگر غیر ممکن نباشد بسیار سخت خواهد بود .
- نمای ظاهری. یک بخش غیر قابل انکار یک کاربرد تحت وب ظاهر و حس آن می باشد .

These slides are designed to accompany *Software Engineering: A Practitioner's Approach, 7/e* (McGraw-Hill 2009). Slides copyright 2009 by Roger Pressman.

مهندسی نرم افزار

- چند واقعیت ساده
  - *فعالیت زیادی باید انجام شود تا مسئله مورد نظر قبل از توسعه نرم افزار درک شود*
  - *طراحی یک فعالیت طاقت فرسا می باشد .*
  - *نرم افزار باید کیفیت بالایی داشته باشد.*
  - *نرم افزار باید قابل نگهداری باشد .*
- یک تعریف
  - *مهندسی نرم افزار ، ایجاد و استفاده از اصول مهندسی صحیح به منظور دستیابی به نرم افزاری اقتصادی است که قابل اطمینان باشد و بر روی ماشین های واقعی بطور کارآمد اجرا شود .*

These slides are designed to accompany *Software Engineering: A Practitioner's Approach, 7/e* (McGraw-Hill 2009). Slides copyright 2009 by Roger Pressman.

مهندسی نرم افزار

- تعریف IEEE
  - مهندسی نرم افزار (۱) بکارگیری روشی سیستماتیک ، اصولی ، و قابل بررسی به منظور توسعه ، اجرا ، و پشتیبانی نرم افزار می باشد . که آن یعنی بکارگیری مهندسی در نرم افزار. (۲) مطالعه ی شیوه های مختلف بخش (۱).

These slides are designed to accompany *Software Engineering: A Practitioner's Approach, 7/e* (McGraw-Hill 2009). Slides copyright 2009 by Roger Pressman.

یک تکنولوژی لایه ای

مهندسی نرم افزار

These slides are designed to accompany *Software Engineering: A Practitioner's Approach, 7/e* (McGraw-Hill 2009). Slides copyright 2009 by Roger Pressman.

چارچوب فرآیند

چاچوب فرآیند

فعالیت های چارچوب

وظایف کاری

محصولات کاری

مراحل و موارد قابل تحویل

نقاط بازبینی کیفیت

فعالیت های چتری

فعالیت های چارچوب

- Communication
- Planning
- Modeling
  - Analysis of requirements
  - Design
- Construction
  - Code generation
  - Testing
- Deployment

These slides are designed to accompany *Software Engineering: A Practitioner's Approach, 7/e* (McGraw-Hill 2009). Slides copyright 2009 by Roger Pressman.

فعالیت های چتری

- کنترل و ردیابی پروژه نرم افزار (باعث میشود تا تیم به ارزیابی میزان پیشرفت و اقدامات اصلاحی برای حفظ برنامه زمانبندی دست یابد)
- مدیریت ریسک (ارزیابی ریسکهایی که ممکن است بر کیفیت پروژه و یا نتایج تاثیر گذار باشند)
- تضمین کیفیت نرم افزار (فعالیت های مورد نیاز برای حفظ کیفیت نرم افزار)
- بررسی فنی رسمی (ارزیابی محصولات کاری مهندسی برای شناسایی و حذف خطاها قبل از انتشار آنها به فعالیت بعدی)
- اندازه گیری (تعریف و جمع آوری معیارهای فرایند ، پروژه و محصول برای کمک به تیم نرم افزار در ارائه نرم افزاری مطابق با نیازهای مشتری)
- مدیریت پیکربندی نرم افزار (مدیریت اثرات تغییرات)
- مدیریت قابلیت استفاده مجدد (تعریف معیارها برای استفاده مجدد از محصول کاری و ایجاد مکانیسمی برای رسیدن به استفاده مجدد از مولفه ها)
- آماده سازی و تولید محصول کاری (فعالیت هایی برای ایجاد مدل ها ، اسناد ، گزارشات ، فرم ها ، لیست ها ، و غیره)

These slides are designed to accompany *Software Engineering: A Practitioner's Approach, 7/e* (McGraw-Hill 2009). Slides copyright 2009 by Roger Pressman.

قابلیت تطبیق مدل فرآیند

- فعالیت های چارچوب قابل اعمال برای هر پروژه ای می باشند . . . ولی
- وظایف (و درجه دقت) هر فعالیت بر اساس ویژگی هایی مانند :
  - نوع پروژه
  - ویژگی های پروژه
  - توافق موجود در تیم پروژه
  - و . . .

