A Computing Curricula Series Report
2020 December 31

Computing Curricula 2020

CC2020

Paradigms for Global Computing Education

encompassing undergraduate programs in

Computer Engineering

Computer Science

Cybersecurity

Information Systems

Information Technology

Software Engineering

with data science

Association for Computing Machinery

IEEE

IEEE computer society

A C C

Computing Curricula 2020

CC2020

Pa ad G ba C Ed ca

encompassing undergraduate programs in

Computer Engineering

Computer Science

Cybersecurity

Information Systems

Information Technology

Software Engineering

with data science

C :

. .

: AC D
@ . EEE C @ . .

B : 978-1-4503-9059-0
D : 10.1145/3467967
: https://dl.acm.org/citation.cfm?id=3467967

When available, you may order additional copies from:
AC D , . . B 30777, , 10087-0777

This report was possible by financial support from:

A C (AC)

EEE C (EEE-C)

and

AC

A (A)

E A
C A (ED-)

E

A C ( )

C (C)

The following societies and organizations have officially endorsed this report.

CC2020 Task Force

C -Chai

Ali Clea
E I

Alle Pa i h
A
A

Edi ial C i ee

A C
Eastern Institute of Technology
New Zealand

I
Hofstra University
United States of America

A
University of Alabama
United States of America

George Mason University
United States of America

!"##$%&'()*++%""##(
(

C
University of Bologna
Italy

A C
Eastern Institute of Technology
New Zealand

E C -
Latin American Center for
Computing Studies (CLEI)
Peru

F
Gannon University
United States of America

G -E
Open University
Israel

I
Hofstra University
United States of America

A
University of Alabama
United States of America

A
KTH Royal Institute of
Technology
Sweden

Keio University
Japan

H
Bentley University
United States of America

G
Vrije Universiteit
Netherlands

A
ACM India
India

George Mason University
United States of America

Bentley University
United States of America

Peking University
China

(continued on next page)

CC2020 Task Force (continued)

H A
Imam Mohammad Ibn Saud Islamic Univ.
Saudi Arabia

A
Brazilian Computer Society
Brazil

B
West Texas A&M University
United States of America

C e

A E DI : C IB A D E IE E

EFE E CE

E ec e S a

-
-
-
-
-
-

1.1: CC2020 Expectations

,-,-,.((/$*0#1"(/2$3*4#5(6%4%*&(7&8(9%44%*&((

,-,-:.((/$*0#1"(!"$7"#'%#4(

,-,-;.((/$*0#1"(<%=#$4%">(

1.2: Project Stakeholders

,-:-,.((/$*43#1"%=#(!"28#&"4(7&8("?#%$(@27$8%7&4((

I am considering a computing degree that fits my preferences. Among the candidate schools, there are several computing programs available. Are graduates of these programs expected to work primarily as individuals (e.g., doing coding) or also work with other people?

,-:-:.(() 2$$#&"(!"28#&"4((

Which areas of study does the information systems curriculum of my university emphasize more (with more detailed coverage or longer duration) than the current information systems curriculum guidelines?

,-:-;.((A&824"$>(

Our industry requires our employees to have specific knowledge at relevant knowledge levels and several key dispositions. Are there outcomes of a course in curriculum XYZ that are appropriate for the continued professional education for our employees?

,-:-B.(( ) * + 32"%& ' (C8217"*$4(7&8( ) 2$$%12D2 + (<#=#D*3#$4(

What knowledge areas are applicable for my course? Could I adopt an existing course from elsewhere to fill a gap or provide an alternative in my curriculum?

,-:-E.((/ $*F#44%*&7D(G44*1%7"%*&45(C8217"%*&7D(H$ ' 7&%I7"%*&45(7&8(G2"?*$%"%#4((

Could we accept students from a specified curriculum X to complete curriculum Y?

1.3: Project Background

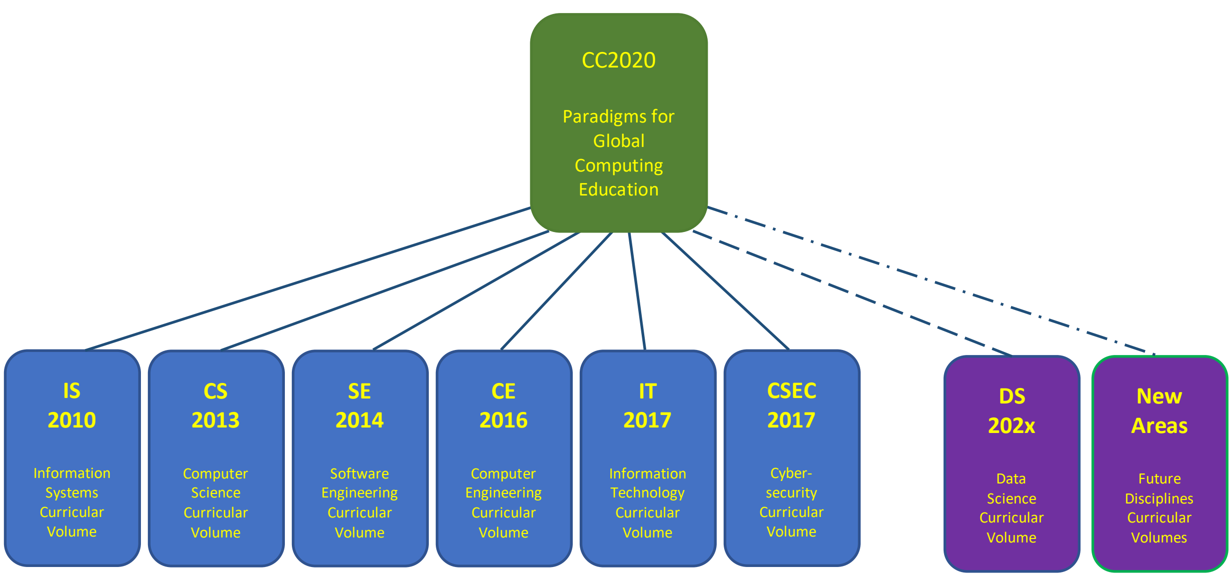
,-;-, .((J $%#F(K%4"*$>(

,-;-:.((/$*0#1"(H$'7&%I7"%*&(7&8(!"$21"2$#(

,-B-,.(() 2$$#&"(<%41%3D%&#(!"$21"2$#(

[1]

,-B-:.((L% + #D%&#(*F() 2$$%12D7$(@2%8#D%&#4(



F 1.2 C C

1 AC .

F 1.3. L ( ) CC2020 :// . 2020. /

1.7: Digest of Chapter 1

2.1: What is Computing?

:-,-,.((C7$D>(9 #7&%&'4(

:-,-:.((O#1#&"(P&8#$"7Q%&'4(

Two areas have been central in the last decade: the continued and increased need for information security, and data as a resource and driver for decision making. The protection of digital information and data; the protection of software and hardware systems and networks from unauthorized access, change, and destruction; and the education of users to follow best security practices are crucial to every organization. We rely upon a connected, networked, and complex cyberspace with vulnerabilities that is almost continuously under attack. …

During the last decade, computing has taken a new, more empirically driven path with the maturing of machine learning, the emergence of data science, and the "big data" revolution. Data science combines computing and statistical methods to identify trends in existing data and generate new knowledge, with significant applications throughout all sectors of the economy, including marketing, retail, finance, business, health care and medicine, agriculture, smart cities, and more. …

Software tools and systems for animation, visualization, virtual reality, and conceptualization have emerged as a medium for the arts (digital media and multimedia practices) and are driving advances in the entertainment industry (computer-generated graphics in films and video games, and digital methods in music recording), as well as training and education using virtual environments.

Computing has become more pervasive among a host of academic disciplines, beyond just the practical use of ubiquitous software tools. New algorithmic approaches and discoveries are helping to drive advances across a range of fields, leading to new collaborations and an increased demand for deeper knowledge of computing among academics and researchers, challenging conventional disciplinary boundaries.

2.2: Landscape of Computing Disciplines

:-:-,.((C7$▯>(<#=#▯*3+#&"4(

Most aspects of computers, data processing and the related fields discussed in this study now meet (the specifications of a discipline articulated in the paper) or may be meeting them in the next ten years.

:-:-:.(( ) *&"# + 3*7>(G8=7&1#4(

F 2.1 C ,
CC2005

2.3: Status of Computing Discipline Reports

:-;-, .(( ) * + 32"#$(C& ' %&##$%& ' (

:-;-: .(( ) * + 32"#$(! 1%#&1#(

:-;-;.(( ) >N#$4#12$%">(

:-;-B.((A&F*$+7"%*&(!>4"#+4(

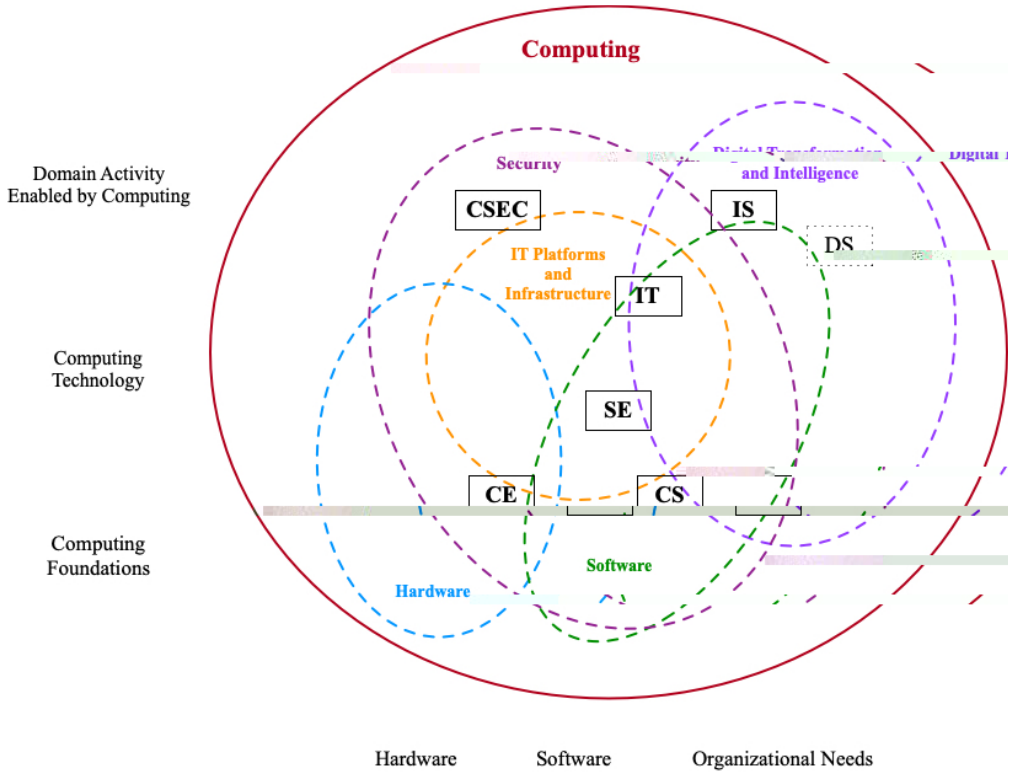
:-;-E.((A&F*$ + 7"%*&(L#1?&*D*'>(

:-;-R.((!*F"S7$#(C&'%&##$%&'(

:-;-T.((<7"7(!1%#&1#(UP&8#$(<#=#D*3+#&"V((

2.4: Extensions of Computing Disciplines

:-B-,.(() *+32"%&'(A&"#$$#D7"%*&4?%34(



F 2.2. A

<u>Legend</u>: Curricular reports: CE=computer engineering; CS=computer science; CSEC=cybersecurity; IS=information systems; IT=information technology; SE=software engineering; DS=data science (under development).

:-B-,.((C+#$'%&'() 2$$%12?7((

:-B-:.(( ) * + 32"%& ' (W(X((

:-B-;.((X(W( ) * + 32"%& ' ((

:-B-E.(() *+32"%&'(%&(/$%+7$>(7&8(!#1*&87$>(C8217"%*&((

:-B-R.(() *+32"%&'(!3#1%7D%I7"%*&4((

2.5: Digest of Chapter 2

3.1: Knowledge-Based Learning

;-,-,.((Y#7$&%&'(7&8(Z&*S#8'#(

;-,-:.((Y#7$&%&'(F$*+(Z&*S#8'#()*&"#["4(

;-,-;.((Z&*S#8'#(7&8()*+32"%&'(C8217"%*&(

3.2: Revisiting Computing Curricula 2005

;-:-,.((A&"#&"(*F()) :\\E(

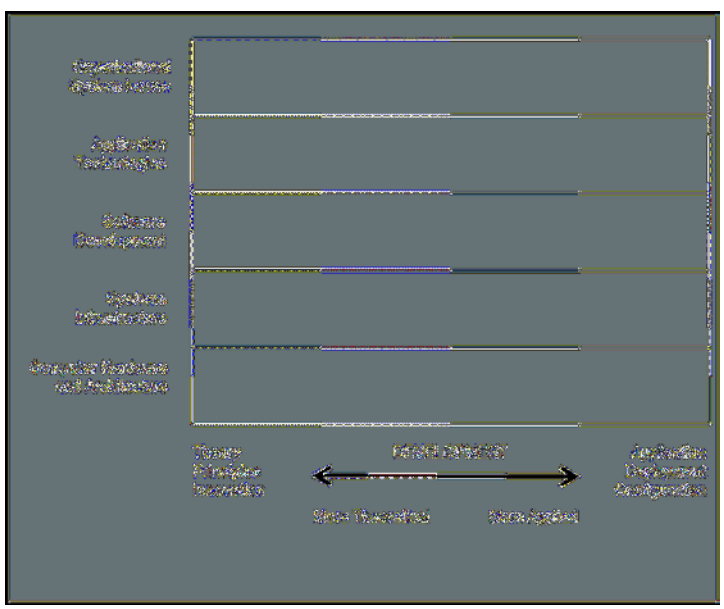
;-:-:.(() *&"#&"(*F( )) : \ \E(

;-:-; .(() * + 37$%4*&(L7ND#4((

3.1. C G (A 02, 28 3.3)

Area	Performance Capability	CE	CS	IS	IT	SE
Algorithms	Prove theoretical results	3	5	1	0	3
	Develop solutions to programming problems	3	5	1	1	3
	Develop proof-of-concept programs	3	5	3	1	3
	Determine if faster solutions are possible	3	5	1	1	3
Application programs	Design a word processor program	3	4	1	0	4
	Use word processor features well	3	3	5	5	3
	Train and support word processor users	2	2	4	5	2
	Design a spreadsheet program (e.g., Excel)	3	4	1	0	4
	Use spreadsheet features well	2	2	5	5	3
	Train and support spreadsheet users	2	2	4	5	2
Computer programming	Do small-scale programming	5	5	3	3	5
	Do large-scale programming	3	4	2	2	5
	Do systems programming	4	4	1	1	4
	Develop new software systems	3	4	3	1	5
	Create safety-critical systems	4	3	0	0	5
	Manage safety-critical projects	3	2	0	0	5
Hardware and devices	Design embedded systems	5	1	0	0	1
	Implement embedded systems	5	2	1	1	3
	Desig[illegible]puter peripherals	[illegible]	5	1	0	0
	[illegible]esign complex sensor systems	5	1	0	0	[illegible]

;-:-B.(() 2$$%12D7$(6%427D4((



F 3.1. F (CC2005)

;-:-E.((@D*N7D(7&8(H"?#$( ) *&4%8#$7"%*&4((

3.3: Limitations of a Knowledge-Based View

;-;-,.((L?#(!Q%DD4(@73(

Figure 3.3. Left: Computing occupations projected growth 2014-2024 across all sectors (job outlook) and in the computing sector. Right: Computing jobs in 2014. (*Courtesy of Bureau of Labor Statistics*)

•

;-;-:.(( ] *&^<#'$##( ) #$"%F%17"%*&4(

3.3. C C C

E - (C IA, C) (C , C) (C) ( ) I (A)	(I , A) (I C2) (I C2) C (A) (I ACA) I (I ACA)

(
;-;-;.((!Q%DD4(M$7+#S*$Q4(

3.4: Digest of Chapter 3

B-,-:.((/ $#=%*24(_ *$Q(*&( ) * + 32"%&'( ) * + 3#"#&1>((

I

I

Competencies represent a dynamic combination of cognitive and metacognitive skills, demonstration of knowledge and understanding, interpersonal, intellectual and practical skills, and ethical values.

E **C**

B-,-;.((A&%"%7D(7&8(<#=#D*3%&'()):\:\(C[3D*$7"%*&4(*F() *+3#"#&1%#4(((

4.2: A Competency Model

B-:-, .((L?#( )) : \ : \( ) * + 3#"#&1>(9 *8#Đ((((

F 4.1. C CC2020 C

B-:-: .(( ) * + 3*&#&"(<#F%&%"%*&4(((((

Knowledge

Skills

Dispositions

4. ***Task***

B-:-; .(( ) * + 3#"#&1>(! "7"# + #&"4((((((

B-:-B.(( ) * + 3*&#&"(C#+#&"4((((((

4.1. E C K

Users and Organizations	Systems Modeling	Systems Architecture and Infrastructure	Software Development	Software Fundamentals	Hardware
Social Issues and Professional Practice Security Policy and Management IS Management and Leadership Enterprise Architecture Project Management User Experience Design	Security Issues and Principles Systems Analysis & Design Requirements Analysis and Specifications Data and Information Management	Virtual Systems and Services Intelligent Systems (AI) Internet of Things Parallel and Distributed Computing Computer Networks Embedded Systems Integrated Systems Technology Platform Technologies Security Technology and Implementation	Software Quality, Verification and Validation Software Process Software Modeling and Analysis Software Design Platform-Based Development	Graphics and Visualization Operating Systems Data Structures, Algorithms and Complexity Programming Languages Programming Fundamentals Computing Systems Fundamentals	Architecture and Organization Digital Design Circuits and Electronics Signal Processing

4.2. E F K

E	
A C	A
C	A
E I	E
-	
C	C -
	A
	A
A / C	, ,
	A
- /L	
	A
C	

4.3. L C B B

Remembering	Understanding	Applying	Analyzing	Evaluating	Creating
Exhibit memory of previously learned materials by recalling facts, terms, basic concepts, and answers.	Demonstrate understanding of facts and ideas by organizing, comparing, translating, interpreting, and giving descriptions.	Solve problems in new situations by applying acquired knowledge, facts, techniques, and rules in a different way.	Examine and break information into parts by identifying motives or causes; make inferences and find evidence to support solutions.	Present and defend opinions by making judgments about information, validity of ideas, or quality of material.	Compile information together in a different way by combining elements in a new pattern or by proposing alternative solutions.

4.4. E D

Element	Elaboration	Element	Elaboration
Adaptable	Flexible; agile, adjust in response to change	Professional:	Professionalism, discretion, ethical, astute
Collaborative:	Team player, willing to work with others	Purpose-driven:	Goal driven, achieve goals, business acumen
Inventive:	Exploratory. Look beyond simple solutions	Responsible:	Use judgment, discretion, act appropriately
Meticulous:	Attentive to detail; thoroughness, accurate	Responsive:	Respectful; react quickly and positively
Passionate:	Conviction, strong commitment, compelling	Self-directed:	Self-motivated, determination, independent
Proactive:	With initiative, self-starter, independent		

B-:-E.(( ) $#7"%& ' ( ) * + 3#"#&1>(! "7"# + #&"4((((((

C : A	
C , B , , , .	
E #	4.3
A 4.1	C
D D 4.1	C
C /E 4.1	C
A C 4.2	A
4.2	A
4.2	A
- /L 4.2	A
D ( ) 4.4	
- I	

C : B	
C A , , .	
E #	4.3
C 4.1	A
4.1	A
A C 4.2	A
4.2	A
A 4.2	A
D ( ) 4.4	
- -	

C : C	
C I , , .	
E #	4.3
A 4.1	E
C 4.2	A
C 4.2	A
C 4.2	A
D ( ) 4.4	
- C	

4.3: From Competencies to Curricula

B-;-,.((A8#&"%F>%&'(7&8(G2"?*$%&'()*+3#"#&1%#4(((((((

-
-
-
-
-

B-;-:.(() * + 3#"#&1>(! 3#1%F%17"%*&4(7&8() 2$$%12D7$(! 3#1%F%17"%*&4(((((((

4.4: Digest of Chapter 4

5.1: On Visualization

E-,-,.((!*+#(J74%1(M2&1"%*&4(

E-,-:.((G&7>4¼4(*F( ) * + 3#"#&1¼#4(

5.1. C

	E	
		J
A		C

-
-
-
-
-

5.2. F

#1	#2	E
	C	H ABC ?
	C	H ?
	J	H ABC AC E C ?
	C	H ABC ?
		ABC ?
		H ABC ( ) - C 202 ?

