



SDLC

تحلیل و طراحی سیستمها
دکتر قاسمی
دانشگاه آزاد تهران شمال

چرخه حیات توسعه سیستم ها (SDLC).

❖ چرخه عمر توسعه سیستم ها (SDLC).

- چارچوب کلی برای مدیریت فرآیند توسعه سیستم ها را فراهم می کند

❖ دو رویکرد اصلی برای SDLC

- رویکرد پیش بینی - فرض می کند که پروژه می تواند از قبل برنامه ریزی شود
- رویکرد تطبیقی - انعطاف پذیرتر، فرض می کند که پروژه نمی تواند از قبل برنامه ریزی شود

❖ همه پروژه ها از برخی تغییرات SDLC استفاده می کنند

انتخاب رویکرد پیش‌بینی‌کننده در مقابل انطباق با SDLC

The choice of SDLC varies depending on the project

Predictive
SDLC

Adaptive
SDLC

Requirements well understood
and well defined.
Low technical risk.

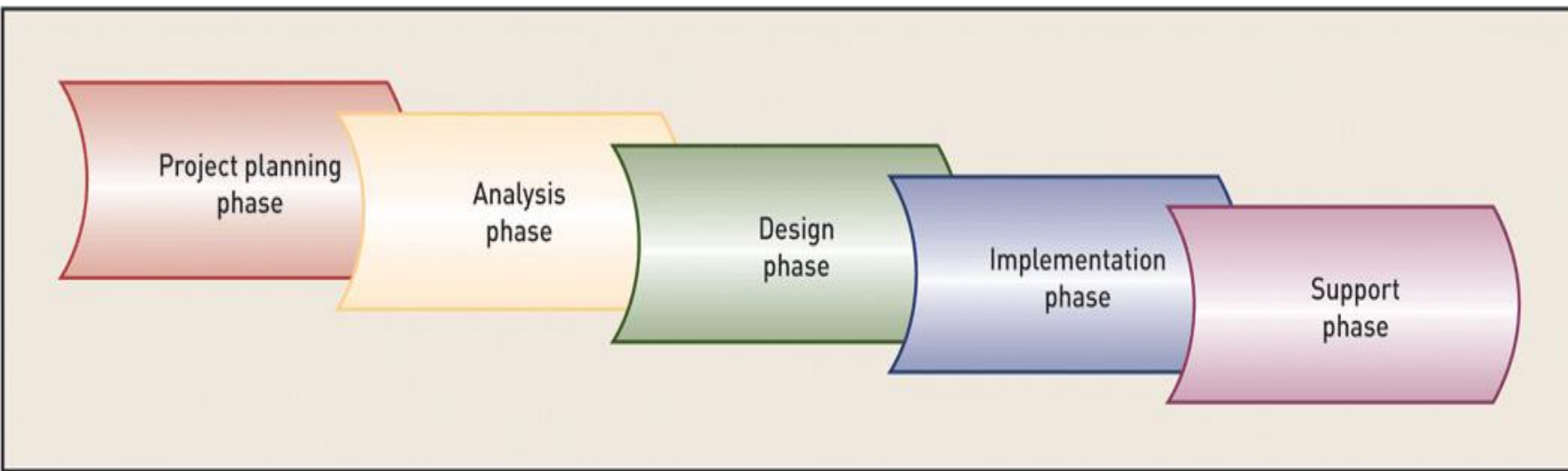
Requirements and needs
uncertain.
High technical risk.