متفاوت خواهد بود :

ضرورت تجربه

- Polya ضرورت حل مسئله و در نتیجه ضرورت تجربه مهندسی نرم افزار را چنین بیان نموده است :

۱. *درک مسئله (ارتباط و تحلیل)*
۲. *طرح یک راه حل(مدل سازی و طراحی نرم افزار)*
۳. *انجام طرح (تولید کد)*
۴. *آزمایش نتیجه به منظور دقت(آزمایش و تضمین کیفیت)*

These slides are designed to accompany *Software Engineering: A Practitioner's Approach, 7/e* (McGraw-Hill 2009). Slides copyright 2009 by Roger Pressman.

درک مسئله

- چه کسی در رسیدن به راه حل ذینفع است ؟ ذینفعان چه کسانی هستند ؟
- مجهول ها چه هستند ؟ چه داده هایی ، چه اعمالی و چه ویژگی هایی برای حل مسئله لازم هستند ؟
- مسئله قابل تجزیه است ؟ آیا امکان شکستن مسئله به مسائل کوچکتر که ساده تر و قابل حل باشند وجود دارد ؟
- آیا مسئله می تواند به صورت گرافیکی نمایش داده شود ؟ می توان یک مدل تحلیلی ایجاد کرد ؟

These slides are designed to accompany *Software Engineering: A Practitioner's Approach, 7/e* (McGraw-Hill 2009). Slides copyright 2009 by Roger Pressman.

طرح یک راه حل

- آیا مسائل مشابهی را قبلا دیده اید ؟ آیا الگوهایی وجود دارند که در مسئله شناخته شده باشند؟آیا نرم افزاری وجود دارد که داده ها ،عملکرد ها و ویژگیهای مورد نیاز را پیاده سازی کرده باشد ؟
- آیا الگوی مشابهی قبلا حل شده است ؟ در اینصورت ، آیا اجزای آن قابل استفاده مجدد هستند ؟
- آیا زیر مسائل قابل تعریف هستند ؟ در اینصورت ، آیا راه حلی برای زیر مسائل وجود دارد ؟
- آیا راه حل را می توانید به شکلی ارائه دهید که منجر به پیاده سازی موثری شود ؟ آیا یک مدل طراحی قابل ایجاد است ؟

These slides are designed to accompany *Software Engineering: A Practitioner's Approach, 7/e* (McGraw-Hill 2009). Slides copyright 2009 by Roger Pressman.

دنبال نمودن طرح

- آیا این راه حل منطبق بر طرح مسئله می باشد ؟ آیا کد منبع قابل ردیابی به مدل طراحی می باشد ؟
- آیا هریک از اجزای راه حل درست هستند ؟ آیا طراحی و کد مرور شده اند یا اثبات صحت الگوریتم انجام شده است ؟

These slides are designed to accompany *Software Engineering: A Practitioner's Approach, 7/e* (McGraw-Hill 2009). Slides copyright 2009 by Roger Pressman.

بررسی نتیجه

- آیا امکان آزمایش هر جزء از راه حل وجود د دارد؟ آیا استراتژی آزمایش مناسبی اجرا شده است ؟
- آیا راه حل ارائه شده نتایجی تولید می کند که با داده ها، عملکردها ، و ویژگی های مورد نیاز مطابقت داشته باشد ؟ آیا نرم افزار با نیازهای تمام ذی نفعان اعتبار سنجی شده است؟

These slides are designed to accompany *Software Engineering: A Practitioner's Approach, 7/e* (McGraw-Hill 2009). Slides copyright 2009 by Roger Pressman.

اصول کلی Hooker

- دلیل وجود تمام این ها
- ساده نگری
- حفظ دیدگاه
- آنچه را شما تولید می کنید ، دیگران استفاده می کنند.
- دیدگاهی باز به آینده داشته باشید .
- از قبل طرحی برای استفاده مجدد داشته باشید .
- فکر کنید !

These slides are designed to accompany *Software Engineering: A Practitioner's Approach, 7/e* (McGraw-Hill 2009). Slides copyright 2009 by Roger Pressman.

داستان های نرم افزار

- هنوز هم بسیاری از مدیران و دست اندرکاران به آنها معتقد هستند .
- گاهی اوقات عناصری از حقیقت را دارا می باشند .
- مدیر و هر فرد درگیر در پروژه باید واقعیت کسب و کار مربوط به نرم افزار را درک کنند .

سوالات کلیدی

- What is the difference between software engineering and computer science?
- What is the difference between software engineering and system engineering?
- What are the attributes of good software?
- What are the key challenges facing software engineering?
- Professional and ethical responsibility

همه آنها چگونه شروع می شوند؟

- *SafeHome:*
  - Every software project is precipitated by some business need—
    - the need to correct a defect in an existing application;
    - the need to the need to adapt a ‘legacy system’ to a changing business environment;
    - the need to extend the functions and features of an existing application, or
    - the need to create a new product, service, or system.