5.2: Competency-based Visualization Examples

E-:-, .((! "28#&"(P4#( ) 74#(
(

	Dispositions
	Proactive
	Self-directed
	Passionate
	Purpose-driven
	Professional
	Responsible
	Adaptable
	Collaborative
	Responsive
	Meticulous

( ) B

	Dispositions
✔	Proactive
	Self-directed
✔	Passionate
	Purpose-driven
	Professional
✔	Responsible
	Adaptable
	Collaborative
✔	Responsive
	Meticulous

( ) A

F 5.1. C

F 5.2.

!

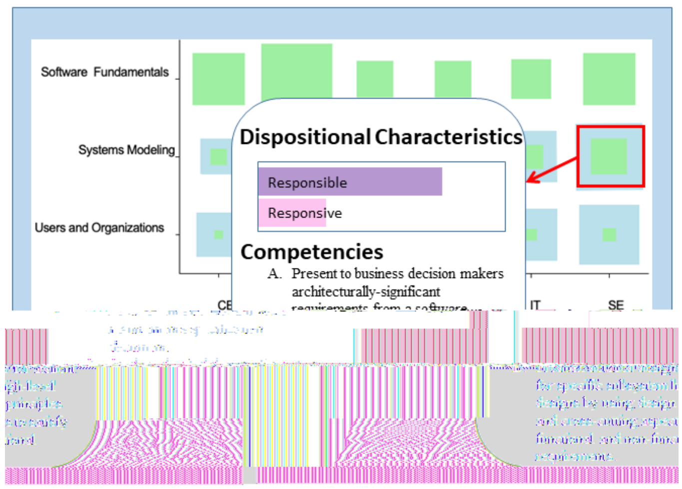
( ) C ( ) F

F 5.3 D

!

!

F 5.4.



F 5.5. D

E-:-:.((A&824"$>(P4#( ) 74#(

F 5.6

F 5.7. D

F 5.8. C C I

5.3: Knowledge-based Visualization Examples

E-;-,.(()*+32"%&'(C8217"*$(

E-;-:.((C8217"%*&7D(G2"?*$%">(

F 5.9. C

E-;-;.((6%427D%I7"%*&(*F("?#(Y7&84173#(*F()*+32"%&'(Z&*SD#8'#(L

5.3 L C K

	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max
1.1. Social Issues and Professional Practice	2	5	2	4	2	4	3	5	2	4	3	5
1.2. Security Policy and Management	1	3	2	3	4	5	2	3	2	4	2	4
1.3. IS Management and Leadership	0	2	0	2	1	2	4	5	1	2	1	2
1.4. Enterprise Architecture	0	1	0	1	1	2	3	5	1	3	1	3
1.5. Project Management	1	3	2									

F 5.10. C

E-;-B.((H"?#$(6 427 I7" *&4(

5.4: Challenges Concerning Competency Visualization

E-B-, .(( ) *&4%4"#&"(6 *17N2D7$>(

E-B-: .((C&"%">( ) * + 37$%4*&(

E-B-; .((6 %427D%I7"%*&(">3#4(

6.1: Global Context and Computing Programs

I : , , , , , ,
, , , , , - , , , ,

6.2: Computing Nomenclature((

R-:-, .((<#'$##( ] 7 + #45(`*N( / *4%"%*&4(7&8(`*N(L%"D#4((

R-:-:.((<#'$##( ] 7+#4(7&8("?#(_ *$Q3?71#(

R-:-;.((P4#(*F("?#(_ *$8(aC&'%&##$b(((

6.3: Worldwide Computing Degree Structures

R-;-, .(( ) * + 32"%& '(C8217"%*&(%&(GF$%17(

R-;-: .(( ) * + 32"%& '(C8217"%*&(%&(G24"$7D74%7((

R-;-; .(( ) * + 32"%& '(C8217"%*&(%&( ) ?%&7((

R-;-B.(( ) * + 32"%& ' (C8217"%*&(%&(C2$*3#((

R-;-E.(( ) * + 32"%& ' (C8217"%*&(%&(A&8%7(

6.1. E I (2017)

	(100,000)	F (100,000)	(100,000)
C (A)	15.27	14.16	29.43
C (-E C)	A	A	9.68
A C E	5.30	2.20	7.50

R-;-R.(( ) * + 32"%& ' (C8217"%*&(%&(`737&(

R-;-T.(( ) * + 32"%& ' (C8217"%*&(%&("?#(9 %88#(C74"(

R-;-c.(( ) * + 32"%& '(C8217"%*&(%&(Y7"%&(G + #$%17((

R-;-d.(( ) * + 32"%& '(C8217"%*&(%&( ] *$"?(G + #$%17((

R-;-, \.(( ) * + 32"%& '(C8217"%*&(%&("?#(P&%"#8(Z%& ' 8* + ((

6.4: Global Economics and Computing Education

R-B-, .((A&&*=7"%*&(! 371#4((


(a) Makerspace at OpenAir-Africa's Maker Movement


(b) Makerspace Lab at Lindenwood University

F 6.1. E

R-B-:.((M*$1#4(!?73%&'(G178#+%1(/$*'$7+4(

R-B-;.((A&&*=7"%*&(%&()*+32"%&'((

R-B-B.((C&"$#3$#&#2$4?%3(%&( ) * + 32"%& ' (

6.5: Professionalism and Ethics

R-E-, .((C"?%14(%&("?#( ) 2$$%12 2 + ((

R-E-:.((/$*F#44%*&7D(7&8(C"?%17D(_ *$Q((

R-E-;.(() 2D"2$7D(!#&4%"%=%">(7&8(<%=#$4%">(

7.1: Transforming to Competencies

T-,-,.((<%4"%&'2%4?%&'()*+3#"#&1>(F$*+(Z&*SD#8'#((

T-,-:.(() 2$$%12D7$(<>&7+%14 (

T-,-;.(() *&=#>%&'() *+32"%&'() *+3#"#&1%#4 (

T-,-B.((Z&*SD#8'# Transfer((

T-, -T.(( ] ##8(F*$(Y*17D(G873"7"%*&((

-
-
-
-

T-:-, .((/ $*F#44%*&7D(G8=%4*$>(J *7$84((

T-:-: .((_ *$Q^ ! "28>(7&8( ) **3#$7"%=#(/ $* ' $7 + 4((

T-:-; .((A&"#$&4?%3(/ $* ' $7 + 4 (

7.3: Institutional Resource Requirements

T-;-, .((G""$71"%& ' (7&8(O#"7%&%& ' (G178# + %1(C8217"*$4((

T-;-: .(( ] ##8(F*$(G8#e27"#(Y7N*$7"*$>(O#4*2$1#4((

7.4: Program Quality Assurance and Accreditation

T-B-, .((G11$#8%"7"%*&(H=#$=%#S((

T-B-: .((J#&#F%"4(*F(/$*'$7+^!3#1%F%1(G11$#8%"7"%*&((

-
-

T-B-; .((f 27D%">(G442$7&1#((

T-B-B.((@D*N7D(O#1* ' &%"%*&((

7.5: Digest of Chapter 7

8.1: Technology Trends for CC2020 and Beyond

c-,-,.(() 2$$#&"(7&8(C+#$'%&'(L#1?&*D*'%#4((

c-,-:.((C[%4"%&'()*+32"%&'(G$#74(S%"?(]*(C&8*$4#8() 2$$%12D2+((

c-,-;.((C+#$'%&'()*+32"%&'(G$#74((

– – – –

8.2: Public Engagement and the CC2020 Project

c-:-, .(( )) : \ : \(/ $*0#1"(_ #N4%"#((

c-:-: .((O#D7"%& ' ( ) 2$$%12D7(7&8( ) * + 3#"#&1%#4((

c-:-;.((/ $*0#1"(<%44#+%&7"%*&((

8.3: The Role of Competency in Future Curricular Guidelines

c-;-,.((O#1#&"() 2$$%12D7$(<#=#D*3+#&"(

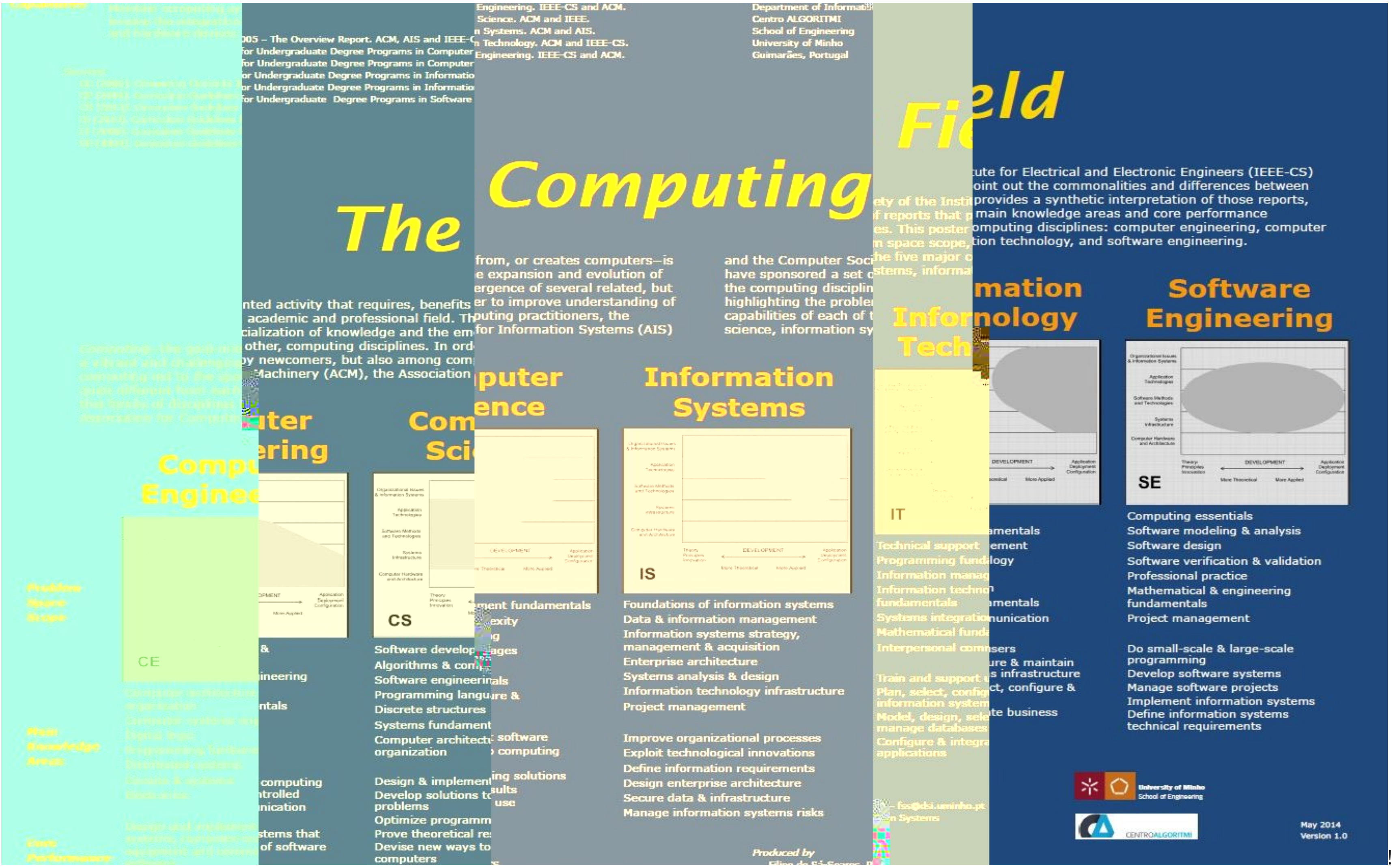
c-;-:.((M2"2$#() 2$$%12D7$(<#=#D*3+#&"(

8.4: Competency Advocacy

8.5: Future Activities

8.6: Digest of Chapter 8

Appendix A: Poster Explaining CC2005 Curricular Visuals



Appendix B: Computing Skills Frameworks

Guide to the Enterprise Information Technology Body of Knowledge

! "#$%%& '())*%+,- . / 0 1, ' %21,%34/%5621, . -3(16%78/%%

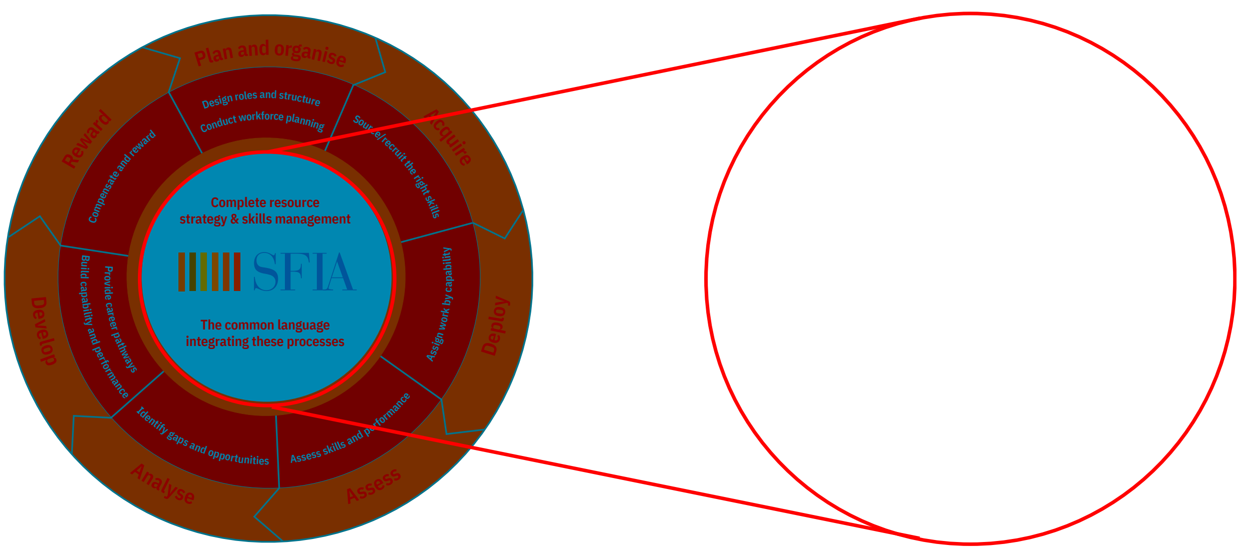


Figure B.1. The Context for SFIA – Supporting Skills and Competency Development in Industry

Figure B.2. The SFIA Context – Experience at the core

Figure B.3. The 102 SFIA Professional Skills – Skills and Competencies Performed by a Role or Individual

B.2: Skills and the European Competency Framework

Dimension 1	Five e-Competence areas derived from the ICT business macro-processes PLAN – BUILD – RUN – ENABLE – MANAGE. The main aim of dimension 1 is to facilitate navigation through the framework.
Dimension 2	A set of reference e-Competences for each area, with a generic description for each competence. Forty competences identified in total provide the European generic reference definitions of the framework.
Dimension 3	Proficiency levels of each e-Competence provide European reference level specifications on e-Competence levels e-1 to e-5, which are related to EQF levels 3-8.
Dimension 4	Samples of knowledge and skills relate to e-Competences in dimension 2. They are provided to add value and context and are not intended to be exhaustive.

Figure B.4. Four dimensions of e-CF framework

Table B.1

e-Competence Level	EQF Level

Figure B.5. The European Competency Framework Overview

B.3: Skills and the i Competency Dictionary

! "9"#$%% : -* ' % ; (<3(16 - , =%

Figure B.6. The iCD Task Dictionary Structure

! "9">$%% : -* ' % ; (<3(16 - , =% ? 4 - ,3%

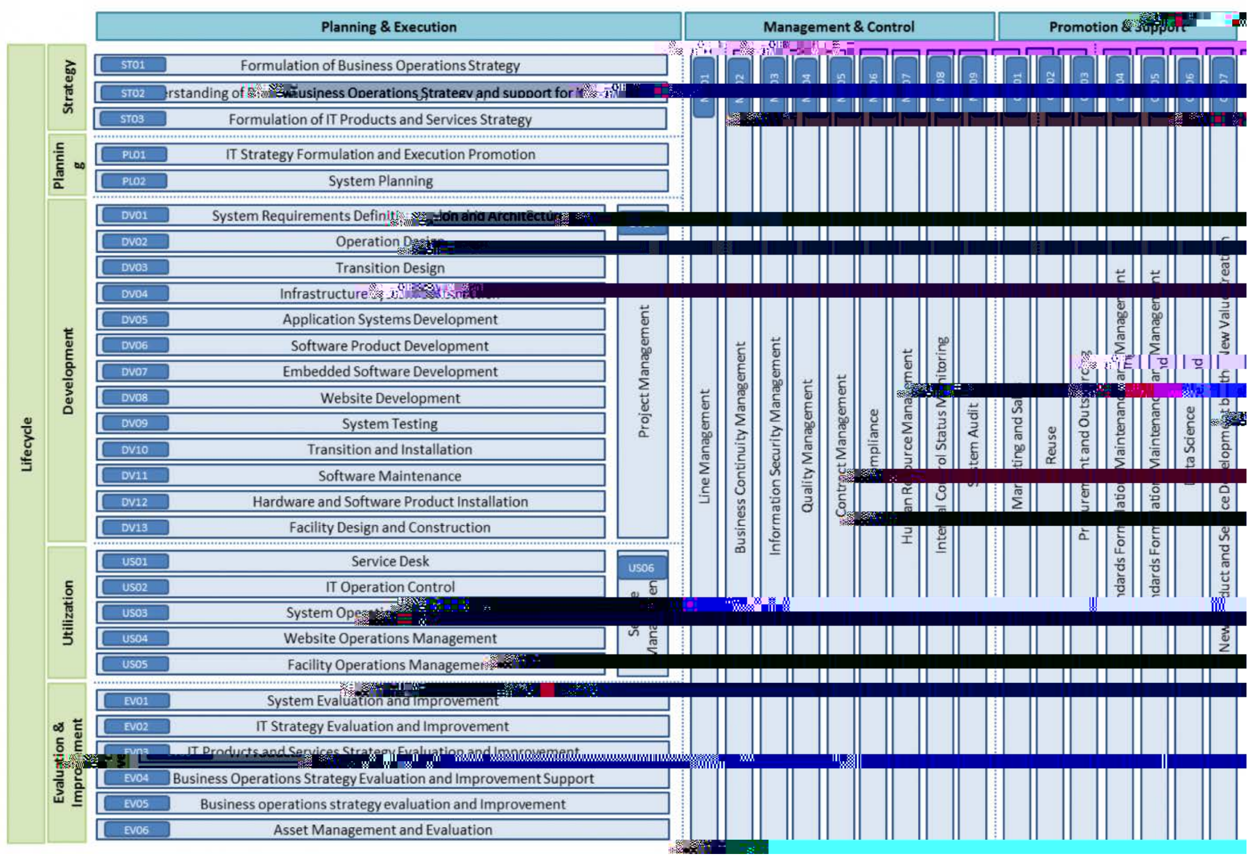


Figure B.7. The iCD Task Dictionary Chart

! "9"9$%%@A- . B)/*%12% : -* ' %@C-)D-3(16% ; (-861*3(<%E/C/)%-6F%? ,(3/,(-%

Diagnostic Level	Diagnostic Criteria
L0	No knowledge or experience
L1	Has knowledge based on training
L2	Can carry out with support or has such experience
L3	Can carry out independently or has such experience
L4	Can instruct others or has such experience