رویکرد پیشگویانه سنتی به SDLC

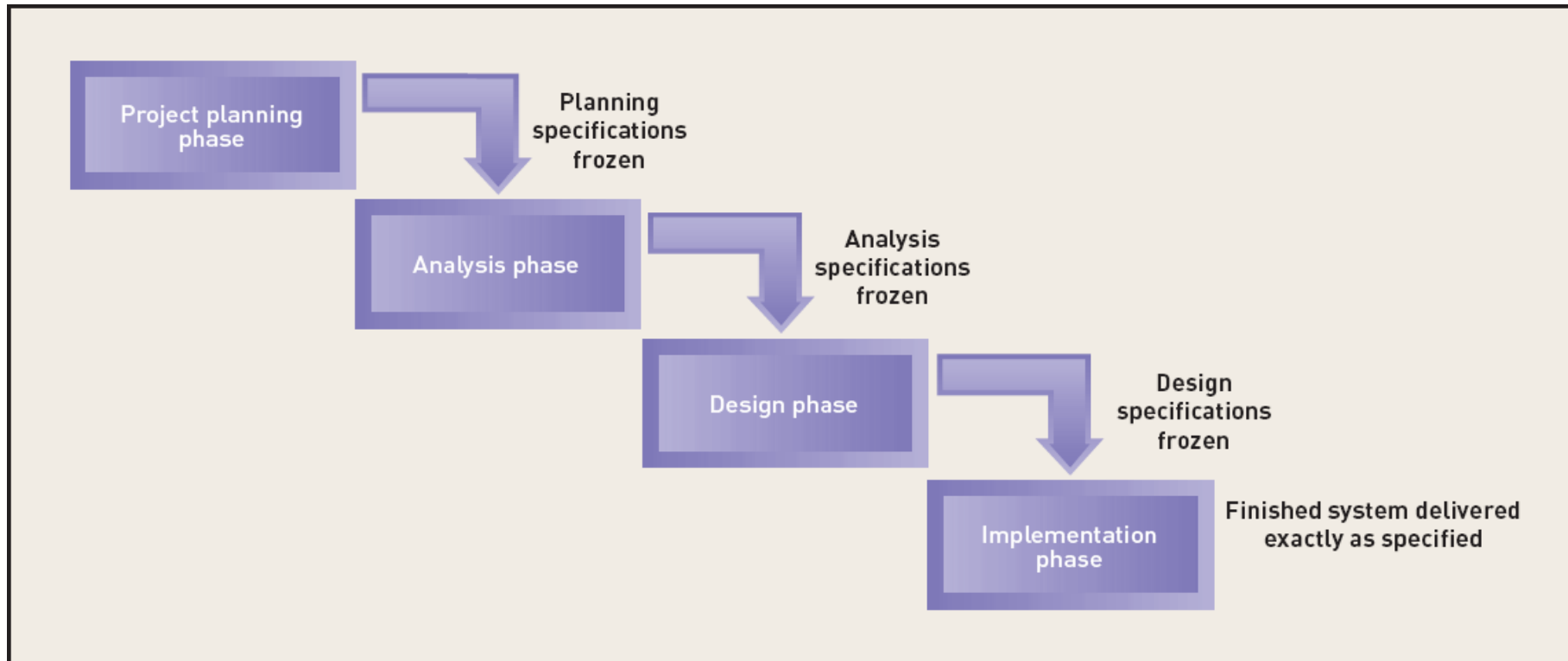
- ❖ برنامه ریزی پروژه - شروع، اطمینان از امکان سنجی، برنامه ریزی برنامه، اخذ تاییدیه برای پروژه
- ❖ تجزیه و تحلیل - درک نیازهای تجاری و الزامات پردازش
- ❖ طراحی - تعریف سیستم راه حل بر اساس الزامات و تصمیمات تجزیه و تحلیل
- ❖ پیاده سازی - ساخت، آزمایش، آموزش کاربران و نصب سیستم جدید
- ❖ پشتیبانی - سیستم را در حال اجرا نگه دارید و بهبود دهید

SDLC و حل مسئله

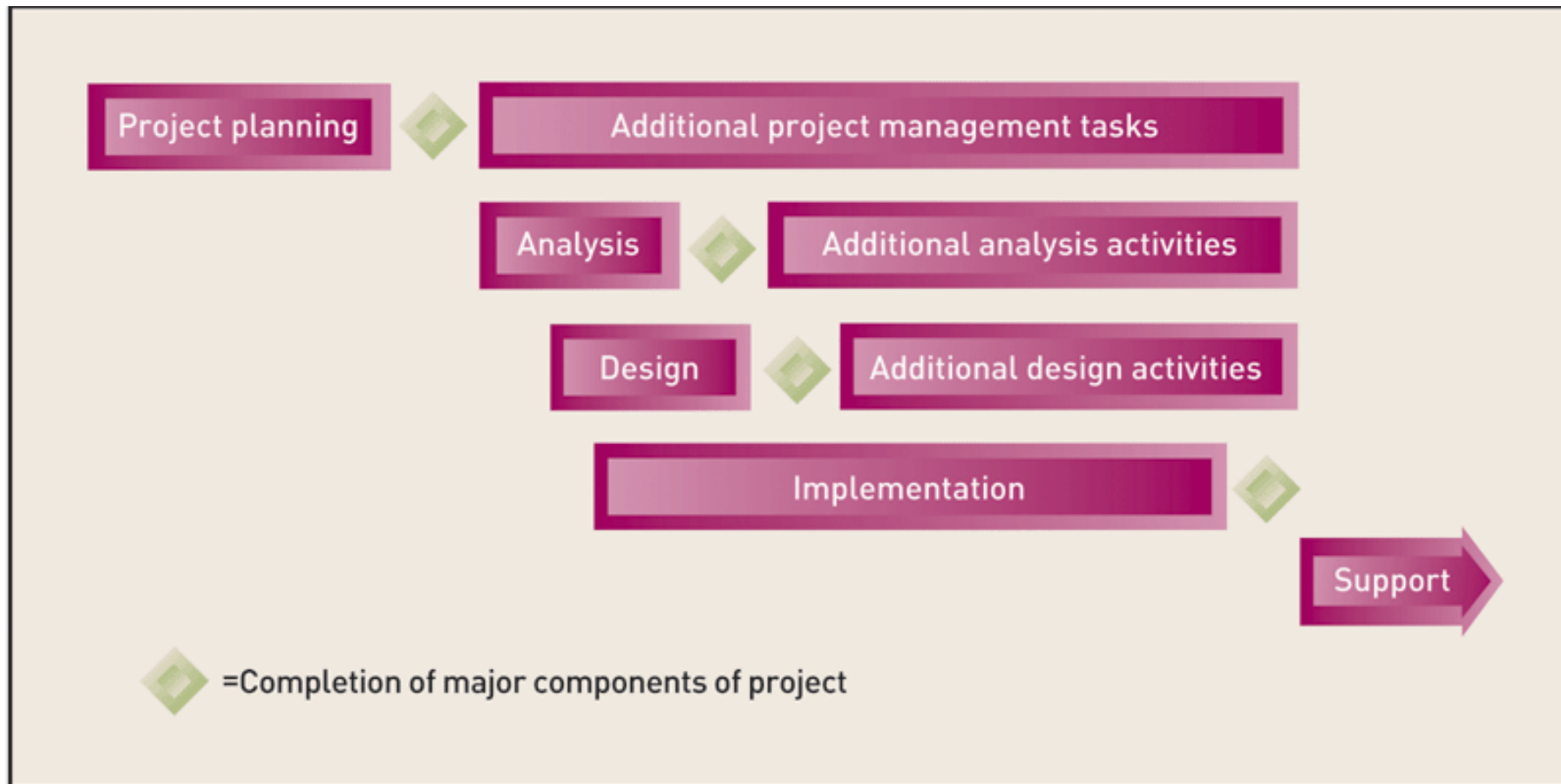
- سازمان مشکل را تشخیص می دهد (برنامه ریزی پروژه).
- تیم پروژه بررسی می کند، نیازمندی های مسئله و راه حل را درک می کند (تحلیل).
- راه حل با جزئیات مشخص شده است (طراحی).
- سیستمی که مشکل را حل می کند ساخته و نصب می شود (پیاده سازی).
- سیستم مورد استفاده، نگهداری و بهبود (پشتیبانی)



