UNLV

UNIVERSITY OF NEVADA, LAS VEGAS

Program Review Self-Study

Program Reviewed: Computer Science

Degrees: M.S. and Ph.D.

Program Chair or Director: Laxmi Gewali

Dean: Rama Venkat

Date of Report: 03/20/2017

GENERAL INSTRUCTIONS

1. Please provide Faculty CVs as a single electronic file (PDF preferred) or on a thumb drive *for the external reviewers*.
2. **Please complete the program review self-study using this template**.
3. If this review is covering several degree levels, please be sure to address ***each level*** in your responses to the questions.
4. Contacts for questions:

- Chair of the Faculty Senate Program Review Committee found here: http://facultysenate.unlv.edu/committees/program-review or the Chair of the Graduate College Program Review Committee found here: http://www.unlv.edu/graduatecollege/program-review-committee
- Gail Griffin in the Office of the Vice Provost for Academic Affairs: Gail Griffin, gail.griffin@unlv.edu, 702-895-

I. Program Description

A. College/Department/Program

1. College or School: Engineering
2. Unit: Computer Science - Web address: http://www.unlv.edu/cs
3. Program(s) being reviewed: Master of Science in Computer Science, Ph.D. in Computer Science
   a. Degrees and their abbreviations:

   C.S. M.S. and C.S. Ph.D.

B. Primary individual completing this worksheet

1. Name: Laxmi Gewali
2. Title: Department Chair
3. Date of self-study: November 30, 2016
4. Campus Phone: 702-895-4028
5. Mail Stop: 4019
6. E-mail: Laxmi.Gewali@unlv.edu
7. Fax Number: 702-895-2639

C. Other faculty involved in writing this report: Wolfgang Bein, Ajoy K. Datta, Yoohwan Kim, John Minor

D. Please provide as Appendix 1 the most recent catalog description(s) of the program(s):

1. Is the description correct? If not, what needs to be changed?

   For the M.S. catalogue the following changes must be made:

   1. The unit was organized as a school for a number of years, before the elimination of the B.S., M.S, and Ph.D. in Informatics in 2012 due to budget cuts in Nevada. There are three instances of "School", which should be "Department".
   2. On page 1, item 3, the word "Another" in “Another set of official transcripts” should be removed as the word “another” is simply confusing to the applicant.
   3. On page 1, item 5, opening parenthesis is missing before "CS326".

   For the Ph.D. catalogue the following changes must be made:

   1. There are four instances of "School", which should be "Department" as explained above.
   2. On page 3, item 8. "Dissertation proposal defense" should be changed to "Dissertation defense" to avoid confusion. (Similar changes on page 5, item 8; page 7, item 9; page 9, item 9).

3. On page 3, item 11, the value "2.5 years" must be changed to "two years". (Similar changes on page 5, item 11; page 7, item 12; page 10, item12).

II. Centrality to Mission

A. Department/Program Mission

What is the program's mission statement (or the department's if the program does not have one)?

The department's mission statement is

The mission of the Department of Computer Science is to educate future computer scientists in academic programs that are competitive with those of the best schools in the country.

The statement is guided by a recognition that computer science is a broad science encompassing many areas with theoretical and applied components.

B. Department/Program Mission Alignment

Briefly describe how this program is aligned to the mission of the University as described in the most recent mission statement, UNLV Mission http://www.unlv.edu/about/mission, and how it supports achievement of the institution's mission:

UNLV's mission statement – recently enhanced by UNLV's plan to be transformed into a top tier institution – reads:

UNLV's diverse faculty, students, staff, and alumni promote community well-being and individual achievement through education, research, scholarship, creative activities, and clinical services. We stimulate economic development and diversification, foster a climate of innovation, promote health, and enrich the cultural vitality of the communities that we serve.

The graduate CS graduate programs align well with UNLV's mission statement. A number of indicators are listed below:

Indicator: Scholarly research and creative activity, Carnegie Classification.

In 2016, the department had 8 journal publications and 23 publications in peer-reviewed conferences. More than half of these were co-authored with students in the graduate programs. In the same year, research expenditures were $321,028, which is trending up from previous years. (In 2016 the dollar amount of proposals submitted was $6,778,182.) Collaborations exist with all departments in the college and beyond. Further there are international collaborations, specifically with Japan and Germany, to which students of our programs contribute. The Center for Information Technology and Algorithms is a multidisciplinary unit (one director from CS, the other from EE), operated by faculty, staff and students at UNLV. It facilitates national and international collaborations, and interdisciplinary work across our own campus and with other academic entities. For example, in 2014 and 2015, one of our Ph.D. students joined in on two research trips to the University of Electro-Communications, Tokyo.

Journal publications in CS have high citation counts. For example; in the area of computer science theory Dr. Lawrence Larmore has 2364 citations with an h-index of 25; in cybersecurity Dr. Hal Berghel (an ACM fellow) is a regular contributor to the widely read IEEE Computer column; in the area of big data science, Dr. Justin Zhan currently oversees a multi-million-dollar grant portfolio, and in the area of accessibility, Dr. Andreas Steffik received the White House Champion of Change Award from the Executive Office of the President in 2016.

Indicator: Student achievement of learning outcomes.