Figure B.8. Examples of Task Evaluation Diagnostic Level and Criteria

! "9"G$%%& ' ())% ; (<3(16 - , =%

Figure B.9. The iCD Skill Dictionary Structure

! "9"H$%%& ' ())% ; (<3(16 - , =% ? 4 - , 3%

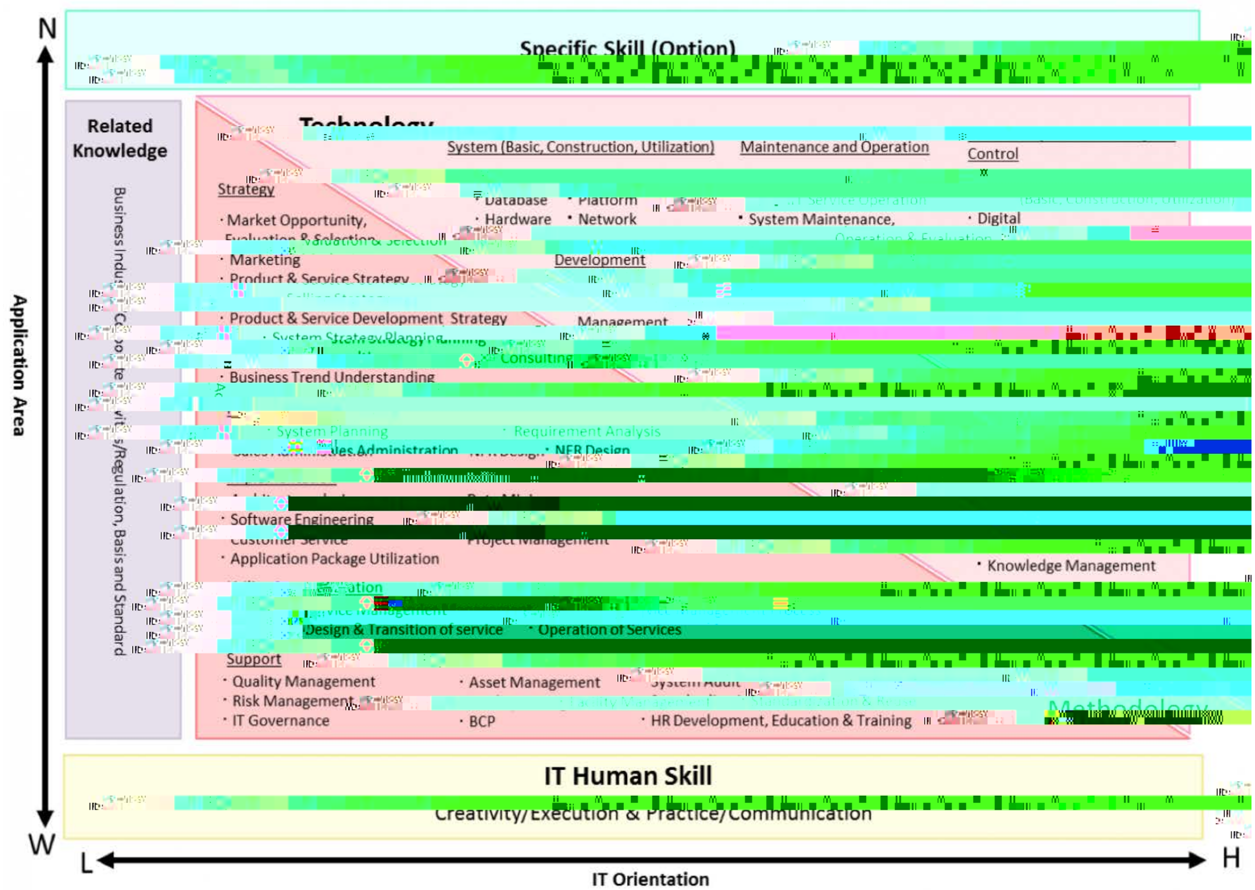


Figure B.10. The iCD Skill Dictionary Chart

! "9"I$%%& ' ())%J , 12(<(/6<=%E/C/)%

Figure B.11. Skill Proficiency Level

B.4: Skills via Enterprise Information Technology

Appendix C: Preliminary Draft Competencies – Examples

implied process

C.1: Initial CC2020 Explorations of Competencies

? "#"#$% ; , -23(68% ? 1 . B/3/6<(/*%

in context.

? "#">$%&3, -3/8=%21 ,%K/6/ , -3(68% ? 1 . B/3/6<(/*%

The Business Continuity and Information Assurance competency area mainly concerns the continuity, auditing, and assurance of information systems. It generally covers areas such as risk avoidance, security management, and quality auditing. The challenging issues related to business continuity and information assurance span from tactical and strategic to technical and operational levels. They often involve a range of processes from management, such as policy and standard-setting, to hands-on skills, such as system contingency and recovery planning.

Knowledge *Skills*

Dispositions

C.2: Draft Competencies by Discipline

?">"#$%?1 . BD3/,%@68(6//,(68%;,-23%?1 . B/3/6<(/*%

Self-contained Disposition Version

For each Knowledge Area:

A. Communicate the essential elements of the history of computer engineering, including the development of tools, standards, and constraints to a technical audience.
[ ]

B. Exercise all CE competencies in a contextually appropriate manner, demonstrating proper consideration of ethics, cultures, background, and human relationships.
[ ]

CE-CAE C c a d E ec c

1. Analyze and design circuits using electronic devices and innovate in the context of new and existing systems using those components to create new functions on varying levels of complexity bearing in mind the tradeoffs involved.
[*History & Overview; Tools & standards; electrical quantities, elements & circuits; electronic materials & devices; MOS transistors; data storage cells*]

CE-CAL C A

1. Design and/or implement classic and application-specific algorithms including parallel in multi-threading ones by relevant tools within engineering, marketing, commercial or legal constraints in the respectful and meaningful interaction with users and customers.
[*Relevant tools; algorithms - common ones, analysis, strategies*]
2. Analyze correctness, efficiency, performance, and complexity of the algorithms using order of complex terms and present honestly and comprehensively the results of the analysis for either a professional or non-professional audience.
[*Algorithmic complexity; scheduling algorithms; computability theory*]

CE-CAO C e A c ec e & O a a

1. Manage the design of computer hardware components and integrate such components to provide complete hardware systems that function reliably and efficiently demonstrating sensitivity for the context of the design envelope within which they were conceived.
[*Measuring performance; Processor organization; Distributed systems architecture; Multi/Many-core architectures; Peripheral subsystems*]

2. Simulate and evaluate the performance of parallel and sequential hardware solutions and tradeoffs involved in designing complex hardware systems considering the

Embedded Disposition Version

For each Knowledge Area:

A. Communicate the essential elements of the history of computer engineering, including the development of tools, standards, and constraints to a technical audience.
[ ]

CE-CAE C c a d E ec c

1. Analyze and design circuits for a local engineering company using electronic devices and innovate in the context of new and existing systems using those components to create new functions on varying levels of complexity bearing in mind the tradeoffs involved.
[*History & Overview; Tools & standards; electrical quantities, elements & circuits; electronic materials & devices; MOS transistors; data storage cells*]

CE-CAL C A

1. Design and/or implement classic and application-specific algorithms including parallel in multi-threading ones by relevant tools within engineering, marketing, commercial or legal constraints in the respectful and meaningful interaction with users and customers.
[*Relevant tools; algorithms - common ones, analysis, strategies*]
2. Analyze correctness, efficiency, performance, and complexity of the algorithms using order of complex terms and present honestly and comprehensively the results of the e X0424By.CSB,.u aen fBu.ylto g inei

Self-contained Disposition Version	*Embedded Disposition Version*
design of memory and arithmetical units as well as characterizing system performance using appropriate metrics. [*Processor organization; Memory system organization & architecture; Computer arithmetic; Input/Output interfacing and communication*]	complex hardware systems considering the design of memory and arithmetical units as well as characterizing system performance using appropriate metrics. [*Processor organization; Memory system organization & architecture; Computer arithmetic; Input/Output interfacing and communication*]
CE-DIG D a De 1. Using appropriate tools, design digital circuits including the basic building blocks of Boolean algebra, computer numbering systems, data encoding, combinatorial and sequential elements. [*Tools & standards; numbering systems & data encoding; Boolean algebra; digital logic, combinatorial & sequential*] 2. Design a control or datapath circuit using programmable logic and considering relevant system design constraints and testability concerns. [*Control & datapaths; programmable logic; system constraints; fault models & testing*]	**CE-DIG D a De** 1. Manage the design of a computer system for a manufacturer using appropriate tools, design digital circuits including the basic building blocks of Boolean algebra, computer numbering systems, data encoding, combinatorial and sequential elements. [*Tools & standards; numbering systems & data encoding; Boolean algebra; digital logic, combinatorial & sequential*] 2. Design a control or datapath circuit for a small company using programmable logic and considering relevant system design constraints and testability concerns. [*Control & datapaths; programmable logic; system constraints; fault models & testing*]
CE-ESY E bedded S e 1. Design and/or implement basic and advanced I/O techniques, both synchronous and asynchronous and serial/parallel, including interrupts and time considerations. [*Parallel/ serial I/O; synchronous/asynchronous I/O; interrupts and timing*] 2. Design and implement an example of an embedded system in a non-electronic device, including sensor feedback, low-power, and mobility. [*Data acquisition & sensors; embedded systems characteristics; low-power operation*]	**CE-ESY E bedded S e** 1. Present to a group of peers the design and implementation of basic and advanced I/O techniques, both synchronous and asynchronous and serial/parallel, including interrupts and time considerations. [*Parallel/ serial I/O; synchronous/asynchronous I/O; interrupts and timing*] 2. Design and implement for a professional seminar an example of an embedded system in a non-electronic device, including sensor feedback, low-power, and mobility. [*Data acquisition & sensors; embedded systems characteristics; low-power operation*]
CE-NWK C e Ne 1. Develop, deploy, maintain, and evaluate the performance of wireless and wired networking solutions in the context of relevant standards and the needs of stakeholder groups and demonstrating awareness of the foundations and history of the area. [*History and overview; Relevant tools, standards*] 2. Relate general networking competence to integrated solutions in the Internet of Things considering security and privacy aspects and the impact of solutions on citizens and society. [*Network architecture; Local and wide-area networks; Network protocols; Network applications; Network management; Data communications; Performance evaluation; Wireless sensor networks*]	**CE-NWK C e Ne** 1. Develop, deploy, maintain and evaluate the performance of wireless and wired networking solutions for a manufacturer in the context of relevant standards and the needs of stakeholder groups and demonstrating awareness of the foundations and history of the area. [*History and overview; Relevant tools, standards*] 2. Relate general networking competence to integrated solutions in the Internet of Things considering security and privacy aspects and the impact of solutions on citizens and society. [*Network architecture; Local and wide-area networks; Network protocols; Network applications; Network management; Data communications; Performance evaluation; Wireless sensor networks*]
CE-PPP P e a a P e a P ac ce 1. Analyze the importance of communication skills in a team environment and within a computer engineering group setting, discuss and determine how these skills contribute to the optimization of organization goals. [*Communication and teamwork*] 2. Evaluate the philosophical and cultural attributes necessary for maintaining a global relationship in solving a computer engineering problem that involves a	**CE-PPP P e a a P e a P ac ce** 1. Analyze the importance of communication skills in a team environment and within a computer engineering group setting, discuss and determine how these skills contribute to the optimization of organization goals. [*Communication and teamwork*] 2. Evaluate the philosophical and cultural attributes necessary for maintaining a global relationship in solving a computer engineering problem that involves a

Self-contained Disposition Version	*Embedded Disposition Version*

Self-contained Disposition Version	*Embedded Disposition Version*
CE-SRM S e Re ce Ma a e e 1. Analyze the role of single user, mobile, networked, client-server, distributed, and embedded operating systems, interrupts, and real-time support in managing system resources and interfacing between hardware and software elements considering economic, environmental, and legal limitations. [*History and overview of operating systems, Managing system resources, Operating systems for mobile devices, Support for concurrent processing*] 2. Design and implement an appropriate performance monitoring procedure for standard and virtual systems. [*Real-time operating system design, System performance evaluation; Support for virtualization*]	**CE-SRM S e Re ce Ma a e e** 1. Analyze the role of single user, mobile, networked, client-server, distributed, and embedded operating systems, interrupts, and real-time support in managing system resources and interfacing between hardware and software elements considering economic, environmental, and legal limitations. [*History and overview of operating systems, Managing system resources, Operating systems for mobile devices, Support for concurrent processing*] 2. Preset to an organization the design and implementation of appropriate performance monitoring procedures for standard and virtual systems. [*Real-time operating system design, System performance evaluation; Support for virtualization*]
CE-SWD S a e De 1. Evaluate and apply programming paradigms and languages to solve a wide variety of software design problems being mindful of trade-offs including maintainability, efficiency, and intellectual property constraints. [*Programming constructs & paradigms; problem-solving; history & overview; relevant tools, standards, constraints*] 2. Design software tests for evaluating a wide variety of performance criteria on subsystems (including usability, correctness, graceful failure, and efficiency) within the context of a complete hardware-software system. [*Software testing & quality*]	**CE-SWD S a e De** 1. Write a report for a manufacturer regarding the evaluation and application of programming paradigms and languages to solve a wide variety of software design problems being mindful of trade-offs including maintainability, efficiency, and intellectual property constraints. [*Programming constructs & paradigms; problem-solving; history & overview; relevant tools, standards, constraints*] 2. Design software testing procedures for an engineering team that evaluates a wide variety of performance criteria on subsystems (including usability, correctness, graceful failure, and efficiency) within the context of a complete hardware-software system. [*Software testing & quality*]

N be D a C e e c e = 24

Ta F ce Me be e CE S b

Barry Lunt (Leader)
Olga Bogyavlenskaya
Eric Durant
John Impagliazzo
Arnold Neville Pears

?">">$%%?1 . BD3/ ,%&<(/6</% ; , -23%?1 . B/3/6<(/*%

AL-A a d C e

A. Present to a group of peers the data characteristics of conditions or assumptions that can lead to different behaviors of specific algorithms and from the analysis, illustrate empirical studies to validate hypotheses about runtime measures.
B. Illustrate informally the time and space complexity of algorithms and use big-O notation formally to show asymptotic upper bounds and expected case bounds on time and space complexity, respectively.
C. Use recurrence relations to determine the time complexity of recursively defined algorithms by solve elementary recurrence relations and present the results to a group of scholars.
D. Determine an appropriate algorithmic approach to an industry problem and use appropriate techniques (e.g., greedy approach, divide-and-conquer algorithm, recursive backtracking, dynamic programming, or heuristic approach) that considers the trade-offs between the brute force to solve a problem.
E.

B. Analyze and evaluate a user interface that considers the context of use, stakeholder needs, state-of-the-art response interaction times, design modalities taking into consideration universal access, inclusiveness, assistive technologies, and culture-sensitive design.
C. Design and develop an int

B. Write a program for a client that correctly terminates when all concurrent tasks terminate by considering actors and/or reactive processes, deadlocks, and properly synchronized queues.
C. Write a test program for a company that reveals a concurrent programming error (e.g., missing an update when two activities both try to increment a variable).
D. Present computational results of the work and span in a program by identifying independent tasks that may be parallelized and determining the critical path for a parallel execution diagram.
E. Implement a parallel divide-and-conquer (and/or graph algorithm) for a client by mapping and reducing operations for the real industry problem and empirically measure its performance relative to its sequential analog.

PL-P a La a e

A. Present the design and implementation of a class considering object-oriented encapsulation mechanisms (e.g., class hierarchies, interfaces, and private members).
B. Produce a brief report on the implementation of a basic algorithm considering control flow in a program using dynamic dispatch that avoids assigning to a mutable state (or considering reference equality) for two different languages.
C. Present the implementation of a useful function that takes and returns other functions considering variables and lexical scope in a program as well as functional encapsulation mechanisms.
D. Use iterators and other operations on aggregates (including operations that take functions as arguments) in two programming languages and present to a group of professionals some ways of selecting the most natural idioms for each language.
E. Contrast and present to peers (1) the procedural/functional approach (defining a function for each operation with the function body providing a case for each data variant) and (2) the object-oriented approach (defining a class for each data variant with the class definition providing a method for each operation).
F. Write event handlers for a web developer for use in reactive systems such as GUIs.
G. Demonstrate program pieces (such as functions, classes, methods) that use generic or compound types, including for collections to write programs.
H. Write a program for a client to process a representation of code that illustrates the incorporation of an interpreter, an expression optimizer, and a documentation generator.
I. Use type-error messages, memory leaks, and dangling-pointer to debug a program for an engineering firm.

SDF-S a e De e e F da e a

A. Create an appropriate algorithm to illustrate iterative, recursive functions, as well as divide-and-conquer techniques and use a programming language to implement, test, and debug the algorithm for solving a simple industry problem.
B. Decompose a program for a client that identifies the data components and behaviors of multiple abstract data types and implementing a coherent abstract data type, with loose coupling between components and behaviors.
C. Design, implement, test, and debug an industry program that uses fundamental programming constructs including basic computation, simple and file I/O, standard conditional and iterative structures, the definition of functions, and parameter passing.
D. Present the costs and benefits of dynamic and static data structure implementations, choosing the appropriate data structure for modeling a given engineering problem.
E. Apply consistent documentation and program style standards for a software engineering company that contribute to the readability and maintainability of software, conducting a personal and small-team code review on program component using a provided checklist.
F. Demonstrate common coding errors, constructing and debugging programs using the standard libraries available with a chosen programming language.
G. Refactor an industry program by identifying opportunities to apply procedural abstraction.

SE-S a e E ee

A. Conduct a review of a set of software requirements for a local project, distinguishing between functional and non-functional requirements, and evaluate the extent to which the set exhibits the characteristics of good requirements.
B. Present to a client the design of a simple software system using a modeling notation (such as UML), including an explanation of how the design incorporated system design principles.

SF-S e F da e a

A. Design a simple sequential problem and a parallel version of the same problem using fundamental building blocks of logic design and use appropriate tools to evaluate the design for a commercial organization and evaluate both problem versions.
B. Develop a program for a local organization that incorporated error detection and recovery that incorporates appropriate tools for program tracing and debugging.
C. Design a simple parallel program for a corporation that manages shared resources through synchronization primitives and use tools to evaluate program performance.
D. Design and conduct a performance-oriented, pattern recognition experiment incorporating state machine descriptors and simple schedule algorithms for exploiting redundant information and data correction that is usable for a local engineering company and use appropriate tools to measure program performance.

E. Calculate average memory access time and describe the tradeoffs in memory hierarchy performance in terms of capacity, miss/hit rate, and access time for a local engineering company.
F. Measure the performance of two application instances running on separate virtual machines at a local engineering company and determine the effect of performance isolation.

SP-S c a I e a d P e a P ac ce

A. Perform a system analysis for a local organization and present the results to them in a non-technical way.
B. Integrate interdisciplinary knowledge to develop a program for a local organization.
C. Document industry trends, innovations, and new technologies and produce a report to influence a targeted workspace.
D. Present to a group of professionals an innovative computer system by using audience-specific language and examples to illustrate the group's needs.
E. Produce a document that is helpful to others that addresses the effect of societal change due to technology.
F. Adopt processes to track customer requests, needs, and satisfaction.
G.

?">"9$%%5621, . -3(16%&=*3/ . *% ; , -23% ? 1 . B/3/6<(/*%

Identifying and designing opportunities for IT-enabled organizational improvement

1. Analyze the current fit between IT strategy and organizational strategy and take corrective action to align the two, when necessary.
2. Understand General Systems theory, including its key principles and applications.
3. Model organizational processes with at least one modern business process modeling language.
4. Extract information systems requirements from future state process models.
5. Build on the foundation of risk-based management theory, apply risk analysis to real organizations.
6. Determine information systems requirements based on demonstrated needs for organizational controls.
7. Identify process performance indicators and monitors, applying industry recommendations like ITIL.
8. Understand emerging technologies to identify innovative business opportunities based on these technologies.
9. Develop business proposals based on the use of emerging technologies in an organization.
10. Apply entrepreneurial and creative thinking to transform organizations using emerging technologies.
11. Analyze and document various business stakeholders' information requirements for a proposed system.
12. Apply modern industrial practices and techniques on system documentation and user interviewing (i.e., ITIL and PMBOK).
13. Apply foundational knowledge of human-computer interaction principles to systems and user interface design.
14. Apply knowledge of data visualization and representation for an application related to analytics and complex data representation.