رویکرد "آبشار" SDLC



رویکرد آبشار اصلاح شده با فازهای همپوشانی



رویکردهای تطبیقی جدیدتر به SDLC

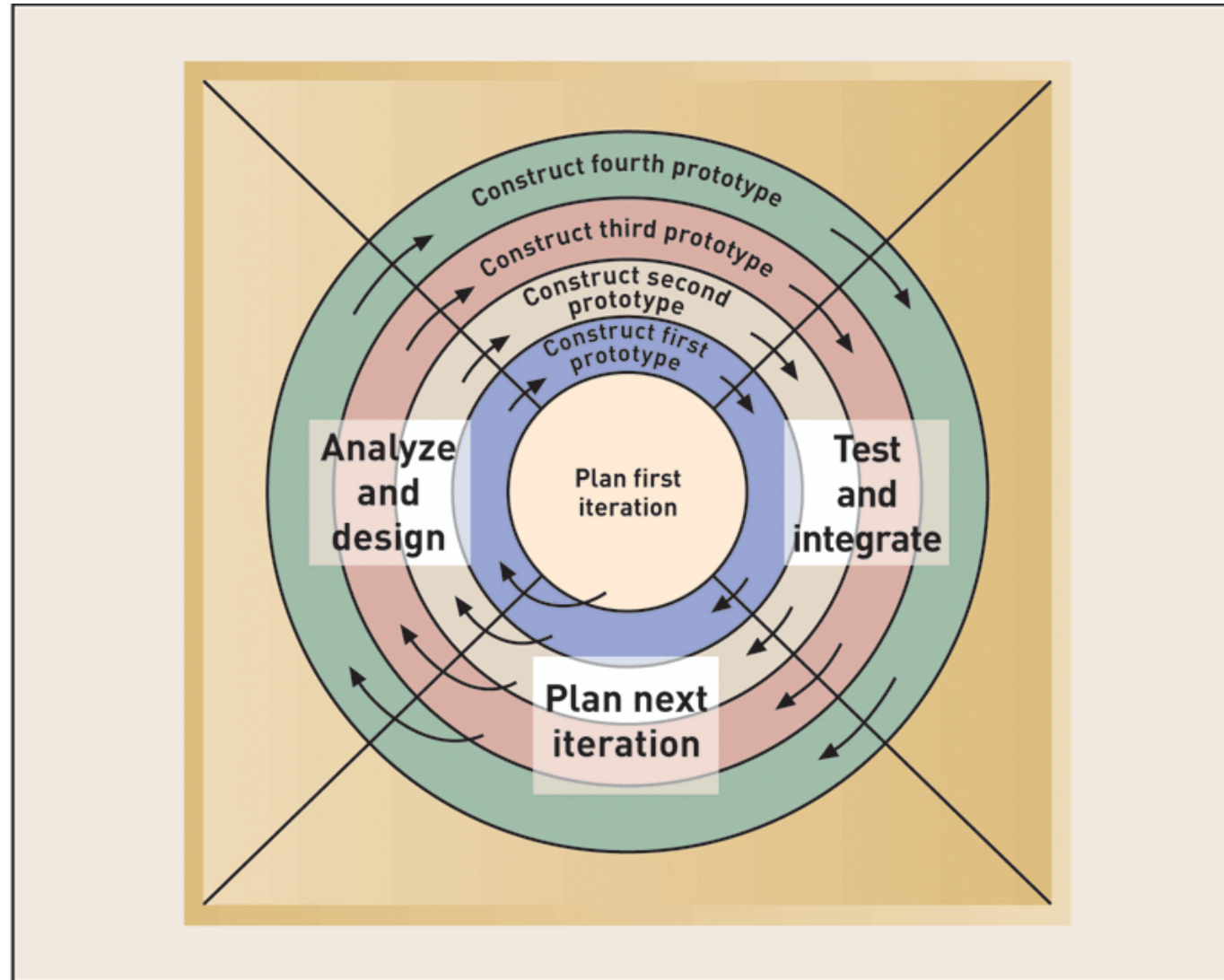
❖ بر اساس مدل مارپیچ spiral model

- چرخه پروژه از طریق فعالیت های توسعه بارها و بارها تا زمانی که پروژه کامل شود تکرار می شود
- در پایان هر چرخه ، نمونه اولیه ایجاد می شود
- بر کاهش خطر تمرکز می کند

