Programming (coding)

specific task in SD, write computer instruction 🡪 solve given problem

code

Programmer software programs

program designs software developer

Computer instructions

Software development

broader creative process, construct a software 🡪 particular purpose

develop application, underlying systems

Computer instruction

Program

design

Java, C#, C++, Python

…

Task

Programming

Software development

***foundation***

SE

express computing ideas

software

🡸🡺 robust real-world requirements

Software in hardware devices

computer engineer

software engineering

computer science

information system

electrics engineer

IS

IT

CS

CE

Software

EE

Hardware

Organizational needs

Computing disciplines

language

new alogrithm

clever software

foundational concepts + techniques + tools

*complex software*

development process

SE & Enginerring:

* decision-based
* measure things
* disciplined process
* … team
* multiples roles
* systematically
* resuse + design
* advance priciples + standard
* best practices

|  |  |
| --- | --- |
| *SE* | *Engineering* |
| *foundation*: CS | natural sciences |
| *discrete* mathematics | continuous mathematics |
| (abstract) *logical* entities | (concrete) physical artifacts |
| no “manufacturing” | manufacturing phase |
| “maintenance”: *evolution* | conventional wear & tear |

Software: non-physical

interact 🡪 other real-world systems (social, physical)

use software process

suitable for purpose

quality

software development

an engineering discipline

SE

Software development life cycle

Waterfall process model

Perform phases in **sequence**

perform phases in **iterations** with

**feedback loops**

Spiral model

Case study

*search* relevant by keywords

keyword search engine

*retrieve* documents

by

title

*KEngine*

Local folder

www.siteX.com

*obtain* collection

of

documents

Document: sequence of words

html document: title + body

Word: provide data (~~html tags~~)

keyword non-keyword

Document *d1*

<html>

<head>

<title> welcome to my page </title>

</head>

<body>

<p>

this is a test page to test the simple Doc parser

</body>

</html>

Document *d2*

<html>

<head>

<title> welcome to my page </title>

</head>

<body>

<p>

another test page

</body>

</html>

*keyword frequencies*

<”test”, <d1, 2>> <”test”, <d2, 1>>

<”page”, <d1, 1>> <”page”, <d2, 1>>

<”to”, <d1, 1>>

<”simple”, <d1, 1>>

<”Doc”, <d1, 1>>

<”parser”, <d1, 1>>

<”another”, <d2, 1>>

*Query:*  a set of keywords 🡪 merge it 🡪 team

result: matches are sorted in **descending** order

Query: {“test”}

* result: { <d1, 2>, <d2, 1> }

Query: {“test”, “page”}

* result: { <d1, 3>, <d2, 2> }

Query: {“simple”, “Doc”, “parser”}

* result: { <d1, 3> }