Analyzing trade-offs

15. Identify and design the technology alternatives and manage risk across various options within an information systems project to select the most appropriate options based on the organizational needs and implement a solution that solves key business problems.
16. Justify an information systems project in terms of technical feasibility, business viability, and cost-effectiveness to demonstrate the project's feasibility.
17. Analyze and compare solution options according to a variety of criteria and policies to evaluate the different possible solutions according to how well they promote the organizational needs.
18. Create a budget for IT-based solutions and sourcing options to enable the organization to determine the financial impact of each option.
19. Analyze the cultural differences that affect a global business environment to show how cultural standards and expectations can have a positive impact on business success to support the process of selecting between options.

Designing and implementing information systems solutions

20. Design an enterprise architecture (EA) using formal approaches by identifying EA change needs and by addressing domain requirements and technology development.
21. Apply a systematic methodology for specifying system solution options based on the requirements for the information systems solution, considering in-house development, development from third-party providers, or purchased commercial-off-the-shelf (COTS) packages.
22. Design and implement a high-quality UX (user experience) for target users to enable effective support for the users' goals in their environment.
23. Design principles of information technology security and data infrastructure at the organizational level that enable them to plan, develop, and perform security tasks and apply them to organizational systems and databases.
24. Design and implement an IT application that satisfies user needs in the context of processes that integrate analysis, design, implementation, and operations.
25. Identify data and information management alternatives and suggest the most appropriate options based on the organizational information needs.
26. Design data and information models aligned with organizational processes and compatible with data and information security management criteria.
27. Select the suitable outsourcing contractors based on the external procurement selection criteria and manage people in development teams including selected contractors in multiple projects and complex situations.
28. Understand the processes, methods, techniques, and tools that organizations use to manage information systems projects.
29. Implement modern project management approaches to information systems project, demonstrating an understanding of complex team-based activities are an inherent part of the project management.

Managing ongoing information technology operations

30. Develop and implement plans of action for optimizing the use of enterprise technology resources.
31. Develop indicators to assess application performance and scalability.
32. Monitor application performance indicators and implement corrective actions.
33. Establish practices for optimized use of information systems and plan for a long term IS viability.

34. Monitor and control an IS to track performance and fit with organizational needs.
35. Implement corrective actions by modifying the system, as necessary.
36. Negotiate and enforce contracts with providers of technology service to maintain the operational integrity of the technologies and services provided and be compliant with the roles and responsibilities of all parties involved.
37. Develop, implement, and monitor a security plan strategy based on a risk management model.
38. Implement corrective security actions, as necessary.
39. Plan and implement procedures, operations, and technologies for managing security and safety ensuring business continuity and information assurance from a disaster recovery situation.

Leadership and collaboration

40. Manage interpersonal relationships in a cross-cultural, cross-functional team.
41. Provide a clearly articulated vision for the team so that it will be able to work towards a common goal.
42. Support each member of the team in their effort to achieve their best possible level of individual performance.
43. Specify sufficiently challenging goals for the team.
44. Create a work breakdown structure, a task dependency model, and a project schedule for a globally distributed project.
45. Ensure that the project has sufficient resources and manage those resources in a context-appropriate way.
46. Allocate project tasks to project resources in an equitable and achievable way.
47. Monitor the progress the project is making.
48. Respect different viewpoints between team members.
49. View differences between team members as richness and a resource.
50. Listen and consider carefully to the viewpoints of all team members.
51. Establish and support decision structures that ensure equal opportunity to participate by all team members.
52. Align the structure of an organization so that it supports the achievement of its goals.
53. Select the organizational form based on criteria known to be effective.
54. Execute the transformation of an organization's structure so that it does not unnecessarily disrupt its work.
55. Monitor the effectiveness of an organizational structure continuously.

Communication

56. Acquire facts and opinions regarding the domain of interest from various stakeholders in relevant organizational contexts using appropriate communication methods.
57. Extract information from digital archives using modern data retrieval tools.
58. Communicate effectively in writing in a broad range of organizational contexts.
59. Select the appropriate form of written communication for a specific organizational situation.
60. Use state of the art virtual collaboration tools (such as wikis, blogs, and shared collaboration spaces) effectively in a variety of organizational situations.
61. Communicate effectively orally with different audiences and using different channels in a variety of organizational situations.
62. Identify and articulate the key elements of a persuasive presentation to support a specific viewpoint.

Negotiation

63. Apply a detailed problem analysis to determine the interests of each party in the negotiation to provide a clear proposal of the funding, time, and staff required.
64. Articulate and justify service levels for an IT service in terms of metrics that guarantee a description of the service being provided, the reliability, the responsiveness, the procedure for reporting problems, monitoring, and reporting service level, consequences for not meeting service obligations, and escape clauses or constraints.
65. Demonstrate the specification and measurements for each area in the level of service definitions to allow the quality of service to be benchmarked.
66. Identify and apply a more positive and confident approach to negotiating for each provider to support the quality enhancement of the project design as well as to ensure quality project preparation and implementation.
67. Classify the key decision points, identify who is involved in making those decisions, and understand the actions and information that will be required for such decisions to be made within an information systems team in the context of competing internal interests.

Analytical and critical thinking, including creativity and ethical analysis

68. Interpret and comply with legislative and regulatory requirements governing IT practices as well as industry standards for IT practices. Understand how culture and ethics shape compliance behavior.
69. Analyze privacy and integrity guide for all IT practices.
70. Identify complex situations and analyze the practices guide to ensure the ethical and legal corporate requirements are met.
71. Identify the value of the systems.
72. Identify the system's vulnerabilities.
73. Identify the occurrence of a threat that may exploit a system vulnerability aimed at compromising the systems.
74. Identify a complex problem in, but separate from, its environment.
75. Apply knowledge and understanding to solve the identified problem.

76. Apply creative problem solving to technology-related issues.
77. Select appropriate data collection methods and techniques for the investigation of domain activities.
78. Capture and structure data and information requirements using appropriate conceptual modeling techniques.
79. Reason effectively with a learned audience based on the results of quantitative analyses.
80. Apply adequate quantitative analysis techniques according to the data analysis goal.
81. Develop innovative and creative models that rely on new uses of existing technology or new technologies themselves.
82. Develop a plan to exploit new and emerging methods and technologies for new purposes within an organization.
83. Devise new ways of structuring and performing domain activities at different levels (individual, team, process, and organization) while considering the enabling and enhancing effects of information technology applications.
84. Estimate the benefits of the new designs, assess the consequences of their implementation, and anticipate potential adverse consequences.

.

Mathematical foundations

85. Identify those domains of interest problems that can be addressed mathematically and find a mathematical formulation for those problems.
86. Use logical thought processes to divide a problem into smaller components and make inferences based on problem components.
87. Select and implement an effective mathematical strategy.
88. Communicate mathematical results effectively to a variety of stakeholders.

N be D a C e e c e = 88

Ta F ce Me be e I a S e S b

Eiji Hayashiguchi (Leader)
Hala Alrumaih
Teresa Pereira
Ariel Sabiguero
Heikki Topi
John Impagliazzo

? ">"G$%%5621, . -3(16%: /<461)18=%? 1 . B/3/6<(/*%

ITE-CSP C be ec P c e

A. Evaluate the purpose and function of cybersecurity technology identifying the tools and systems that reduce the risk of data breaches while enabling vital organization practices. (*Cybersecurity functions*)
B. Implement systems, apply tools, and use concepts to minimize the risk to an organization's cyberspace to address cybersecurity threats. (*Tools and threats*)
C. Use a risk management approach for responding to and recovering from a cyber-attack on a system that contains high-value information and assets such as an email system. (*Response and risks*)
D. Develop policies and procedures needed to respond and remediate a cyber-attack on a credit card system and describe a plan to restore functionality to the infrastructure. (*Policies and procedures*)

ITE-GPP G ba P e a P ac ce

A. Analyze the importance of communication skills in a team environment and determine how these skills contribute to the optimization of organization goals. (*Communication and teamwork*)
B. Evaluate the specific skills necessary for maintaining continued employment in an IT career that involves system development in an environmental context. (*Employability*)
C. Develop IT policies within an organization that include privacy, legal, and ethical considerations as they relate to a corporate setting. (*Legal and ethical*)
D. Evaluate related issues facing an IT project and develop a project plan using a cost/benefit analysis including risk considerations in creating an effective project plan from its start to its completion. (*Project management*)

ITE-IMA I a Ma a e e

Express how the growth of the internet and demands for information have changed data handling and transactional and analytical processing and led to the creation of special-purpose databases. (*Requirements*)

Design and implement a physical model based on appropriate organization rules for a given scenario including the impact of normalization and indexes. (*Requirements and development*)

Create working SQL statements for simple and intermediate queries to create and modify data and database objects to store, manipulate, and analyze enterprise data. (*Testing and performance*)

Analyze ways data fragmentation, replication, and allocation affect database performance in an enterprise environment. (*Integration and evaluation*)

Perform major database administration tasks such as create and manage database users, roles and privileges, backup, and restore database objects to ensure organizational efficiency, continuity, and information security. (*Testing and performance*)

ITE-IST I e a ed S e Tec

Illustrate how to code and store characters, images, and other forms of data in computers and show why data conversion is often a necessity when merging disparate computing systems. (*Data mapping and exchange*)

Show how a commonly used intersystem communication protocol works, including its advantages and disadvantages. (*Intersystem communication protocols*)

Design, debug and test a script that includes selection, repetition, and parameter passing. (*Integrative programming and scripting*)

Illustrate the goals of secure coding and show how to use these goals as guideposts in dealing with preventing buffer overflow, wrapper code, and securing method access. (*Defensible integration*)

ITE-NET Ne

A. Analyze and compare the characteristics of various communication protocols and how they support application requirements within a telecommunication system. (*Requirements and Technologies)*
B. Analyze and compare several networking topologies in terms of robustness, expandability, and throughput used within a cloud enterprise. (*Technologies*)
C. Describe different network standards, components, and requirements of network protocols within a distributed computing setting. (*Network protocol technologies*)
D. Produce managerial policies to address server breakdown issues within a banking system. (*Risk Management*)
E. Explain different main issues related to network management. (*Network Management*)

ITE-PFT P a Tec e

Describe how the historical development of hardware and operating system computing platforms produced the computing systems we have today. (*Computing systems)*

Show how to choose among operating system options and install at least an operating system on a computer device. (*Operating systems*)

Justify the need for power and heat budgets within an IT environment, and document the factors needed when considering power and heat in a computing system. (*Computing infrastructure*)

Produce a block diagram, including interconnections, of the main parts of a computer, and illustrate methods used on a computer for storing and retrieving data. (*Architecture and organization*)

ITE-SPA S e Pa ad

A. Justify the way IT systems within an organization can represent stakeholders using different architectures and the ways these architectures relate to a system lifecycle. (*Requirements and development*)
B. Demonstrate a procurement process for software and hardware acquisition and explain the procedures one might use for testing the critical issues that could affect IT system performance. (*Testing and performance*)
C. Evaluate integration choices for middleware platforms and demonstrate how these choices affect testing and evaluation within the development of an IT system. (*Integration and evaluation*)
D. Use knowledge of information technology and sensitivity to the goals and constraints of the organization to develop and monitor effective and appropriate system administration policies within a government environment. (*System governance*)
E. Develop and implement procedures and employ technologies to achieve administrative policies within a corporate environment. (*Operational activities*)
F. Organize personnel and information technology resources into appropriate administrative domains in a technical center. (*Operational domains)*
G. Use appropriate and emerging technologies to improve the performance of systems and discover the cause of performance problems in a system. (*Performance analysis*)

ITE-SWF S a e F da e a

Use multiple levels of abstraction and select appropriate data structures to create a new program that is socially relevant and requires teamwork. (*Program development*)

Evaluate how to write a program in terms of program style, intended behavior on specific inputs, correctness of program components, and descriptions of program functionality. (*App development practices*)

Develop algorithms to solve a computational problem and explain how programs implement algorithms in terms of instruction processing, program execution, and running processes. (*Algorithm development*)

Collaborate in the creation of an interesting and relevant app (mobile or web) based on user experience design, functionality, and security analysis and build the app's program using standard libraries, unit testing tools, and collaborative version control. (*App development practices*)

ITE-UXD U e E e e ce De

A. Design an interactive application, applying a user-centered design cycle and related tools and techniques (e.g., prototyping), aiming at usability and relevant user experience within a corporate environment. (*Design tools and techniques*)
B. For a case of user-centered design, analyze and evaluate the context of use, stakeholder needs, state-of-the-art interaction opportunities, and envisioned solutions, considering user attitude and applying relevant tools and techniques (e.g., heuristic evaluation), aiming at universal access and inclusiveness, and showing a responsive design attitude, considering assistive technologies and culture-sensitive design. (*Stakeholder needs*)
C. For evaluation of user-centered design, articulate evaluation criteria and compliance to relevant standards (*Benchmarks and standards*)
D. In design and analysis, apply knowledge from related disciplines including human information processing, anthropology and ethnography, and ergonomics/human factors. (*Integrative design*)
E. Apply experience design for a service domain related to several disciplines, focusing on multiple stakeholders and collaborating in an interdisciplinary design team. (*Application design)*

ITE-WMS Web a d M b e S e

Design a responsive web application utilizing a web framework and presentation technologies in support of a diverse online community. (*Web application development*)

Develop a mobile app that is usable, efficient, and secure on more than one device. (*Mobile app development*)

Analyze a web or mobile system and correct security vulnerabilities. (*Web and mobile security*)

Implement storage, transfer, and retrieval of digital media in a web application with appropriate file, database, or streaming formats. (*Digital media storage and transfer*)

Describe the major components of a web system and how they function together, including the webserver, database, analytics, and front end. (*Web system infrastructure*)

N be D a C e e c e = 47

%

?">"H$%%&1230-,/%@68(6//,(68%;,-23%?1.B/3/6<(/*%

Software Requirements

1. Identify and document software requirements by applying a known requirements elicitation technique in work sessions with stakeholders, using facilitative skills, as a contributing member of a requirements team.
2. Analyze software requirements for consistency, completeness, and feasibility, and recommend improved requirements documentation, as a contributing member of a requirements team.
3. Specify software requirements using standard specification formats and languages that have been selected for the project and be able to describe the requirements in an understandable way to non-experts such as end-users, other stakeholders, or administrative managers, as a contributing member of a requirements team.
4. Verify and validate the requirements using standard techniques, including inspection, modeling, prototyping, and test case development, as a contributing member of a requirements team.
5. Follow process and product management procedures that have been identified for the project, as a contributing member of the requirements engineering team.

Software Design

1. Present to business decision-makers architecturally significant requirements from a software requirements specification document.
2. Evaluate and compare tradeoffs from alternative design possibilities for satisfying functional and non-functional requirements and write a brief proposal summarizing key conclusions for a client.
3. Produce a high-level design of specific subsystems that is presentable to a non-computing audience by considering architectural and design patterns.
4. Produce detailed designs for a client for specific subsystem high-level designs by using design principles and cross-cutting aspects to satisfy functional and non-functional requirements.
5. Evaluate software testing consideration of quality attributes in the design of subsystems and modules for a developer/manufacturer.
6. Create software design documents that communicate effectively to software design clients such as analysts, implementers, test planners, or maintainers.

Software Construction

1. Design and implement an API using an object-oriented language and extended libraries, including parameterization and generics on a small project.
2. Evaluate a software system against modern software practices such as defensive programming, error and exception handling, accepted fault tolerances, in a runtime mode that considers state-based table-driven constructions on a large project, as a member of a project team.
3. Develop a distributed cloud-based system that incorporates grammar-based inputs and concurrency primitives for a medium-size project and then conduct a performance analysis to fine-tune the system, as a member of a project team.

Software Testing

1. Perform an integrative test and analysis of software components by using black-box and use case techniques in collaboration with the clients.
2. Conduct a regressive test of software components for a client that considers operational profiles and quality attributes specific to an application following empirical data and the intended usages.
3. Conduct a test utilizing appropriate testing tools focused on desirable quality attributes specified by the quality control team and the client.
4. Plan and conduct process to design test cases for an organization using both clear- and black-box techniques to measure quality metrics in terms of coverage and performance.

Software Sustainment

1. Describe the criteria for transition into a sustainment status and assist in identifying applicable systems and software operational standards.
2. Relate to the needs of operational support personnel for documentation and training and help develop software transition documentation and operational support training materials.
3. Help in determining the impacts of software changes on the operational environment.
4. Describe the elements of software support activities, such as configuration management, operational software assurance, help desk activities, operational data analysis, and software retirement.
5. Perform software support activities; and interact effectively with other software support personnel.
6. Assist in implementing software maintenance processes and plans and make changes to software to implement maintenance needs and requests.

Software Process and Life Cycle

1. Engage with a team to translate a software development process into individual areas of responsibility.
2. Commit to and perform tasks related to assigned or agreed-upon areas of responsibility.
3. Propose and justify software lifecycle process improvements based on team capacity, project progress data, and quality analysis as part of a software development team's retrospective activities.

Software Systems Engineering

1. Provide a description of system engineering concepts and activities to identify problems or opportunities, explore alternatives, create models, and test them.
2. Develop the big picture of a system in its context and environment to simplify and improve system architectures for supporting system designers.
3. Develop interfaces, which interact with other subsystems. Use information hiding to isolate the contents and collaborations within subsystems, so that clients of the subsystem need not be aware of the internal design of subsystems.
4. Work effectively with engineers and developers from other disciplines to ensure effective interaction.

Software Quality

1. Distinguish quality attributes that are discernable at run-time (performance, security, availability, functionality, usability), from those not discernable at run-time (modifiability, portability, reusability, integrability, and testability) and those related to the intrinsic qualities of architecture and detailed design (conceptual integrity, correctness, and completeness).
2. Design, coordinate, and execute, within a project team, software quality assurance plans for small software subsystems and modules, considering how quality attributes are discernable. Correspondingly, measure, document, and communicate appropriately the results.
3. Perform peer code reviews for evaluating quality attributes that are not discernable at run-time.
4. Explain the statistical nature of quality evaluation when performed on software execution; develop, deploy, and implement approaches to collect statistical usage and testing outcome data; compute and analyze statistics on outcome data.
5. Interact with external entities including clients, users, and auditing agencies in conveying quality goals for processes and products.

Software Security

1. Apply the project's selected security lifecycle model (e.g., Microsoft SDL), as a contributing member of a project team.
2. Identify security requirements by applying the selected security requirements method, as a contributing member of a software project team.
3. Incorporate security requirements into architecture, high-level, and detailed design, as a contributing member of a software project team.
4. Develop software using secure coding standards.
5. Execute test cases that are specific to security.
6. Adhere to the project's software development process, as a contributing member of a software project team.
7. Develop software that supports the project's quality goals and adheres to quality requirements.

Software Safety

1. Describe the principal activities with the development of software systems, which involve safety concerns (activities related to requirements, design, construction, and quality).
2. Create and verify preliminary hazard lists; perform hazard and risk analyses, identify safety requirements.
3. Implement and verify design solutions, using safe design and coding practices, to assure that the hazards are mitigated, and the safety requirements are met.
4. Be aware of the consequences of the development of unsafe software, that is, the negative effect on those who use or receive services from the software.

Software Configuration Management

[None]

Software Measurement

1. Develop and implement plans for the measurement of software processes and work products using appropriate methods, tools, and abilities.

Human-Computer Interaction

[None]

Project Management

1. Explain the principal elements of management for a small project team.
2. Assist in the managerial aspects of a small project team, including software estimation, project planning, tracking, staffing, resource allocation, and risk management.

3. Develop and implement plans for the measurement of software processes and work products using appropriate methods and tools.
4. Work effectively with other team members in project management activities.

Behavioral Attributes

1. Engage with team members to collaborate in solving a problem, effectively applying oral and/or written communication skills. Work done towards team effort is accomplished on time; it complies with the role played in the team: it uses established quality procedures; and it advances the team effort.
2. Assist in the analysis and presentation of a complex problem, considering the needs of stakeholders from diverse cultures, needs, and/or geographic locations. Help in developing a solution for the problem and presenting it to stakeholders, explaining the economic, social, and/or environmental impact of the proposed solution. Identify areas of uncertainty or ambiguity and explain how these have been managed.
3. Analyze software employment contracts from various social and legal perspectives, ensuring that the final product conforms to professional and ethical expectations, and follows standard licensing practices.
4. Locate and make sense of learning resources, and use these to expand knowledge, skills, and dispositions. Reflect upon one's learning and how it provides a foundation for future growth.