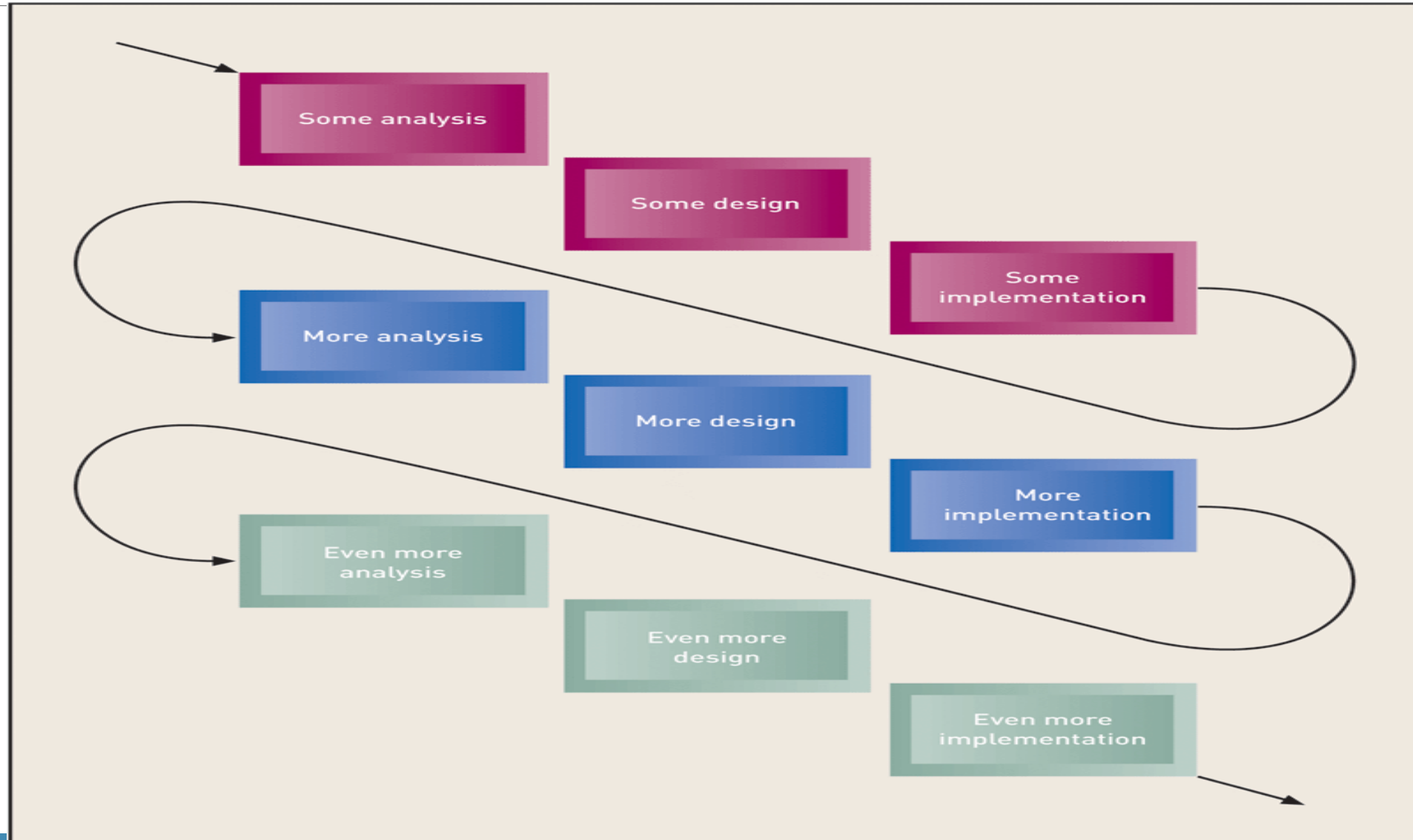
❖ تکرار : فعالیت های کاری تکرار می شوند

- هر تکرار نتیجه قبلی را اصلاح می کند
- رویکرد فرض می کند که هیچ کس بار اول آن را درست متوجه نمی شود
- برای هر تکرار یک سری پروژه کوچک وجود دارد

مدل چرخه زندگی ماریچی The Spiral Life Cycle Model



تکرار فعالیت های توسعه سیستم



فعالیت های برنامه ریزی پروژه

- ❖ مشکل و محدوده کسب و کار را تعریف کنید
- ❖ برنامه ریزی دقیق پروژه را تهیه کنید
- ❖ تایید امکان سنجی پروژه را از کارفرما دریافت کنید
 - اقتصادی،
 - سازمانی،
 - فنی،
 - منابع
 - برنامه زمانبندی
- ❖ کارکنان پروژه (مدیریت منابع) را تعیین کنید.
- ❖ راه اندازی پروژه : اطلاعیه رسمی