N be D a C e e c e = 56

S a e E ee S b Me be a e Ta F ce Me be

Nancy Mead (Leader)
Hala Alrumaih
Marisa Exter
Rich LeBlanc
John Impagliazzo
Barbara Viola

S a e E ee S b Me be a e Ta F ce Me be (C b)

Kai H. Chang, Auburn University
Dick Fairley, Software and Systems Engineering Associates
Kevin Gary, Arizona State University
Thomas Hilburn, Embry-Riddle Aeronautical University
Gabriel Tamura, Universidad Icesi, Colombia
Chris Taylor, Milwaukee School of Engineering
Jim Vallino, Rochester Institute of Technology
Norha M. Villegas, Universidad Icesi, Colombia

?">"I$%% L -*3/,M*%(6%5621, . -3(16%&=*3/ . *% ; , -23%? 1 . B/3/6<(/*%

A. Anal e policies and s andards for b siness con in i and informa ion ass rance and presen he findings o a gro p of peers.
B. Plan proced res, opera ions, and echnologies for managing sec ri and safe in a disas er reco er si a ion.
C. Moni or he pro ec ion and gro h of hard are and sof are i hin an informa ion s s em for a small compan .

D. Iden if and repor da a and informa ion managemen echnolog al erna i es for a small organi a ion and s gges o managemen he mos appropria e op ions based on he organi a ional informa ion needs.
E. Iden if organi a ional policies and processes rela ed o da a and informa ion managemen i hin a eam en ironmen and ho o address informa ion and con en managemen sol ions for polic infringemen .

F. E al a e an en erprise archi ec re (EA) sing formal approaches b iden if ing he EA change needs and b addressing domain req iremen s and echnolog de elopmen hro gh a ri en repor .
G. Describe o a gro p of managers an en erprise archi ec re (EA) highligh ing sof are de elopmen and main enance b ga hering inp from he en erprise o e al a e he le el of main enance in ol ed.

H. Appl s s ainable s s em approaches b incorpora ing m l iple IT prac ices for a corpora e en ironmen in a manner ha ens res personnel pri ac and in egri .
I. De elop a polic concerning con rac s sable i hin an en erprise or go ernmen ha ens res safe and heal h s andards in compliance i h reg la or s a es and req iremen s for m al benefi irrespec i e of c l ral and personal charac eris ics.

J. Repor o he managemen of an organi a ion's ne IS me hods and rends and s gges inno a i e ac i i models ha rel on ne ses of e is ing echnologies.
K. E plain a s of e ploi ing emerging echnologies a differen le els (indi id al, eam, process, and organi a ion) and address he enabling or enhancing effec s of informa ion echnolog applica ions.
L. Repor o peers he benefi s of a ne informa ion s s em design and highligh he po en ial ad erse conseq ences of he s s em.

M. Iden if he professional managemen skills needed o design and manage an effec i e IS organi a ion ha ens res opera ional efficienc in ser ice deli er .
N. Anal e and repor IS projec managemen principles ha s ppor heir se in he organi a ion.
O. E al a e he se of informa ion s s ems and reso rces and presen he finding o he managemen of an organi a ion.

P. Iden if he effec of IS on ind s ries, firms, and ins i ions and s gges o organi a ional managers plans for ma imi ing firm benefi s associa ed i h IS design, deli er , and se.
Q. Repor o peers some o ersigh mechanisms b hich an organi a ion e al a es, direc s, and moni ors organi a ional IT b le eraging one or more go ernance frame orks and organi a ional decision-making prac ices.
R. Recommend o organi a ional managers some prac ices for minimi ing en ironmen al effec s and s gges a s for long- erm organi a ional iabili .

S. E al a e an in egra ed comm nica ion ne ork for a medi m-si e organi a ion ha incl des local-area and ide-area ne ork echnologies and specif req iremen s for a large-scale ne ork e pansion.
T. Anal e and pro ide a ri en repor of an implemen a ion archi ec re for organi a ional da a processing s s em ha ses bo h in ernal hard are reso rces.
U. Enhance he financial aspec s of a con rac ha in ol es pro iders of se eral IT infras r c re ser ices.

V. Describe o an a dience he req iremen s for an IT ar ifac ha enhances he a e is ing domain ac i i ies are s r c red and performed.
W. Repor on an IT ar ifac ha mee s specified req iremen s considering non-f nc ional req iremen s and organi a ional cons rain s.
X. Deplo an IT applica ion ha sa isfies ser needs in he con e of processes ha in egra e anal sis, design, implemen a ion, and opera ions.

24

Appendix D: Competency-Based Computing Curricula

knowing
what *knowing how* *knowing why*

task
knowledge *skills* *dispositions*

knowledge area *knowledge unit* *learning outcome*

D.1: Competency in Computing Baccalaureate Education

D.2: The CC2020 Definition of Competency

Competencies, in the most general terms, are “things” that an individual must demonstrate to be effective in a job, role, function, task, or duty. These “things” include job-relevant behavior (what a person says or does that result in good or poor performance), motivation (how a person feels about a job, organization, or geographic location), and technical knowledge/skills (what a person knows/demonstrates regarding facts, technologies, a profession, procedures, a job, an organization, etc.).

Skills

Dispositions

Figure D.2. Conceptual Structure of a Competency Specification

Figure D.3. Atomic Competency Specification: (A)

Figure D.4. Composite Competency Node: (C)

Figure D.5. Competency Tree of Atomic (A) and Composite (C) nodes

knowing what *knowing how*

(knowing why

%

%

; "9"#$% : 4/ % ? 1 . B/3/6<=%&3-3/ . /63M*%N1)/%(6%-%?1 . B/3/6<=%&B/<(2(<-3(16%

; "9">$%O610)/F8/P%QO610(68% R4-3PS%-*%-%?1 . B16/63%12%?1 . B/3/6<=%%

D.3.2.1: C D a K

knowledge area, knowledge unit, learning outcome

knowledge area, knowledge unit, learning outcome

D.3.2.2: P a a F a a K

D.3.2.3: A a D a K

; "9"9$%%& ' ())*P%QO610(68%T10PS%-*%?1 . B16/63*%12%?1 . B/3/6<=%

observable knowledge in action

knowing how

observable knowledge in action

; "9"G$%% ; (*B1*(3(16*P%QO610(68% R 4=PS%-*%-% ?1 . B16/63%12% ?1 . B/3/6<=%

D.4: Structuring Competency Statements for Competency Specification

; "G"#$%% ; /C/)1B(68% ? 1 . B/3/6<=%&3-3/ . /63*%-6F%&B/<(2(<-3(16*%

, /

%

; "G">$%%@)-U1, -3(68% ? 1 . B/3/6<=%&3-3/ . /63*%%

.

Figure D.6. Sample Free-form Competency Statement for Systems Requirements

Requirements Analysis and Specification

Written Communication

Requirements Analysis and Specification

Systems Analysis and Design

Requirements Analysis and Specification

Collaboration and Teamwork

Requirements Analysis and Specification *Collaboration and Teamwork*

Purposefully

engaged *Judgement* *Collaborative*

Figure D.7. Example Systems Requirements Competency Specification

Figure D.8. Cloud Services in Enterprise Architecture

Enterprise Architecture

IS Management and Leadership

Virtual Systems and Services

IS Management and Leadership

Research and self-starter/learner
Oral communication and presentation
Written communication

Enterprise Architecture *IS Management and Leadership*

Proactive *Self-Directed*
Professional

Figure D.9. Cloud Services in Enterprise Architecture

Propose an enterprise architecture based on the organizational business model and consistent with the mission and objectives of the organization. The architecture should propose appropriate leading-edge technologies consistent with the organizational requirements.

Figure D.10. Cloud Services in Enterprise Architecture

Meticulous *Inventive*

knowledge area, knowledge unit, learning outcome

D.6: Competency in Future Curricular Guidelines

D.7: Summary

Appendix E: From Competencies to Curricula

E.1: Competency in Future Curricular Guidelines

@"#"#$%%&3- ' /41)F/ ,*%

Prospective students **parents**

Current students

Industr

Computing educators

Educational authorities

@"#">$%%?1 . B/3/6<=%: -,8/3*%

	Education	Workforce
Singular		
Aggregate		

@"#"9$%%VD3<1 . /%@AB/<3-3(16*%-6F%E/-,6(68%&B/<(2(<-3(16*%

@">"#$%%+,//W21,.%X-,,-3(C/*%C*"%&/.(W21,.-)%&B/<(2(<-3(16*%

@">">$%%@)(<(3(68%<1 . B/3/6<(/*%

-
-
-
-
-

@">"G$%% ; /,(C(68%&/ . (W21, . -)%&B/<(2(<-3(16*%2,1 . %+,//W21, . %X-,,-3(C/*%

@">"H$%% 7D341,(68%+,//W21, . %X-,,-3(C/*%2,1 . %?1 . B/3/6<=%?1 . B16/63*%

E.3: Using Competency Specifications as a Foundation for Curriculum Specifications

Competencies

Learning experiences *sets of learning activities*

curriculum

@"9"#$%%@A(*3(68% L 1F/)*%

Figure E.1 Process for deriving learning experiences from competency specifications (adapted from MSIS 2016)

@"9"G$%%&B/<(2=(68%J,18,-. %VD3<1. /*%-*%?1. B/3/6<(/*%2,1. %J/F-818(<-)%N/ZD(,/. /63*%

E.4: Competencies and Stakeholder Value

knowledge area, knowledge unit, learning outcome

Appendix F: Repository Development

F.1: Repository Development

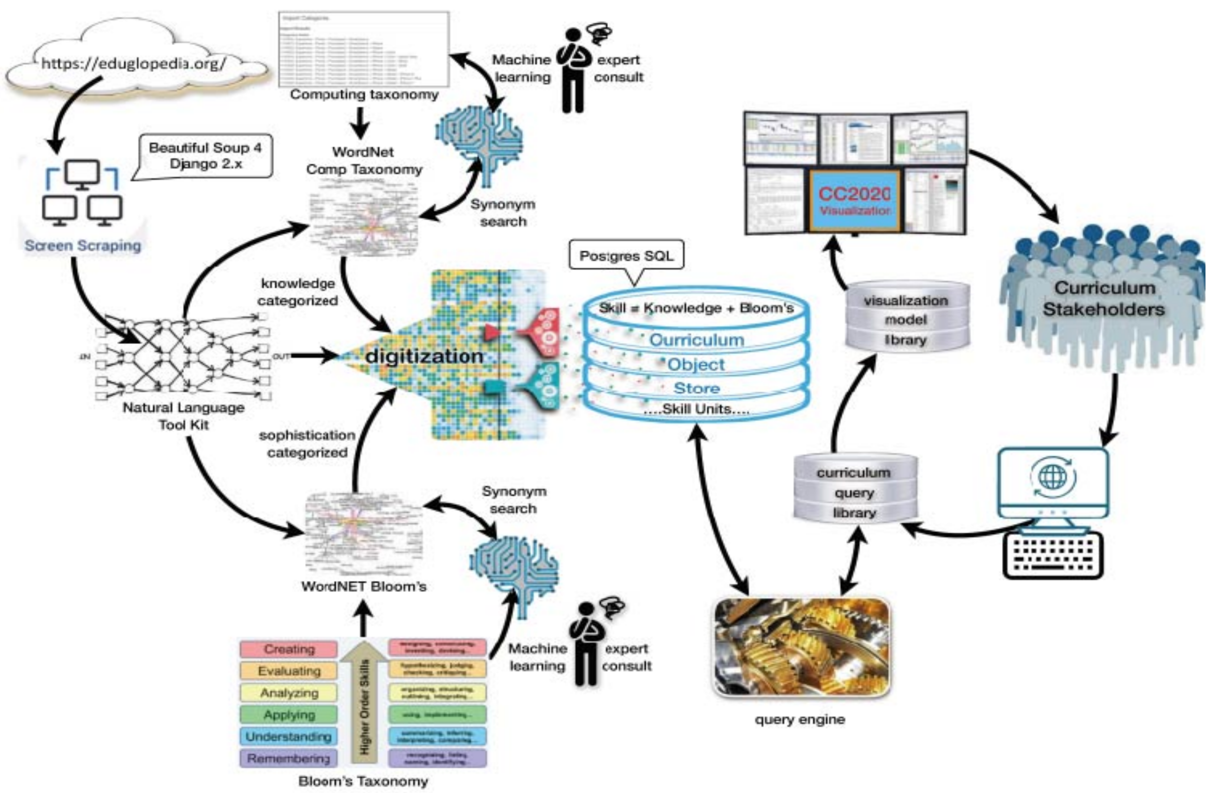


Figure F.1. Repository development process

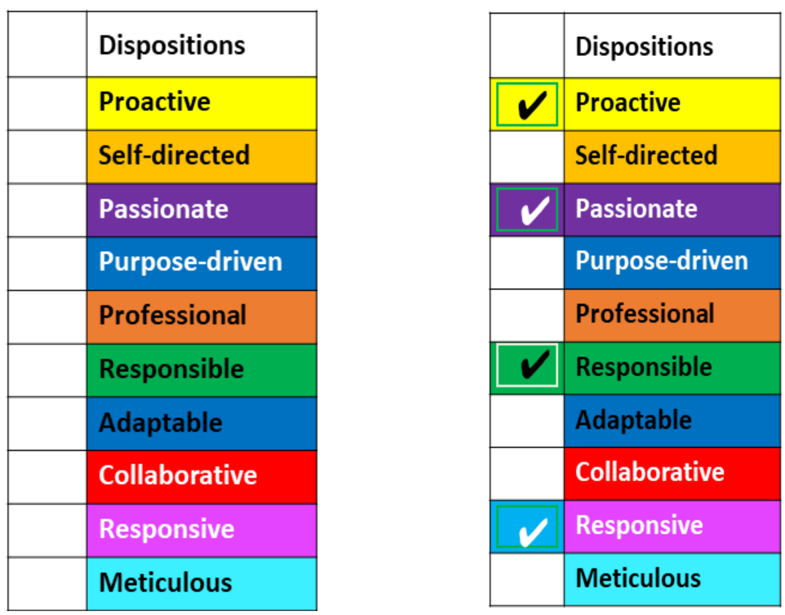
-
-
    -
    -
-

Appendix G: Additional Visualizations and Analyses

G.1: Use Case-based Analysis

. .1.1 .1.2
1 .

K"#"#$%% ? -*/%#$% [D/*3(16%2,1 . %J ,1*B/<3(C/%&3DF/63%



(a) Before choosing (b) After choosing

Figure G.1. Choosing dispositions by a prospective student

Figure G.2. The student's choice of computing categories
!!

(a) Choosing knowledge areas (b) Final result

Figure G.3. Detailed choice of knowledge areas!

Figure G.4. Mapping of chosen knowledge categories to the curricular guidelines

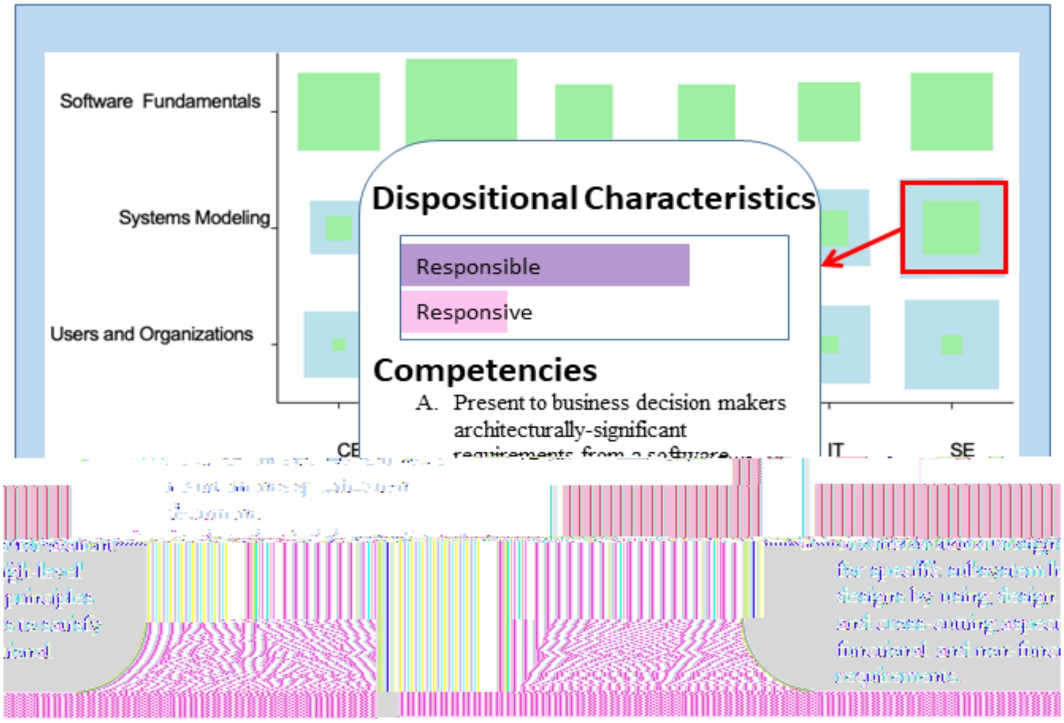


Figure G.5. Disposition and competency details

K"#">$%%? -*/%>$% [D/*3(16%2,1 . %56FD*3,=%

, , /
.
, . ,
.

.2 .3 1, ,
, ,
.
, .6.

Figure G.6. Result of knowledge areas selection

Figure G.7. Detailing skill and disposition

Figure G.8. Comparison of CS and IT based on knowledge level

K"#"9$%%? -*/%9$% [D/*3(16%2,1 . %: /-<4/,%

Figure G.9. Chosen knowledge areas for the new course design

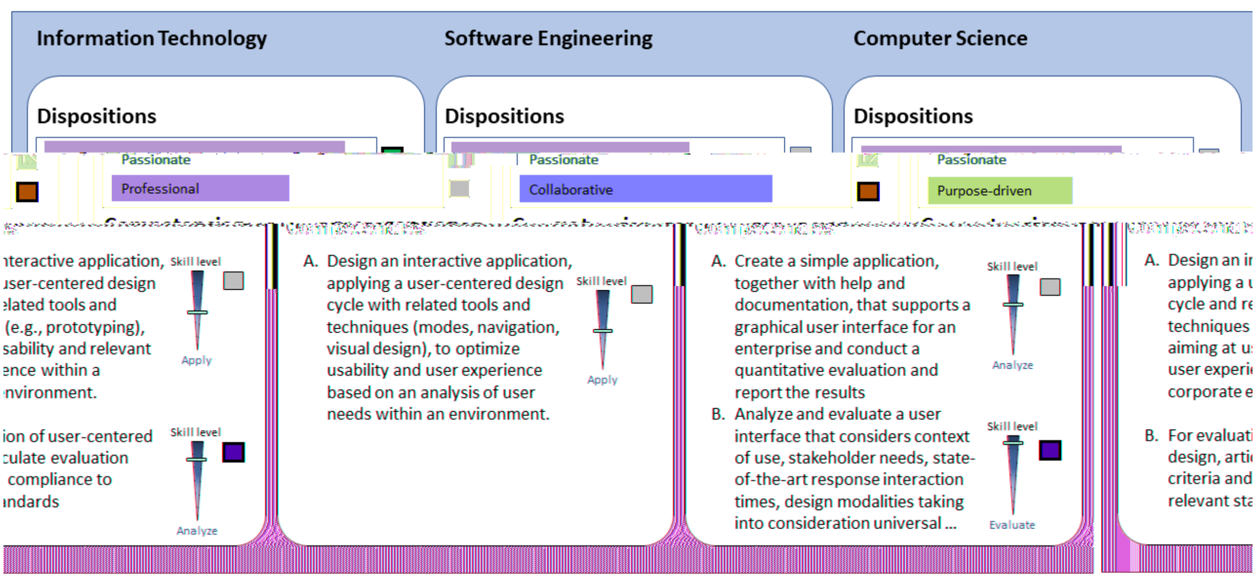


Figure G.10. Potentially relevant competencies with their skill level, and dispositions for the User Experience Design area.

Figure G.11. Resulting chosen set of competencies and dispositions

Figure G.12. Indicating the Institution to be assessed and choosing the curriculum

Figure G.13. Weight-range of knowledge areas in the curriculum guidelines for SE

Figure G.14. Inserting weights found in the BA curriculum description of the department
(note: the middle portion has been elided)

Figure G.15. Resulting state of the department's curriculum compared to the guidelines

G.2: Comparison of Competency Specifications

.16 . #
, , .
, , - .
. , -2
(-3) (-4) . ,
(-3) (-4) .