فعالیت های فاز تحلیل

- ❖ جمع آوری اطلاعات برای یادگیری دامنه مشکل
- ❖ الزامات سیستم را تعریف کنید
- ❖ ساخت نمونه های اولیه برای کشف نیازمندی ها را آغاز کنید
- ❖ اولویت بندی الزامات را انجام دهید
- ❖ جایگزین ها را تولید و ارزیابی کنید
- ❖ توصیه ها را با مدیریت مرور کنید

فعالیت های طراحی

- ❖ طراحی و یکپارچه سازی شبکه
- ❖ طراحی معماری اپلیکیشن
- ❖ طراحی رابط های کاربری
- ❖ طراحی رابط های سیستم
- ❖ طراحی و integrate پایگاه داده
- ❖ نمونه اولیه برای جزئیات طراحی Prototype
- ❖ طراحی و integrate کنترل های سیستم

فعالیت های اجرا Implementation

- ❖ اجزای component نرم افزار را بسازید
- ❖ تایید و تست کنید Verification And Validation
- ❖ تبدیل داده ها Convert
- ❖ آموزش کاربران و مستندسازی سیستم
- ❖ سیستم را نصب کنید Install

فعالیت های پشتیبانی

❖ سیستم را حفظ کنید Maintain

- وصله های کوچک، تعمیرات و به روز رسانی

❖ تقویت سیستم Enhance

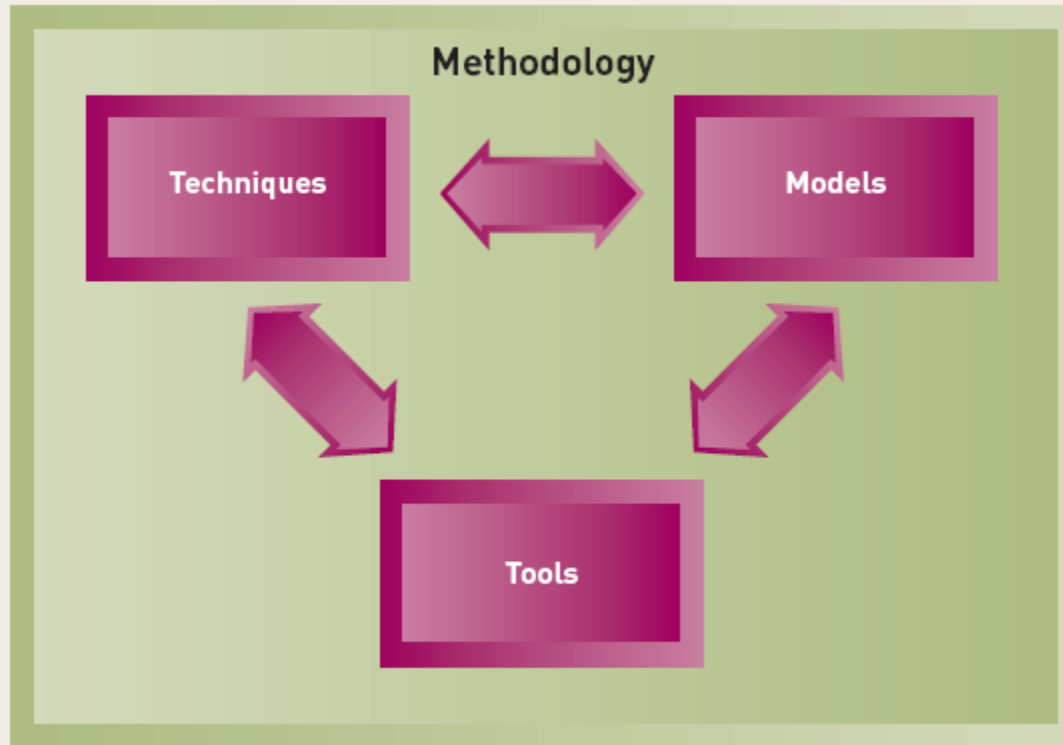
- ارتقا یا پیشرفت های کوچک برای گسترش قابلیت های سیستم
- پیشرفت های بزرگتر ممکن است نیاز به پروژه توسعه جداگانه داشته باشد

❖ پشتیبانی از کاربران Support

- میز کمک و/یا تیم پشتیبانی

❖ روش‌شناسی‌ها

- دستورالعمل‌های جامعی که باید برای تکمیل هر فعالیت SDLC دنبال کنید
- مجموعه‌ای از مدل‌ها، ابزارها و تکنیک‌ها



مدل‌ها Models

❖ مدل‌ها

- بازنمایی یک جنبه مهم از دنیای واقعی، اما نه مشابه چیز واقعی
- انتزاع برای جدا کردن جنبه استفاده می‌شود
- نمودارها و نمودارها
- کمک‌های برنامه ریزی و بودجه ریزی پروژه

Some models of system components

Flowchart
Data flow diagram (DFD)
Entity-relationship diagram (ERD)
Structure chart
Use case diagram
Class diagram
Sequence diagram