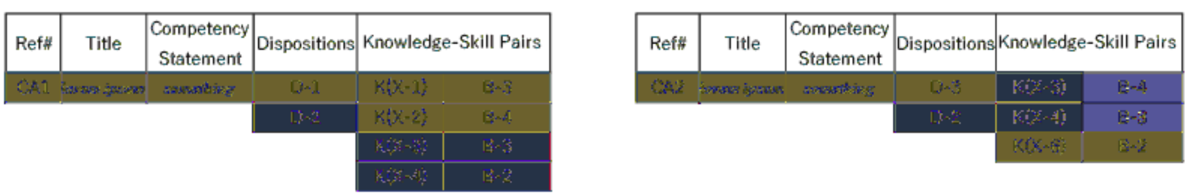


Figure G.16. Side by side comparison of competency specification
(Note: Values are examples and not actual values.)

G.3: Various Visualizations of Knowledge

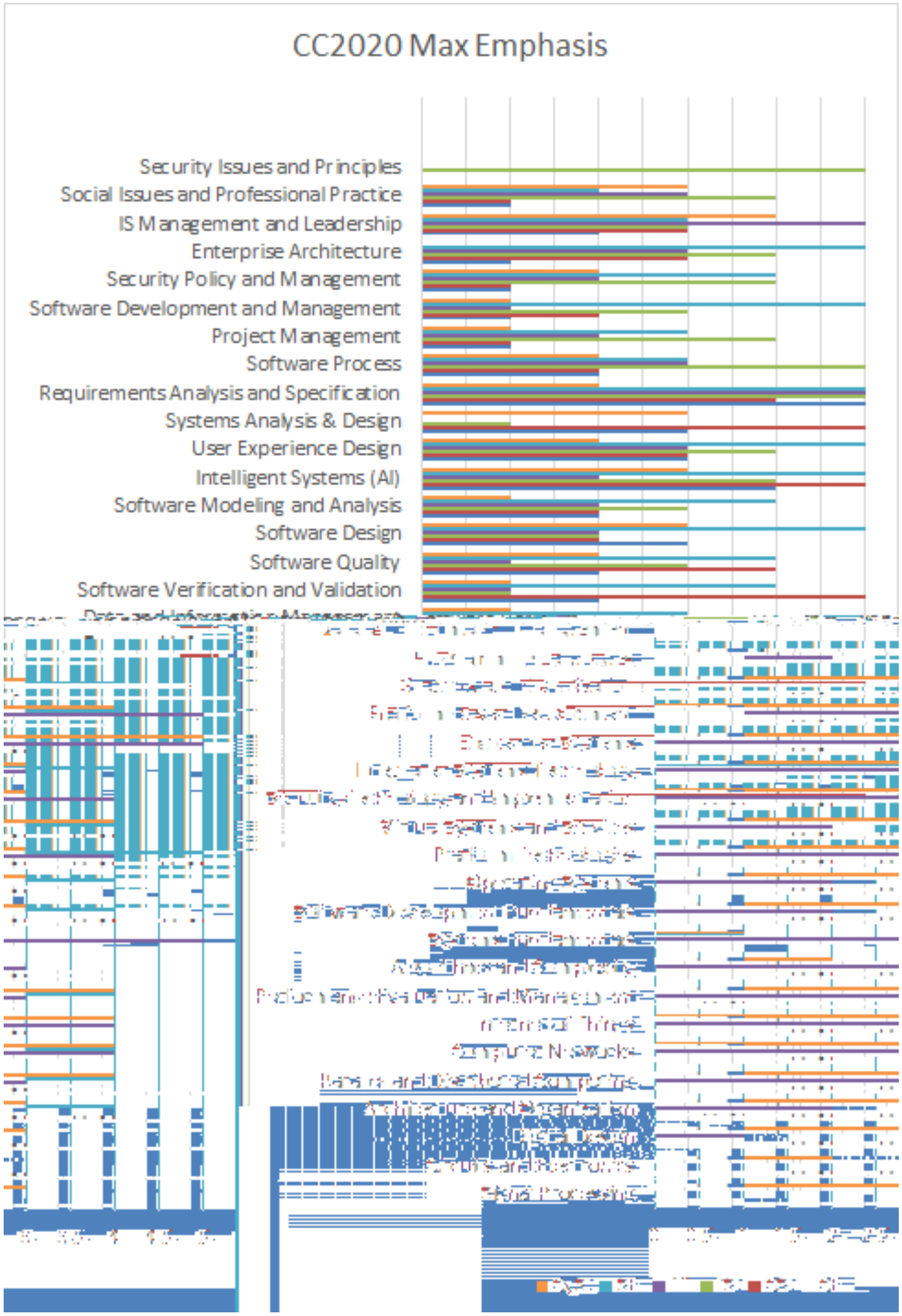


Figure G.17. Bar chart showing the maximum emphasis of knowledge areas

CC2020 Max

Figure G.18. Radar Chart showing maximum emphasis of knowledge areas

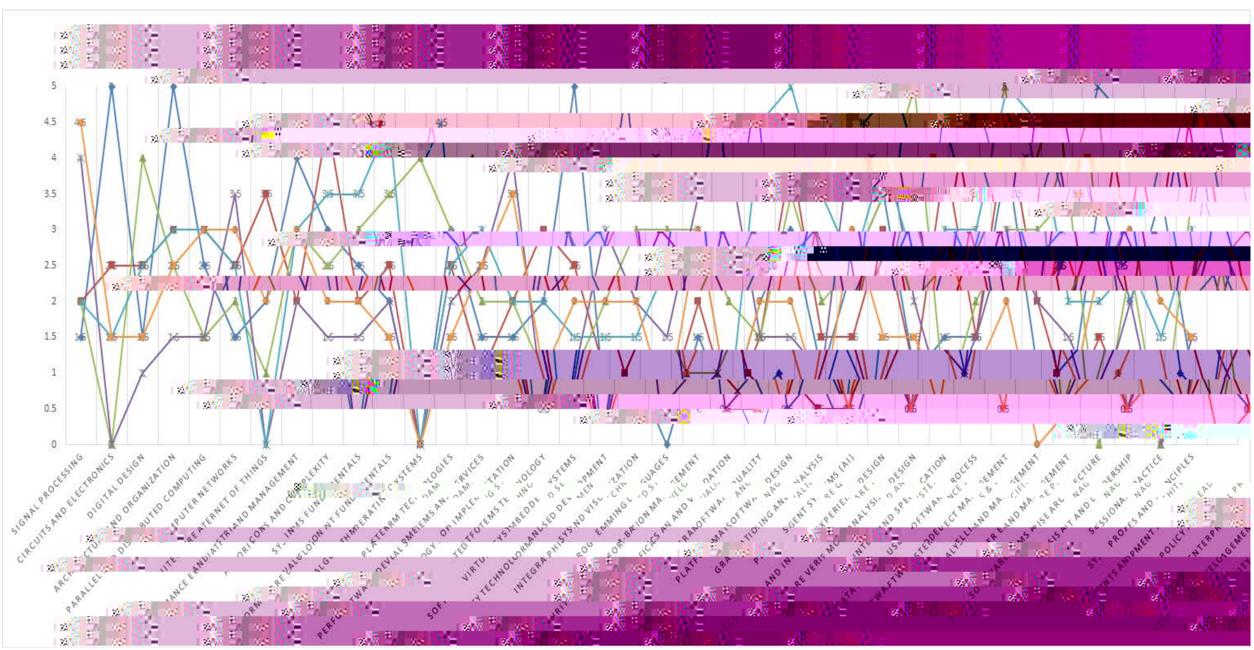


Figure G.19. Line Chart showing maximum emphasis of knowledge areas

Figure G.20. Ribbon Chart comparing the maximum emphasis of knowledge areas between CE2005 and CE2020

Figure G.21. Ribbon Chart comparing the maximum emphasis of knowledge areas between CE2020 and IS2020

G.4: Visualizing Full Curricula

Figure G.22. Graph-based structure of the core components of CS2013.

Figure G.23. Close-up of the HCI part of Figure G.22

Appendix H: Glossary and Nomenclature

H.1: CC2020 Report Definitions

Term	CC2020 Definitions

Term	CC2020 Definitions
	of either 4 to 6, 7 to 8 or 9, or 11 – 14 years.

H.2: Definitions/Nomenclature on a Global Scale!

!

Tables H.2 provides translations for the CC2020 working definitions into Arabic, Hindi, Japanese, Chinese and Russian. Similarly, Table H.3 translates the list of working definitions into French, Italian, German, Spanish (Latin American and European), and Portuguese.

Table H.2 Definition Equivalents for Arabic, Hindi, Japanese, Chinese, and Russian

Term	CC2020 Definitions	Arabic	India (Hindi)	Japanese	Chinese	Russian
Accreditation	Official approval given by an organization stating that somebody or something has achieved a required standard	اعتماد أكاديمي	मान्यता	認定	学科评估	Аккредитация
Algorithm	A set of rules to be followed in calculations or other problem-solving operations	خوارزمية	विधि	アルゴリズム	算法	Алгоритм
AP	Advanced placement not used outside the US			アドバンスド・プレイスメント	大学先修课	Not used
Baccalaureate	A bachelor's degree	بكالوريوس	स्नातक	学士	学士	Бакалавриат
Chair of Department	Head of Department or Chair of Department	رئيس القسم	विभागाध्यक्ष	学部長 or 学科長	系主任	Заведующий кафедрой
Class	A group of students studying the same course or degree	صف دراسي	कक्षा	クラス	班级	Группа, поток (several groups at a big lecture)
College	Outside the USA it can be another name for a High School or an organizational unit in a university; in the US it is a term for a post-secondary education that includes universities and colleges.	كلية	महाविद्यालय	大学	学院	Not used in the sense
Community College	Two-year school post high school primarily used in the USA, very rarely used elsewhere	كلية المجتمع			大专	Not used
Competency	Knowledge + Skills + Dispositions in context	كفاءة	योग्यता	コンピテンシ	胜任力	Компетенции

Term	CC2020 Definitions	Arabic	India (Hindi)	Japanese	Chinese	Russian
Core course/curriculum	Compulsory courses towards a degree	مقررات أساسية / خطة دراسية	पाठ्यक्रम	必修コース/カリキュラム	核心课程/课程体系	Обязательные курсы
Course	A component of a degree or in some countries a whole degree	مقرر دراسي	विषय	コース	课程	Курс
Credit hours	The number of hours for each credit towards a degree	ساعات معتمدة		単位取得時間	学时	Часы
Credits	The points a student receives after passing the assessments towards the course or degree	نقاط مكتسبة		単位	学分	Кредиты
Curriculum	All the different courses of study that are taught in a school or for a particular subject	خطة دراسية	पाठ्यक्रम	カリキュラム	课程体系	Учебный план
Engineering	Concerned with the design, building, and use of something. It does not imply a title of engineer	هندسة	अभियांत्रिकी	エンジニアリング or 工学	工程	Разработка, проектирование
Faculty	Teachers and researchers in a university	عضو هيئة تدريس	संकाय	講師 or 教員 (or 学部)	学部	ППС (abbreviation) Профессорско-преподавательский состав
Freshman	Freshman a term for a first-year degree student, generally common	طالب السنة الجامعية الأولى		1年生	一年级学生	

Term	CC2020 Definitions	Arabic	India (Hindi)	Japanese	Chinese	Russian
Information Technology (IT)	A branch of "Computing" with an approved curriculum. A fairly common global term for the "computing" technology industry as a whole. Used in many places interchangeably with Information and Communication Technology	تقنية المعلومات	सूचना प्रौद्योगिकी	情報技術	信息技术	Инормационные технологии
Junior	US term for a third-year student	طالب السنة الجامعية الثالثة		3年生	三年级学生	No special term for this
K-12	Kindergarten to year 12 (rarely used outside the US and Canada)	مراحل التعليم العام			K-12	Детский сад to year 7
Lecturer	A rank of faculty or a teacher in a university	محاضر	व्याख्याता	講師 or 教員	讲师	Лектор
Middle School	Also known as intermediate school. Different meanings in different countries generally two- or three-year schools at either 4 to 6, 7 to 8 or 9, or 11 – 14 years.	المرحلة المتوسطة	माध्यमिक विद्यालय	中学校	中学	Средняя школа 7-15/16
Module	Either a course or a part of a course	وحدة			模块	Раздел, модуль
Paper	Usually, a product a student produces to pass a course or examination, or a published article	ورقة أو مقالة علمية	परीक्षा	試験答案やレポートなどの成果物	论文	Работа, документ
Professor or a visiting professor	A Visiting Professor or in some countries (US) a part-time Professor	أستاذ جامعي أو أستاذ زائر	प्राध्यापक	非常勤教授 or 非常勤講師	访问教授	No special term for this
Program(me)	All the courses that make up a degree	برنامج دراسي		プログラム	培养方案	Специальность
Quarter	A quarter of an academic year		तिमाही	学期（クオーター）	NA	Not used
Semester	Half an academic year	فصل دراسي	छमाही	学期（セメスター）	学期	Семестр
Senior	US term for a fourth-year student	طالب السنة الجامعية الرابعة		4年生	四年级学生	Старшекурсник
Sophomore	US term for a second-year student	طالب السنة الجامعية الثانية		2年生	二年级学生	No special term for this

Term	CC2020 Definitions	Arabic	India (Hindi)	Japanese	Chinese	Russian
Subject	Similar to a course but not always used at university level	موضوع	विषय	科目	科目	Предмет
Technology	The application of scientific knowledge for practical purposes, especially in industry	التقنية	प्रौद्योगिकी	テクノロジ	技术	Технология
Trimester	One third of an academic year					Not used
Undergraduate	Studying towards a bachelor's degree	طالب المرحلة الجامعية	पूर्वस्नातक	学部生	本科生	Студент

!
!
! !

!

!

Table H.3 Definition Equivalents for French, Italian, German, Spanish (Latin America), Spanish (Europe), and Portuguese

Term	**CC2020 Definitions**	**French (Europe)**	**Italian**	**German**	**Spanish Latin America**	**Spanish Europe**	**Português**
Accreditation	Official approval given by an organization stating that somebody or something has achieved a required standard	Accréditation	Accreditamento	Akkreditierung	Acreditación	Acreditación	Credenciamento
Algorithm	A set of rules to be followed in calculations or other problem-						

Term	**CC2020 Definitions**	**French (Europe)**	**Italian**	**German**	**Spanish Latin America**	**Spanish Europe**	**Português**
College	Outside the USA it can be another name for a High School or an organizational unit in a university; in the USA it is a term for a university	Université	Collegio	Hochschule (Polytechnic)	Universidad	Universidad	Faculdade ou Instituto de Tecnologia
Community College	Two-year school post high school primarily used in the USA, very rarely used elsewhere	Classe préparatoire (to enter prestigious schools) Brevet de Technicien du Supérieur Institut Universitaire de Technologie	Centro di formazione		Not used	Not used	Curso Técnico Profissionalizante
Competency	Knowledge + Skills + Dispositions in context	Compétences	Competenza	Kompetenz	Compentencia	Compentencia	Competência
Core course/ curriculum	Compulsory courses towards a degree	Cours du tronc commun	Corsi obbligatori	Kernpflichtfach (core course)/ Kerncurriculum (core curriculum)	Asignaturas básicas	Asignaturas obligatorias	Disciplinas obrigatórias
Course	A component of a degree or in some countries a whole degree	Un cours	Insegnamento	Studiengang, Lehrgang	Curso	Curso	Disciplina
Credit hours	The number of hours for each credit towards a degree	Le temps de présentiel	Ore corrispondenti ad un credito formativo	Semesterwochenstunden	Horas por crédito	Horas por crédito	Hora-aula
Credits	The points a student receives after passing the assessments towards the course or degree	Les crédits ou ECTS	Crediti	Leistungspunkte	Créditos	Créditos	Créditos

Term	CC2020 Definitions	French (Europe)	Italian	German	Spanish Latin America	Spanish Europe	Português
Curriculum	All the different courses of study that are taught in a school or for a particular subject	Contenu pédagogique Programme pédagogique	Curriculum	Studienplan/ Lehrplan	Plan de estudios	Plan de estudios	Currículo
Engineering	Concerned with the design, building, and use of something. It does not imply a title of engineer	Ingénierie	Ingegneria	Technik	Ingeniería	Ingeniería	Engenharia
Faculty	Teachers and researchers in a university	Faculté / Institut /École (institution) Enseignant-Chercheur (people)	Collegio dei professori	Kollegium, Lehrkörper	Profesor	Profesor	Corpo docente
Freshman	Freshman a term for a first-year degree student, generally common	Un étudiant de première année	Matricola	Studienanfänger (m)/ Studienanfängerin (f)	Estudiante de primer semestre	Estudiante de primer curso	Calouro
Graduate, Post-Graduate	Graduate—has completed a bachelor's degree. Post-Graduate—completed master's and/or Doctoral degree	Licence (bac+3) Master (bac+5) Doctorat (bac+8)	laureato; laureato magistrale;	Studienabsolvent, Postgraduierter (m)/ Postgraduierte (f)	Graduado, Maestro, Doctor	Graduado, Post graduado	Graduado, Pos-graduado (Mestrado e Doutorado)
Informatics	European term for computing or sometimes information systems or computer science	Informatique (CS never Used in french)	Informatica	Informatik	Informática	Informática	Informática

Term	CC2020 Definitions	French (Europe)	Italian	German	Spanish Latin America	Spanish Europe	Português
Information and Communication Technology (ICT)	A fairly common global term for the "computing" technology industry as a whole. Used in some places interchangeably with Information Technology	Technologie de l'Information et de la Communication	Tecnologie dell'Informazione e della Comunicazione	Informations- und Kommunikationstechnologie	Tecnologías de la Información y la Comunicación	Tecnologías de la Información y la Comunicación	Tecnologia da Informação e Comunicação
Information Technology (IT)	A branch of "Computing" with an approved curriculum. A fairly common global term for the "computing" technology industry as a whole. Used in many places interchangeably with Information and Communication Technology	Technologie de l'Information	Tecnologie dell'Informazione	Informatik/ Informationstechnologie/ Informationstechnik	Tecnologías de la Información	Tecnologías de la Información	Tecnologia da Informação
Junior	USA term for a third-year student	Un étudiant de troisième année		Student/ Studentin im 3. Studienjahr	Estudiante de tercer semestre	Estudiante de tercer curso	Veterano do terceiro
K-12	Kindergarten to year 12 (rarely used outside the USA and Canada)	Le primaire (3-10yo) Le secondaire au Collège (11-15yo)		vom Kindergarten bis zum Abitur	Educación preuniversitaria	Educación preuniversitaria	Educação Básica
Lecturer	A rank of faculty or a teacher in a university	Enseignant	Docente	Dozent (m) / Dozentin (f)	Profesor de tiempo completo	Profesor Titular	Professor

Term	CC2020 Definitions	French (Europe)	Italian	German	Spanish Latin America	Spanish Europe	Português
Middle School	Also known as intermediate school. Different meanings in different countries generally two- or three-year schools at either 4 to 6, year 7 to 8 or 9, or 11 - 14.	École maternelle (3-6) École élémentaire (6-10) Collège (11-15) Lycée (16-18)	Scuola media	Hauptschule (Year 5-9), Mittelschule (Year 5- 10), Gymnasium (year 5-12)	Educación básica	Educación primaria (4-12)	Ensino Fundamental
Module	Either a course or a part of a course	Un module / une unité d'enseignement	Modulo	Modul	Módulo	Módulo	Módulo
Paper	Usually a product a student produces to pass a course or examination, or a published article		Scritto: prodotto da uno studente per superare un esame	wissenschaftliche Ar beit	Prueba, examen	Examen, Trabajo, Artículo, Prueba	Artigo, if the last work of the degree is called Trabalho Final de Curso
Professor or a visiting professor	A Visiting Professor or in some countries (USA) a part-time Professor	Professeur invité	Professore	Professor, Gastprofessor (visiting professor)	Profesor visitante	Profesor visitante (visiting professor) Profesor asociado	

Term	CC2020 Definitions	French (Europe)	Italian	German	Spanish Latin America	Spanish Europe	Português
Semester	Half an academic year	Un semestre	Semestre	Semester	Semestre	Semestre (often known as “Cuatrimestre” because classes last for 4 months + one of examinations)	Semestre
Senior	USA term for a fourth-year student	Un étudiant de quatrième année		Student/ Studentin im 4. Studi enjahr	Estudiante de cuarto semestre	Estudiante de cuarto curso	Veterano do quarto (if last year Formando)
Sophomore	USA term for a second-year student	Un étudiant de deuxième année		Student/ Studentin im 2. Studi enjahr	Estudiante de segundo semestre	Estudiante de segundo curso	Veterano do segundo ano
Subject	Similar to a course but not always used at university level	Un sujet	Materia	Fach	Asignatura	Asignatura	Matéria
Technology	The application of scientific knowledge for practical purposes, especially in industry	Une technologie	Tecnologia	Technologie	Tecnología	Tecnología	Tecnologia
Trimester	One third of an academic year	Un trimestre (3m) Un semestre (5m)	quadrimestre	Trimester	Trimestre	Trimestre	Trimestre
Undergraduate	Studying towards a bachelor’s degree	La Licence (L1-L3)	non laureato	grundständiges Studium	Estudios de grado	Estudios de grado	Graduação

!

Appendix I: Sustainable Computing and Engineering Competence in China

C
. O ,
. T F C T -E C E ,
, B B B 1
,
. T C C F
2018 Future Computer Education Summit (FCES 2018) I 1 . T
. T
B B C
. T , ,
.

I.1: Adaptable and Sustainable Competencies

O 2017 2019, B B , C E
S C B 1 . T C
C . T (IT)
(AI) . T
. T AI
. T
(. ., I 4.0) , - ,
.

C . S
, ,
. P . Y

. I , , .

I - - -

, , . T , -

K ==> A , A ==> N K . S

.

I.3: Factors Affecting Agile Computing and Engineering Education

I , . I - - - . U , . T , , - , , . U , , , . C , , , , , .

I , - . C , , , , . I , , - . B , . T , , .

A - . F , *iterative learning* . T .

- K (-1).
- U , , (-2 -3 -).
- D (-4 , ,).

U ,

- . I MOOC
- .

I.4: Open Education Ecosystems for Agile Education

I . U
. F
, , . U
, ,
. S ( ) -
, ( )
, ( )
- , ( ) -
, ( ) -
, , .

I . T
. T , IT
, , , , ,
. I , IT-
, , , .

Q . U -
. A
, ,
. Q (KPI) (
), (
), (
, ( ).

A
. E
. A
-
. I
-
. A -
, - -
, ,
,
- . T
. F I.3
.

Figure I.3. Open Ecosystems for Agile Education
(*Courtesy of Prof. Xu Xiaofei*)

I.5: Service-oriented Computing Education

S - . T ,

. E :

Q G ==> S L S C

S - . I -
, , ,

. F , -
:

- S
,
- I
,
- O - -
,
- I - ,
- P ,
- S .

Figure I.4. Promise of service-oriented education
(*Courtesy of Prof. Xu Xiaofei*)

U , . F I.4 -
.

I ,
, ,
, . A
, . T
(. ., , -)
C .

This appendix was written by John Impagliazzo and derived from notes and information received at the ACM Turing Conference (TURC) he attended in China. The "Blue Book" author has reviewed the narrative.

Appendix J: Contributors and Reviewers

T CC2020 S C T F , , . A 540 CC2020 R .

First Name	Last Name	Affiliation	Country
A 407		V	V
-	-	J N U	C
J	A	T U A	G
S	A -	W W W C (W3C)	I
I	A	P U	U S
H	A	I M I S I U	S A
M	A	I T D , IMAM U - R	S A
B	A	S U	U S
R	A	B C S	B
J L	AS M	E S P L	E
J	B	W T A&M U	U S A
O	B	P S U	R
P	B	U . R S	I

First Name	Last Name	Affiliation	Country
P	C	U B	P GRIN, I)
A	C	E I T	N
T	C	A U T	N
E	C -V	L A C C S (CLEI)	P
A	C	U S P	I
Y	D	P U	C
C	H	U . N	R S F , F I
A	D	U B	U S A
J	D	F H	R F I , G , G I
J	D	F U H	G
J F	D	U V	C
J M	D	U C	S
T M	D	P U C P	B
D	D	-	C
E	D	M S E	U S A
MJ	E	U S	S
M	E	P U	U S A
D	F	S S E A	U S
A	F	U N M	U S
S	F	G U	U S A
S	G M	S D M C E T	I
J	G -E	O U	I
K	G	A S U	U S
B F	G	U V	C
M	G	C R C	U S
G	G	U C	I

First Name	Last Name	Affiliation	Country
B	M M	M U S T	U S A
T	M V	IBM	U S A
N	M	C M U	U S A
G	M	S M C S , H W U	U K
M	M	U S M , D C S	I
M	M	A U A	M
L	M	C AIPO- S S HCI	S
E	N	U . R T V	P I E , E I
V	N	F U S C	B
S	O	U . V	I
A	P	U A	U S A
A	P	KTH R I T	S
T	P	I P V C	P
D	P	S H U	U S
M	R	D C	U S A
RITSI	R E I T S I	-	S
U	R C	UNMSM	P
A	S	U R	U
F	S	U P C	S
N	S	U M	I
Y	S	U L	F
I	S	SFIA F	I
N	S	W S U	A
W	S	UNESPAR	B
	S	U N	A
Y	S	P U	C
S	T	K U	J
G	T	U I	C
C	T	M S E	U S

First Name	Last Name	Affiliation	Country
J B	T	N - UK N T F	U K
Y	T	B D	C

References

R1: References for Report

[Aac1] American Association of Community Colleges (AACC); https://www.aacc.nche.edu/. Accessed 2020 Dec 6.
[Abe1] ABET. Accreditation Board for Engineering and Technology; https://www.abet.org/. Accessed 2020 Dec 6.
[Acc1] Accenture: Technology Vision 2020: We, the Post-Digital People; https://www.accenture.com/us-en/insights/technology/technology-trends-2020. Accessed 2020 Dec 6.
[Acm00] ACM Website; https://www.acm.org/. Accessed 2019 May 6.
[Acm01] ACM Curricula Reports Website; https://www.acm.org/education/curricula-recommendations. Accessed 2019 March 11.
[Acm02] ACM (2005). *Computing Curricula 2005 The Overview Report.* ACM and IEEE Computer Society. ACM SIGCSE Bulletin (March 2006); https://doi.org/10.1145/1124706.1121482.
[Acm03] ACM (2010). *IS 2010 Curriculum Guidelines for Undergraduate Degree Programs in Information Systems,* Association for Computing Machinery (ACM) and Association for Information Systems (AIS); https://doi.org/10.1145/2593310.
[Acm04] ACM (2013). *Computer Science Curricula 2013: Curriculum Guidelines for Undergraduate Degree Programs in Computer Science,* Association for Computing Machinery and IEEE Computer Society; https://doi.org/10.1145/2534860.
[Acm05] ACM (2014). *Software Engineering Curricula 2014 Curriculum Guidelines for Undergraduate Degree Programs in Software; Engineering,* Association for Computing Machinery, and IEEE Computer Society; https://doi.org/10.1145/2594168.
[Acm06] ACM (2016). *Computer Engineering Curricula 2016 Curriculum Guidelines for Undergraduate Degree Programs in Computer Engineering*, Association for Computing Machinery (ACM) and IEEE Computer Society; https://doi.org/10.1145/3025098.
[Acm07] ACM (2017). *Information Technology Curricula 2017 Curriculum Guidelines for Baccalaureate Degree Programs in Information Technology,* Association for Computing Machinery (ACM) IEEE Computer Society (IEEE-CS); https://doi.org/10.1145/3173161.
[Acm08] ACM (2017). *Cybersecurity Curricula 2017 Curriculum Guidelines for Post-Secondary Degrees in Cybersecurity*. Association for Computing Machinery (ACM), IEEE Computer Society (IEEE-CS), Association for Information Systems Special Interest Group on Information Security and Privacy (AIS SIGSEC), International Federation for Information Processing Technical Committee on Information Security Education (IFIP WG 11.8); https://doi.org/10.1145/3184594.
[Acm09] ACM (2020). *Information Technology Curricular Guidance for Transfer Programs*; http://ccecc.acm.org/files/publications/IT-Transfer2020.pdf. Accessed 2020 Dec 6.
[Acm10] ACM (2017). *Computer Science Transfer Curriculum 2017 Computer Science Curricular Guidance for Associate-Degree Transfer Programs with Infused Cybersecurity,* The Association for Computing Machinery (ACM) Committee for Computing Education in Community Colleges (CCECC); http://dx.doi.org/10.1145/3108241.
[Acm11] ACM (2017) Topi, H., Karsten, H., Brown, S. A., Carvalho, J. A., Donnellan, B., Shen, J., et al. MSIS 2016: Global Competency Model for Graduate Degree Programs in *Information Systems*, 40, 1 (2017); https://doi.org/10.17705/1CAIS.04018 .
[Acm12] ACM (2001). *Computing Curricula 2001: Curriculum Guidelines for Undergraduate Degree Programs in Computer Science,* Association for Computing Machinery and IEEE Computer Society; https://www.acm.org/binaries/content/assets/education/curricula-recommendations/cc2001.pdf. Accessed 2019 Jun 26.
[Acm13] ACM (1968). Curriculum 68: Recommendations for academic programs in computer science: a report of the ACM curriculum committee on computer science. *Communications of the ACM* 11, 3 (1968), 151-197.
[Acm14] ACM. (1972). Ashenhurst, Robert L (Ed.). Curriculum recommendations for graduate professional programs in information systems. *Communications of the ACM,* 15, 5 (1972), 363-398.
[Acm15] ACM. (1973). Couger, J. (Ed.). Curriculum Recommendations for Undergraduate Programs in Information Systems, *Communications of the ACM*, 16, 12 (1973), 727-749.
[Acm16] ACM Education Policy Committee (2018) Lighting the Path: From Community College to Computing Careers; https://www.acm.org/binaries/content/assets/education/lighting-the-path-from-community-college-to-computing-careers.pdf. Accessed 2020 Dec 20.
[Acm17] ACM (2020) Cybersecurity Curricular Guidance for Associate-Degree Programs; http://ccecc.acm.org/files/publications/Cyber2yr2020.pdf. Accessed 2020 Dec 6.
[Acm18] ACM Europe Council. Informatics for All; https://urop.acm.org/i4all. Accessed 2021 Jan 5.
[Als1] Alsop, S. *Beyond Cartesian Dualism: Encountering Affect in the Teaching and Learning of Science*. Vol. 29. (Springer Science &

[Bai2] Bairaktarova, D. and Woodcock, A. Engineering Student's Ethical Awareness and Behavior: A New Motivational Model, *Scince and Engineering Ethics*, 23, 4 (2017), 1129–1157.

[Ban1] Wilder, C.R. and Ozgur, C.O. Business Analytics Curriculum for Undergraduate Majors, *INFORMS Transactions on Education* 15, 2 (2015), 180-187; https://doi.org/10.1287/ited.2014.0134.

[Bar1] Baron, R. M., and Kenny, D. A. The moderator-mediator variable distinction in social psychological research: Conceptual, strategic, and statistical considerations. *Journal of Personality and Social Psychology,* 51 (1986), 1173-1182; https://psych.wisc.edu/henriques/mediator.html. Accessed 2019 Sep 12.

[Bau1] Baumgartner, I. and Shankararaman, V. Actively linking learning outcomes and competencies to course design and delivery: experiences from an undergraduate Information Systems program in Singapore, in *Proceedings of the 2013 IEEE Global Engineering Education Conference*, (Berlin, Germany, 2013), 238–246.

[Bhe1] BHEF (2016) *Data Science and Analytics (DSA) Competency Map, The Business Higher Education Framework (BHEF),* version 1.0, produced in November 2016; https://s3.goeshow.com/dream/DataSummit/Data%20Summit%202018/BHEF_2016_DSA_competency_map_1.pdf. Accessed 2020 Jan 06.

[Bil1] Billett, S. Realising the educational worth of integrating work experiences in higher education, *Studies in Higher Educaton*, 34, 7 (2009), 827–843.

[Blo1] Bloom, B.S. (Ed.). Engelhart, M.D., Furst, E.J., Hill, W.H., Krathwohl, D.R. (1956). *Taxonomy of Educational Objectives, Handbook I: The Cognitive Domain*. (New York, David McKay Co. Inc.).

[Bls1] Bureau of Labor Statistics, U.S. Department of Labor, *Occupational Outlook Handbook,* 2016-17 Edition, Computer and Information Technology Occupations; https://www.bls.gov/ooh/computer-and-information-technology/home.htm. Accessed 2017 Dec 2.

[Bls2] Bureau of Labor Statistics, U.S. Department of Labor, *Occupational Outlook Handbook,* 2016-17 Edition, Computer and Information Technology Occupations; https://www.bls.gov/ooh/computer-and-information-technology/information-security-analysts.htm. Accessed 2017 Dec 2.

[Blu1] *The Blue Book Working Group of the Forum of Chinese Twenty-Experts on Computing Education in China.* (Computer Education for Sustainable Competence, China Higher Education Publishing, January 2019); in Chinese.

[Bol1] Bologna Declaration; http://www.ehea.info/cid100210/ministerial-conference-bologna-1999.html. Accessed 2020 Dec 6.

[Bol2] Bologna Working Group on Qualifications Frameworks. *A Framework for Qualifications of the European Higher Education Area*. (The Danish Ministry of Science, Technology and Innovation, 2005).

[Bur1] Burning Glass. (2017). *The Digital Edge: Middle-Skill Workers and Careers*; https://www.burning-glass.com/wp-content/uploads/Digital_Edge_report_2017_final.pdf. Accessed 2019 May 14.

[Cas1] Cassel, L., and Topi, H. (2016). *Strengthening Data Science Education Through Collaboration*; https://digital.library.villanova.edu/Item/vudl:622682. Accessed 2020 Dec 6.

[Cas2] Caspersen, M.E., Gal-Ezer, J., McGettrick, A., and Nardelli. E. Informatics as a fundamental discipline for the 21st century, *Communications of the ACM*, 62, 4 (2019), 58; https://cacm.acm.org/magazines/2019/4/235598-informatics-as-a-fundamental-discipline-for-the-21st-century/fulltext. Accessed 2020 Dec 6.

[Ccw1] CC2020 Project Website. https://www.cc2020.net/. Accessed 2019 May 9

[Cha1] Chambers D.W. and Gerrow, J.D. Manual for Developing and Formatting Competency Statements, *Journal of Dental Education*, 58, 5 (1994), 361–66.

[Che1] Juan Chen, Li Shen, Jianping Yin, Chunyuan Zhang. Design of Practical Experiences to Improve Student Understanding of Efficiency and Scalability Issues in High-Performance Computing (Poster). *SIGCSE '18: Proceedings of the 49th ACM Technical Symposium on Computer Science Education*. Pages 1090. February 2018. Baltimore, Maryland, USA. https://doi.org/10.1145/3159450.3162239.

[Che2] Juan Chen, Yingjun Cao, Linlin Du, Youwen Ouyang, and Li Shen. Improve Student Performance Using Moderated Two-Stage Projects. CompEd '19: Proceedings of the ACM Conference on Global Computing Education. Pages 201-207. May 2019, Chengdu, China. https://doi.org/10.1145/3300115.3309524.

[Che3] Juan Chen, John Impagliazzo, Li Shen. High-Performance Computing and Engineering Educational Development and Practice. 2020 IEEE Frontiers in Education Conference (FIE), Uppsala, Sweden, October 2020, pp. 1-8, https://doi.org/10.1109/FIE44824.2020.9274100.

[Chy5] Chyung, S., Stepich, D. and Cox, D. Building a Competency-Based Curriculum Architecture to Educate 21st-Century Business Practitioners, *The Journal of Education for Business,* 81, 6 (2006), 307-314.

[Cio] CIO Council. (2012). Clinger-Cohen core competencies and learning objectives. Washington, DC: CIO Council.

[Cis1] CISCO. Cisco Training and Certifications; https://www.cisco.com/c/en/us/training-events/training-certifications.html. Accessed 2020 Dec 6.

[Cla1] Classroom, The. (2019) Knowledge Based Learning; https://www.theclassroom.com/knowledge-based-learning-5403738.html. Accessed 2020 Dec 12.

[Cle1] Clear, T. Thinking Issues: Meeting Employers' eExpectations of DevOps Roles: Can Dispositions Be Taught? *Inroads,* 8, 2 (2017), 19–21; https://doi.org/10.1145/3078298.

[Cle2] Clear, A., Clear, T., Impagliazzo J. and Wang, P. (2020). From Knowledge-based to Competency-based Computing Education: Future Directions, submitted for Review.

[Col1] Collins Dictionary; http://www.collinsdictionary.com/dictionary/english/communication-skills. Accessed 2017 Dec 2.

[Com1] Computing at School, Computing in the national curriculum: A guide for primary teachers, 2013; https://www.computingatschool.org.uk/data/uploads/CASPrimaryComputing.pdf. Accessed 2020 Dec 12.

[Com2] Computing at School, Computing in the national curriculum: A guide for secondary teachers, 2014; https://www.computingatschool.org.uk/data/uploads/cas_secondary.pdf. Accessed 2020 Nov 22.

[Com3] CompTIA; https://www.comptia.org/. Accessed 2020 Nov 22

[Con1] Conceiving-Designing-Implementing-Operating (CDIO); http://www.cdio.org/. Accessed 2019 May 9.

[Cos1] Computer Society of IEEE. http://www.computer.org/. Accessed 2019 May 6.

[Cou1] Couger, J. (Ed.). Curriculum recommendations for Undergraduate programs in information systems. *Communications of the ACM* 16, 12 (1973), 727-749.

[Cpt1] CompTIA, *Information Technology (IT) Industry & Association*; https://www.comptia.org/. Accessed 2017 Dec 2.

[Cpt2] CompTIA, *Building Digital Organizations*, February 2017; https://www.comptia.org/content/research/building-digital-organizations.

Accessed 2020 Dec 23.
[CSf] CSforALL; Computer Science For All; https://wwwcsforall.org. Accessed 2021 Jan 21.
[Csp1] CSpathshala. www.cspathshala.org/. Accessed 2019 Aug 17.
[CSTA] Computer Science Teachers Association; https://www.csteachers.org. Accessed 2021 Jan 21.
[Cua1] Cuadros-Vargas, E. (2018). Escuela Profesional de Ciencia de la Computacion; https://education.spc.org.pe/Peru/CS-UTEC/Plan%202018/CS-UTEC-poster.pdf. Accessed 2019 May 9.
[Cua2] Cuadros-Vargas, E. (2018). 3.9 Compatibilidad de la carrera con relación a estandares internacionales. https://education.spc.org.pe/Peru/CS-UTEC/Plan%202018/3_9_Compatibilidad_carrera_.html. Accessed 2019 May 9.
[Dab1] Dabbagh, N., Benson, A. D., Denham, A., Joseph, R., Al-Freih, M., Zgheib, G., ... and Guo, Z. Massive open online courses. In *Learning technologies and globalization* (Springer, Cham, 2016), 9–13.
[Dat1] Data Science Task Force; http://dstf.acm.org/. Accessed 2020 Nov 22.
[Dat2] Data Science Draft Report 1. (2019). http://www.cs.williams.edu/~andrea/DSReportInitialFull.pdf. Accessed 2019 May 6.
[Dat3] Data Science Draft Report 2. (2019) http://dstf.acm.org/DSReportInitialFull.pdf. Accessed 2020 Jan 6.
[Dat4] DataUSA. 2019. https://datausa.io/profile/cip/computer-and-information-sciences-and-support-services. Accessed 2019 Aug 17.

[Har1] Harrow, A. *A Taxonomy of Psychomotor Domain: A Guide for Developing Behavioral Objectives*. (David McKay Co., Inc., New York, USA, 1972).

[Har2] Harvard; https://www.campusservices.harvard.edu/system/files/documents/1865/harvard_competency_dictionary_complete.pdf. Accessed 2020 Nov 25.

[Hel1] Helfert, M. Business informatics: An engineering perspective on information systems. *Journal of Information Technology Education: Research*, 7, 2008, 223–245.