Some models used to manage the development process

Gantt chart
Organizational hierarchy chart
Financial analysis models – NPV, ROI

❖ ابزار

- پشتیبانی نرم افزاری که به ایجاد مدل ها یا سایر اجزای مورد نیاز پروژه کمک می کند
- از برنامه های طراحی ساده گرفته تا ابزارهای CASE پیچیده تا نرم افزارهای مدیریت پروژه

Project management application
Drawing/graphics application
Word processor/text editor
Visual modeling tool
Integrated development environment (IDE)
Database management application
Reverse-engineering tool
Code generator tool

❖ تکنیک

- مجموعه ای از دستورالعمل هایی که به تحلیلگران کمک می کند تا یک فعالیت یا کار توسعه سیستم را تکمیل کنند
- می تواند دستورالعمل های گام به گام یا فقط توصیه های کلی باشد

Strategic planning techniques
Project management techniques
User interviewing techniques
Data-modeling techniques
Relational database design techniques
Structured analysis technique
Structured design technique
Structured programming technique
Software-testing techniques
Object-oriented analysis and design techniques

دو رویکرد برای توسعه سیستم

❖ رویکرد سنتی : مجموعه از ماژولها

- توسعه سیستم ساخت یافته نیز نامیده می شود structured system development
- تکنیک تحلیل و طراحی ساختاریافته (SADT).
- شامل مهندسی اطلاعات (IE) است

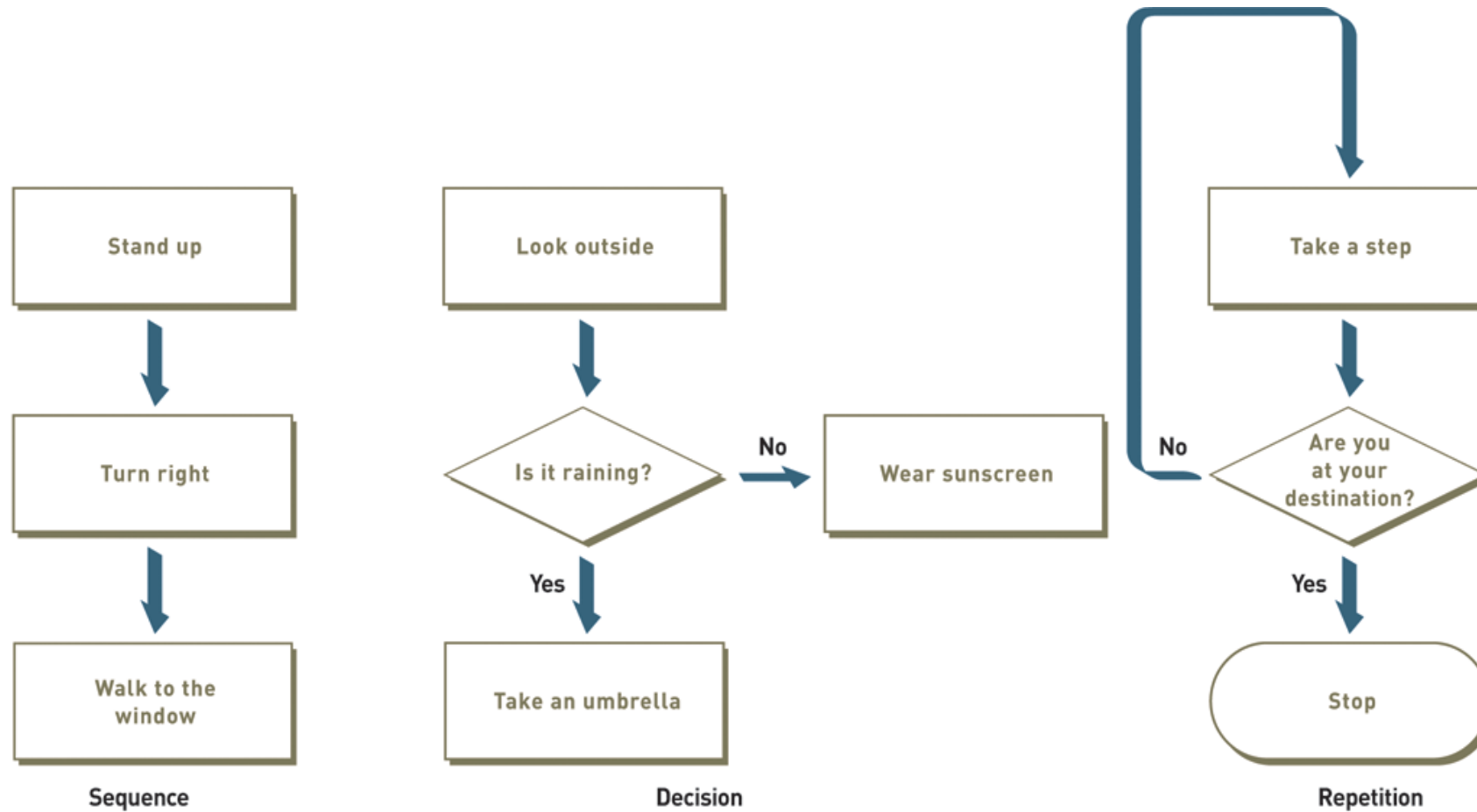
❖ رویکرد شی گرا : مجموعه ای از کلاسها

- OOA، OOD، و OOP نیز نامیده می شود
- سیستم اطلاعات را به عنوان مجموعه ای از اشیاء تعاملی می بیند که برای انجام وظایف با هم کار می کنند

❖ برنامه نویسی ساختاریافته

- کیفیت برنامه های کامپیوتری را بهبود می بخشد
- به برنامه نویسان دیگر اجازه می دهد تا به راحتی کد را بخوانند و تغییر دهند
- هر ماژول برنامه یک شروع و یک پایان دارد
- سه ساختار برنامه نویسی (توالی، تصمیم، تکرار) .