[Icd1] ICD Translation; http://icdtranslation.com/skill-based-and-knowledge-based-online-training/. Accessed 2020 Nov 22.

[Iee1] IEEE (2017) Los Alamitos, Calif., Dec. 14, 2017; https://www.prnewswire.com/news-releases/top-10-technology-trends-for-2018-ieee-computer-society-predicts-the-future-of-tech-300571274.html. Accessed 2020 Nov 22.

[Iee2] The Informatics Europe and European Commission Joint Report on Industry-University cooperation; https://www.informatics-europe.org/news/544-bridging-the-digital-talent-gap-towards-successful-industry-university-partnerships.

[Iee3] IEEE Computer Society (2014) Software Engineering Competency Model: Version 1.0, SWECOM 2014.

[Imp1] Impagliazzo, John. (2018) The Role of Competency and the Future of Computer Education. *Communications of the China Computer Federation*; 14, 9 (September 2018); in Chinese.

[Imp2] Impagliazzo, J., et al. Developing an Overview of Computing/Engineering Curricula via the CC2020 Project. In *Proc. of the IEEE EduNine Conference (2018)*. IEEE Education Society; https://ieeexplore.ieee.org/document/8450965. Accessed 2020 Nov 22.

[Imp3] Impagliazzo J., Parrish, A., and Clear, A. Innovative Computing Curricula and the CC2020 Project. In *Proc. of the Frontiers in Education (FIE) Conference (2018)*; https://www.computer.org/csdl/proceedings-article/fie/2018/08658622/18j95f9XE7m. Accessed 2020 Nov 22.

[Ind1] India 1; https://www.ugc.ac.in. Accessed 2020 Nov 22.

[Ind2] India 2; www.naac.gov.in. Accessed 2020 Nov 22.

[Ind3] India 3; www.aicte-india.org. Accessed 2020 Nov 22.

[Ind4] India 4; www.nbaind.org. Accessed 2020 Nov 22.

[Ins1] Institut de France, L'Académie des Sciences, "L'enseignement de l'informatique en France – Il est urgent de ne plus attendre," Mai 2013; https://www.academie-sciences.fr/pdf/rapport/rads_0513.pdf. Accessed 2020 Nov 22.

[Inv1] Investopedia; http://www.investopedia.com/terms/s/soft-skills.asp. Accessed 2017 Dec 2.

[Ipa1] Information-technology Promotion Agency – Japan (IPA). IT Human Resources Development: i Competency Dictionary; https://www.ipa.go.jp/english/humandev/icd.html. Accessed 2019 Aug 18.

[ITec] Info-Tech Research Group, 2020 Tech Trend Report; https://www.infotech.com/research/ss/2020-tech-trend-report. Accessed 2021 Feb 15.

[Kak1] Kakeshita, T. National Survey of Japanese Universities on Computing Education: Analysis of Departments Majored in Computing Discipline, *Olympiads in Informatics*, 12 (2018), 69–84; https://ioinformatics.org/journal/v12_2018_69_84.pdf. Accessed 2019 June 29.

[Ken1] Kennedy, D., Hyland, A., and Ryan, N. Learning outcomes and competences, *Introducing Bologna Objectives and Tools*, (2009), 2–3.

[Kpm1] KPMG: Technology Trends Index USA; http://technologytrendsindex.kpmg.com/. Accessed 2020 Nov 22.

[Kra1] Krathwohl, D.R., Bloom, B.S., and Bertram, B.M. Taxonomy of Educational Objectives, the Classification of Educational Goals. Handbook II: Affective Domain. (David McKay Co., Inc., New York, USA,1973).

[Kra2] Kramer, M., Hubwieser, P. and Brinda, T.A Competency Structure Model of Object-Oriented Programming, *2016 International Conference on Learning and Teaching in Computing and Engineering (LaTICE)*, 2016, 1–8.

[Kuh1] Kuh, G.D. The national survey of student engagement: Conceptual and empirical foundations. *New Directions for Institutional Research*, 141 (2009), 5–20.

[Len1] Lenburg, C.B. The framework, concepts and methods of the competency outcomes and performance assessment (COPA) model. *Online Journal of Issues in Nursing*, 4, 2 (1999), 1–12.

[Lin1] Lindenwood University Makerspace Lab, https://www.lindenwood.edu/about/news/details/converged-media-lab-makerspace-to-open-this-fall/. Accessed 2019 Aug 17.

[Liu11 Liu, K., *Semiotics in Information Systems Engineering*, Cambridge University Press, Cambridge, U.K, 2000.

[Loc1] Lockoff, J., Wegewiis, B., Durkin, K., Wagenaar, R., Gonzales, J., Isaacs, A. K., Donà dalle Rose, L.F., and Gobbi, M. (Eds.) (2010). A Tuning guide to formulating degree programme profiles: Including programme competences and programme learning outcomes. Bilbao, Spain: University of Deusto; https://www.unideusto.org/tuningeu/publications/290-tuning-guide-to-formulating-degree-programme-profiles.html. Accessed 2020 Nov 22.

[Lun1] Lunt, B.M., Ekstrom, J.J., Gorka, S., Hislop, G., Kamali, R., Lawson, E., LeBlanc, R., Miller, J. and Reichgelt, H. 2008. Curriculum Guidelines for Undergraduate Degree Programs in Information Technology; https://dl.acm.org/citation.cfm?id=2593311. Accessed 2017 Dec 2

[Lyn1] Lynch, D.R., Russell, J.S., Evans, J.C., and Sutterer. K.C. Beyond the Cognitive: The Affective Domain, Values, and the Achievement of the Vision. *Journal of Professional Issues in Engineering Education and Practice* 135, 1 (2009): 47–56.

[Mar1] Marshall, L. A comparison of the core aspects of the ACM/IEEE Computer Science Curriculum 2013 Strawman report with the specified core of CC2001 and CS2008 Review. *Computer Science Education Research Conference. (2012)* (CSERC 2012, ACM, 2012), 29–34.

[Mar2] Marshall, L. (2014) A graph-based framework for comparing curricula. Ph.D. thesis, University of Pretoria, South Africa.

[Mar3] Marshall, L. (2017) A Topic-Level Comparison of the ACM/IEEE CS Curriculum Volumes. Liebenberg J., Gruner S. (eds) ICT Education, *Proceedings of the 46th Annual Conference of the Southern African Computer Lecturers' Association on ICT Education, SACLA 2017,* held in Magaliesburg, South Africa, in July 2017. *Communications in Computer and Information Science*, vol. 730 (2017), 309-324.

[Mar4] Markus, M.L. and Topi, H. (2015) *Big Data, Big Decisions for Science, Society, and Business: Report on a Research Agenda Setting Workshop. Technical Report*. National Science Foundation (NSF), USA; https://dl.acm.org/doi/book/10.5555/2849516. Accessed 2020 Nov 23.

[Mer1] Merriam-Webster; https://www.merriam-webster.com/dictionary/competency. Accessed 2019 June 3.

[Mer2] Merriam-Webster; https://www.merriam-webster.com/dictionary/disposition/. Accessed 2019 Aug 30.

[Mer3] Merriam-Webster; https://www.merriam-webster.com/dictionary/learning. Accessed 2020 Dec 12.

[Mer4] Merriam-Webster; https://www.merriam-webster.com/dictionary/knowledge?src=search-dict-box#synonyms. Accessed 2020 Dec 12.

[Mic1] Microsoft Certifications; https://docs.microsoft.com/en-us/learn/certifications/. Accessed 2020 Nov 22.

[Min1] Ministry of Education, New Zealand (2019). What are makerspaces? http://elearning.tki.org.nz/Teaching/Future-focused-learning/Makerspaces. Accessed 2019 Jul 3.

[Nas1] National Academies of Science (NAS) (2018). Envisioning the Data Science Discipline: The Undergraduate Perspective: Interim Report;| http://doi.org/10.17226/24886.

[Nas2] National Academies of Sciences, Engineering, and Medicine. 2018. Assessing and Responding to the Growth of Computer Science Undergraduate Enrollments. Washington, DC: The National Academies Press; https://doi.org/10.17226/24926.

[New1] University of Newcastle, Information Technology; https://www.newcastle.edu.au/degrees/bachelor-of-information-technology. Accessed 2020 Nov 22.

[Nov1] Novinson, M. *Top 15 Moneymaking Certifications for 2017*. CRN, The Channel Company; http://www.crn.com/slide-shows/managed-services/300080027/top-15-moneymaking-certifications-for-2016.htm. Accessed 2017 Dec 2.

[Nrc1]

[Sto1] Stoof, A., Martens, R L., and Van Merriënboer, J.J. *What is competence? A constructivist approach as a way out of confusion*, (Onderwijs Res. Dagen ORD Leiden Neth., 2000).
[Tak1] Takada, S., Cuadros-Vargas, E., Impagliazzo, J. et al. Toward the visual understanding of computing curricula. *Educ Inf Technol* 25,! (2020),!4231–4270; https://doi.org/10.1007/s10639-020-10127-1.
[Tan1] Tang, C., Hawthorne, E.K., Tucker, C.S., Cuadros-Vargas, E., Cukierman, D., Simon, Zhang, M. (2016). Global Perspectives on the Role of Two-Year/Technical/Junior Colleges in Computing Education. 21st ACM Conference on Innovation and Technology in Computer Science Education (ITiCSE '16), Arequipa, Peru, 204-205.
[Tec1] TechTarget; https://whatis.techtarget.com/definition/concept-map. Accessed 2020 Nov 23.
[Ted1] Tedre, M. *The Science of Computing: Shaping a Discipline*. (Boca Raton, CRC Press / Taylor & Francis, 2015).
[Ted2] Tedre, M., and Sutinen, E. (2008). Three traditions of computing: what educators should know. *Computer Science Education*, 18, 3 (2008), 153–170.
[Tes1] Tes; https://www.tes.com/teaching-resource/kbl-knowledge-based-learning-11361440. Accessed 2020 Nov 23.
[The1] The Essential Facts; https://www.theesa.com/wp-content/uploads/2019/05/2019-Essential-Facts-About-the-Computer-and-Video-Game-Industry.pdf. Accessed 2020 Dec 12.
[Tre1] Trevelyan, J. *The Making of an Expert Engineer.* (Taylor and Francis Group, Ltd., 2014).
[Top1] Topi, H. Information Systems in CC2020: Comparing Key Structural Elements of Curriculum Recommendations in Computing. *2017 Proceedings of SIGED: IAIM Conference. Association for Information Systems Electronic Library (AISeL)*. https://aisel.aisnet.org/siged2017/9. Accessed 2020 Dec 8.
[Tuc1] Tucker, A.B. Computing curricula 1991. *Communications of the ACM*, 34, 6 (1991), 68-84; http://dl.acm.org/citation.cfm?doid=103701.103710.
[Van1] Van der Klink, M. and Boon, J. The investigation of competencies within professional domains, *Hum. Res. Dev. Int*., 5, 4 (2002), 411–424.
[Van2] Van der Klink, M., Boon, J. and Schlusmans, K. Competences and Vocational Higher Education: Now and in Future, Eur. J. *Vocat. Train*., 40, 1 (2007), 67–82.
[Vor1] Voorhees, R. A., and Bedard-Voorhees, A. Principles for Competency-based education. *Instructional-Design Theories and Models*, IV, (Routledge, 2016), 49-80.
[Wag1] Waguespack, L., Babb, J. S. and Yates, D. Triangulating Coding Bootcamps in IS Education: Bootleg Education or Disruptive Innovation? *Information Systems Education Journal*, 16, 6 (2018), 48.
[Wag5] Waguespack, L. et al., Adopting Competency Mindful of Professionalism in Baccalaureate Computing Curricula, EDSIGCON 2019, 2019; https://www.researchgate.net/publication/336945198_Adopting_Competency_Mindful_of_Professionalism_in_Baccalaureate_Computing_Curricula. Accessed 2020 Dec 12.
[Web1] Weber, H. (2017). The New Virtues of Engineering and the Need for Change in the Engineering Curriculum; https://www.researchgate.net/publication/325924314_The_New_Virtues_of_Engineering_and_the_Need_for_Change_in_the_Engineering_Curriculum. Accessed 28 Dec 2020.
[Wef1] World Economic Forum (2020). Jobs of Tomorrow: Mapping Opportunity in the New Economy; https://www.weforum.org/reports/jobs-of-tomorrow-mapping-opportunity-in-the-new-economy. Accessed 2020 Nov 23.
[Wei1] Weinert, F.E. Definition and selection of competencies: Concepts of Competence, Organization for Economic Co-operation and Development, 1999.
[Wig1] Wiggins, G., McTighe, J., and Ebrary, I. *Understanding by design* (Expanded 2nd edition). (Alexandria, VA, Association for Supervision and Curriculum Development, 2005).
[Wig2] Wiggins, G., and McTighe, J. *The Understanding by Design Guide to Creating High-Quality Units.* (Alexandria, VA, Association for Supervision and Curriculum Development, 2011).
[Wil1] Willcox, K., and Huang, L. Mapping the CDIO Curriculum with Network Models. CDIO, In *13th International CDIO Conference* (2017).
[Zuc1] Zucker, R. ViCurriAS: a curriculum visualization tool for faculty, advisors, and students. In *J. Comput. Sci. Coll.* 25, 2 (2009), 138–145.

R2: Additional References not Cited

- ACM and AIS. (2017c). MSIS 2016: Global Competency Model for Graduate Degree Programs in Information Systems. https://www.acm.org/binaries/content/assets/education/msis2016.pdf. Accessed 2019 May 14.
- ACM (2019). *Computing Competencies for Undergraduate Data Science Curricula* (Initial Draft). http://www.cs.williams.edu/~andrea/DSReportInitialFull.pdf. Accessed 2019 May 9.
- Ashenhurst, R.L. Curriculum Recommendations for Graduate Professional Programs in Information Systems. *Communications of the ACM*, 15, 5 (1972), 364–398.
- Atchison, W.F. Computer Education, Past, Present, and Future. *ACM SIGCSE Bulletin*, 13, 4 (1981), 2–6. http://doi.org/10.1145/989306.989307.
- Barr, V. and Stephenson C. Bringing Computational Thinking to K-12: What Is Involved and What Is the Role of the Computer Science Education Community?" *ACM Inroads* 2, 1 (2011), 48; https://dl.acm.org/doi/10.1145/1929887.1929905.
- Biggs, J.B. Approaches to the enhancement of tertiary teaching. *Higher Education Research and Development* 8, 1 (1989), 7–25.
- Cassel, L.N., Sloan, R.H., Davies, G., Topi, H. and McGettrick, A. The Computing Ontology Project: The Computing Education Application. *SIGCSE Bull*. 39, 1 (2007), 519–20.
- Clear, A., Parrish, A., van der Veer, G.C., Zhang, M. CC2020: A Vision on Computing Curricula. Panel, *Proceedings of the 2017 ACM SIGCSE Technical Symposium*, DOI: 10.1145/3017680.3017690.

- Couger, J.D. Curriculum recommendations for undergraduate programs in information systems. *Communications of the ACM*, 16, 12 (1973), 727–749; http://doi.org/10.1145/362552.362554.
- Conte, S.D., Hamblen, J.W., Kehl, W.B., Navarro, S.O., Rheinboldt, W.C., Young, D.M., Jr, and Atchinson, W.F. An Undergraduate Program in Computer Science—Preliminary Recommendations. *Communications of the ACM,* 8, 9 (1965), 543–552; http://doi.org/10.1145/365559.366069.
- Council of Chief State School Officers. 2013. Knowledge, Skills, and Dispositions: The Innovation Lab Network State Framework for College, Career, and Citizenship Readiness, and Implications for State Policy; https://files.eric.ed.gov/fulltext/ED542708.pdf. Accessed 2020 Nov 24.
- Couger, J.D., Davis, G.B., Dologite, D.G., Feinstein, D.L., Gorgone, J.T., Jenkins, A.M., et al. IS'95: Guideline for undergraduate IS curriculum. *Mis Quarterly*, (1995) 341–359.
- Davis, G.B., Gorgone, J.T., Couger, J., Feinstein, D.L., and Longenecker, H.E., Jr. IS'97: model curriculum and guidelines for undergraduate degree programs in information systems. *ACM SIGMIS Database*, 28,1 (1996), 101–194.
- Davis, M. *The Universal Computer: The Road from Leibniz to Turing*. (New York, A K Peters/CRC Press, 2011).
- Edström, Kristina. 2017. "Exploring the Dual Nature of Engineering Education: Opportunities and Challenges in Integrating the Academic and Professional Aspects in the Curriculum." Edited by Anette Kolmos. PhD in Technology and Learning, Stockholm, Sweden: KTH Royal Institute of Technology; http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-217315. Accessed 2020 Nov 24.
- Frezza, S.T., Clear, A. and Vichare, A.M. 2018. Voices on the Core of Computing. In *2018 IEEE Frontiers in Education Conference (FIE);* http://doi.org/10.1109/FIE.2018.8658484.
- Fuller, U. and Keim, B. Should We Assess Our Students' Attitudes? In *Proceedings of the Seventh Baltic Sea Conference on Computing Education Research*, 187–90. (Koli National Park, Finland, ACM, 2007).
- Fincher, S., and Petre, M. Computer Science Education Research. (CRC Press, 2014).
- Forsythe, G.E. (1967). A University's Educational Program in Computer Science. *Communications of the ACM*, 10, 1 (1967), 3–11; http://doi.org/10.1145/363018.363038.
- Gibbs, G. Teaching students to learn: A student-centered approach. (Open University Press, 1981).
- Gorgone, J. and Gray, P. MSIS 2000: model curriculum and guidelines for graduate degree programs in information systems. *Communications of the AIS*, 3, 1 (2000), 1.
- Gorgone, J. T., Davis, G. B., Valacich, J. S., Topi, H., Feinstein, D. L., and Longenecker, H. E. IS 2002 model curriculum and guideli1es for undergraduate degree programs in information systems. *Communications of the AIS*, 11, 1 (2002).
- Gorgone, J.T., Gray, P., Stohr, E.A., Valacich, J.S., and Wigand, R.T. MSIS 2006: Model Curriculum and Guidelines for Graduate ¥

- Shiveley, J., and Misco, T, 'But How Do I Know About Their Attitudes and Beliefs?': A Four-Step Process for Integrating and Assessing Dispositions in Teacher Education. *The Clearing House: A Journal of Educational Strategies, Issues and Ideas* 83, 1 (2010), 9–14.
- Sicilia, M-A. How Should Transversal Competence Be Introduced in Computing Education? *ACM SIGCSE Bulletin* 41, 4 (2010), 95–98.
- Smith, B. *The Future Computed: Artificial Intelligence and its Role in Society*. (Microsoft Corporations, 2018).
- Spc1; https://education.spc.org.pe/Peru/CS-SPC/Plan2021/docs/CS-SPC-poster-EN.pdf. Accessed 2020 Dec 6.
- Spc1; https://education.spc.org.pe/Peru/CS-SPC/Plan2021/3_9_Compatibilidad_carrera_.html. Accessed 2020 Dec 6.
- Spector, A.Z. (2017) Changing Nature of Computer Science and Its Impact on Undergraduate Education; http://sites.nationalacademies.org/cs/groups/cstbsite/documents/webpage/cstb_173998.pdf. Accessed 2020 Dec 8.
- Tedre, M., Simon, and Malmi, L. Changing aims of computing education: a historical survey. *Computer Science Education*, 28, 2 (2018), 158–186.
- Topi, H., Valacich, J.S., Wright, R.T., Kaiser, K., Nunamaker, J.F., Sipior, J.C., and de Vreede, G.J. IS 2010: Curriculum Guidelines for Undergraduate Degree Programs in Information Systems. *Communications of the Association for Information Systems*, *26*, 18 (2010); http://aisel.aisnet.org/cais/vol26/iss1/18/. Accessed 2020 Nov 25.
- Topi, H., Valacich, J.S., Wright, R.T., Kaiser, K., Nunamaker, J.F., Sipior, J.C., and de Vreede, G.J. IS 2010: Curriculum Guidelines for Undergraduate Degree Programs in Information Systems. *Communications of the Association for Information Systems*, 26, 1 (2010); http://aisel.aisnet.org/cais/vol26/iss1/18/. Accessed 2020 Nov 25.
- The Royal Academy of Engil 25.

Association f