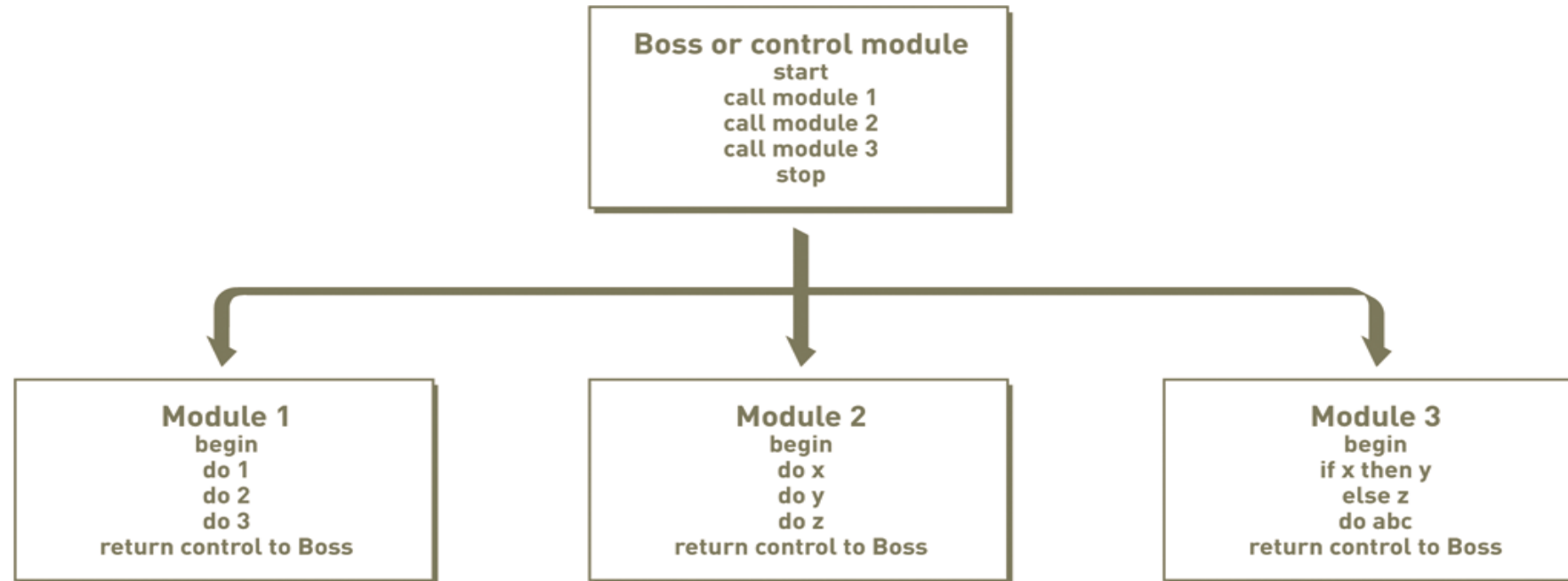
سه ساختار برنامه نویسی ساخت یافته



برنامه نویسی از بالا به پایین

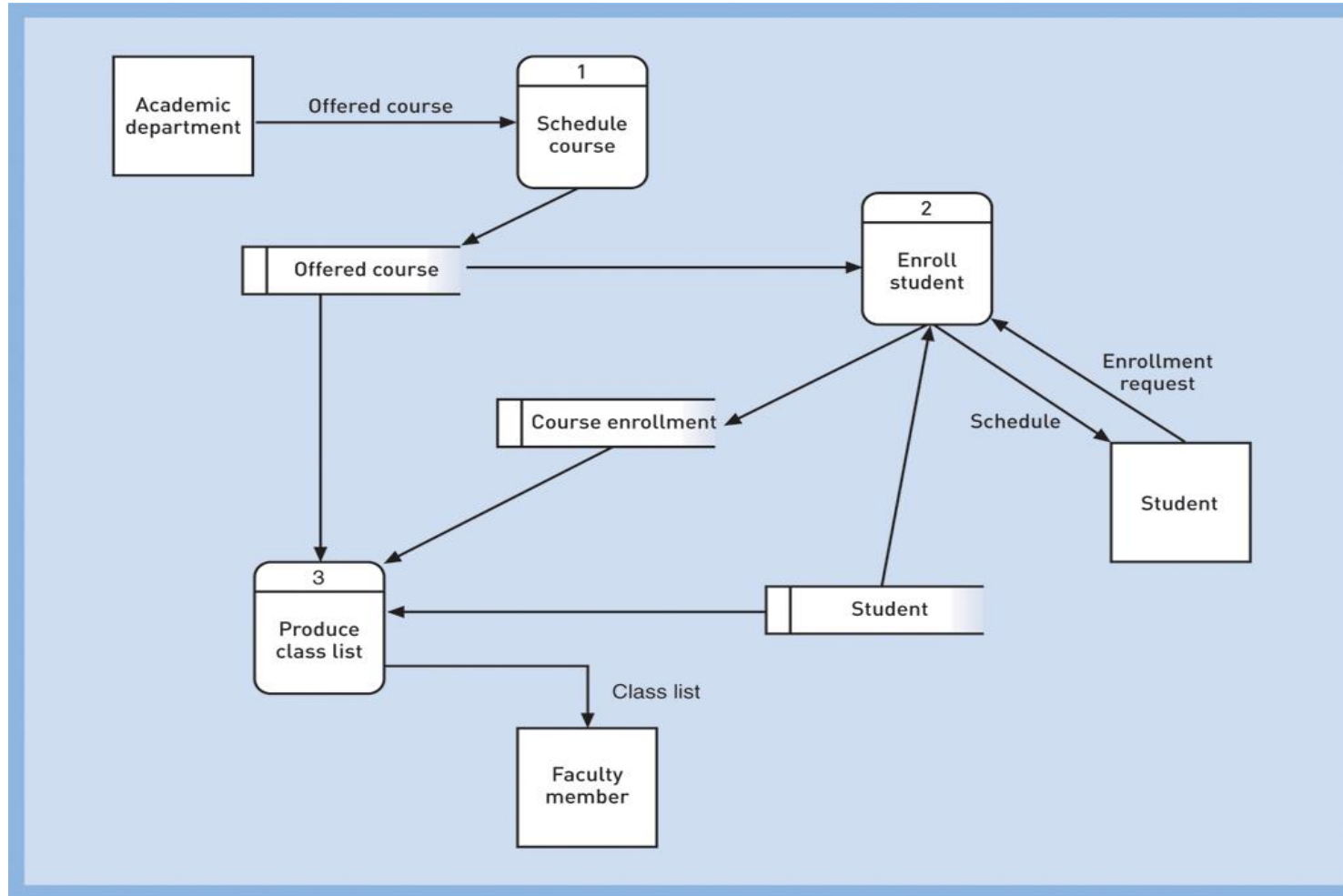
- ❖ برنامه های پیچیده را به سلسله مراتب ماژول ها تقسیم می کند
- ❖ ماژول در بالا با " فراخوانی " ماژول های سطح پایین اجرا را کنترل می کند
- ❖ برنامه نویسی ماژولار
 - مشابه برنامه نویسی از بالا به پایین
- ❖ یک برنامه برنامه های دیگر را فراخوانی می کند تا به عنوان یک سیستم واحد با هم کار کنند

برنامه نویسی از بالا به پایین یا مدولار



شکل 2-13

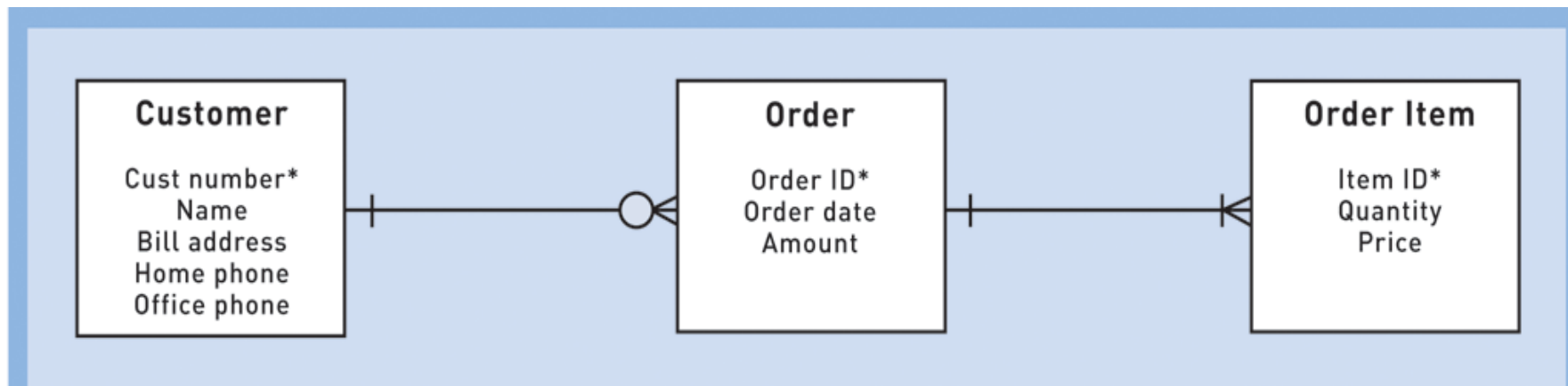
نمودار جریان داده (DFD) با استفاده از تکنیک تحلیل ساخت یافته ایجاد شد



Data Flow Diagram

نموداری که همه مفاهیم و ارتباط داده ای را نشان می دهد

نمودار موجودیت-رابطه (ERD) با استفاده از تکنیک تحلیل ساختاریافته ایجاد شده



تجزیه و تحلیل ساختار یافته منجر به طراحی ساختاریافته و برنامه نویسی ساختاریافته می شود