The graduate faculty meets every semester as a whole to discuss issues of academic standards and student achievement. The programs are regularly assessed through UNLV's Office of Academic Assessment. The previous report was completed in 2015 and is attached in the appendix.

Indicator: Placement into preferred employment or post-graduate educational opportunities.

Career opportunities in Computer Science abound: Graduates find positions as programmer, scientific programmer, system programmer, computer control programmer, interface programmer/designer, client/server software developer, computer graphics specialist, software publicist, software engineer, quality assurance engineer, telecommunications planner, artificial intelligence expert, modeling/simulation analyst, database designer/administrator, market research analyst, competitive intelligence analyst, medical communication specialist, healthcare information technology specialist, operations research analyst, and (in the case of Ph.D.) as postdoc or assistant professor in academia.

The College of Engineering Placement Office is operated by a full time administrative line. Ms. Marian Mason is in charge of assisting our students with internships and job searches. Ms. Mason also organizes TechConnect where we connect Computer Science, MIS and IT students with companies that hire them. The recent edition of TechConnect featured two panel discussions:

- "Get A Job" - where students heard from hiring professionals on what their hiring needs were.
- "Job Experiences" – where alumni discussed their job experiences.

Indicator: Student, faculty, and staff diversity, including maintaining UNLV's Minority Serving Institution (MSI) status and Hispanic Serving Institution (HSI) status.

It is well documented in the literature that there is a lack of women in computer science, even when compared to other engineering disciplines. The department actively supports the "Girls Who Code" program. Currently roughly 25% of our students are women – a respectable number when compared to peer institutions.

The program does not have a particularly high number of minority students, and among the M.S. students more than two thirds of our students are international students. Undergraduate students in C.S. (where there is a much more diverse student population) routinely go into the favorable job market rather than pursuing graduate studies. Thus, if more undergraduate students entered the CS graduate programs the situation would change. The B.S. to Ph.D. program (which makes it possible to get into the Ph.D. program without first obtaining a Master's degree) is one way this is currently addressed.

Indicator: Physical Infrastructure

Office space is currently about 4,000 sq. ft. and lab space is about the same. (Exact numbers are listed further below.) There are numerous laboratories available to students, notably TBE- B 346 for the C.S. graduate students. However, space distribution is inadequate, as outlined further below.

C. Core Themes

Briefly describe how this program supports UNLV's Core Themes (the core themes can be found at http://www.unlv.edu/about/mission):

Core Theme 1: Promote Student Learning and Success

The Computer Science graduate faculty has developed a foundational and contemporary curriculum which is in line with those of the best schools in the country. We offer rigorous foundation courses such as courses in Theory of Computation (CS 718), Advanced Analysis of Algorithms (CS 715), or Advanced Automata and Formal Languages (CS 719). Computer Science, after all, is not foremost about vocational training, and yet, the graduate faculty regularly interfaces with stakeholders, especially alumni and the Industrial Advisory Board (IAD) to ensure the programs are in line with demands of industry and society at large. All departments' endeavors, hiring decisions, and curricular enhancements are very much guided in this way.

Learning Outcomes for the M.S. program:

1. Acquire in-depth knowledge of specialized areas and advanced topics in computer science.
2. Independently analyze, design, and implement an innovative computer application or research project.
3. Prepare a final project or research report on the solution to a computer-related problem.
4. Present the results of their research orally.

Learning Outcomes for the Ph.D. program:

1. Exhibit a breadth of knowledge in the areas of algorithms, programming languages and compilers, theory, operating systems, and computer architecture.
2. Exhibit a depth of knowledge in at least one specialized area of computer science.
3. Conduct a thorough literature survey on a research topic.
4. Prepare a research paper that is publishable.
5. Present the research results in an oral defense.

Core Theme 2: Advance Research, Scholarship, and Creative Activity

As mentioned in Section B, the department nurtures a culture of intellectual enquiry and is a catalyst for scholarship and creative activity. Teaching load reassignments of (2-1) instead of (2-2) are given for grant preparation and exceptional research efforts. Graduate students are encouraged to publish with their advisors and travel grants are made available by the department for students to attend national or even international conferences.

Core Theme 3: Foster a Diverse Campus Population and Engagement with the Community.

The department interfaces with industry leaders though the Industrial Advisory Board (IAD) to stay on top of current trends and demands. The department reaches out to local high schools during regular recruitment periods. International recruitment efforts are also under way through collaboration/ties with universities in Germany, India, Japan, and France.

D. Excellence

List and briefly describe five highlights or areas of excellence of the program:

1. The department is especially strong in the area of algorithmic theory, as there are four faculty members in this category. All of these faculty members have made highly cited contributions to computer science theory. One member of this group, Dr. Wolfgang Bein, is a co-director of the Center for Information Technology and Algorithms, a multidisciplinary unit, and operated by faculty, staff and students at UNLV. It facilitates national and international collaborations, interdisciplinary work across our own campus and with other academic entities.

   The language of science increasingly is no longer mathematics but algorithmics. This is also foundational for STEM education in general. It is well documented that CS theory generally is not funded at the level of more applied areas, and yet the department has increased funding significantly.

2. With the recent hire of Dr. Justin Zhan there has been a boost in funding in the area of big data analytics. This area is interdisciplinary in nature with significant overlap with health science, information security, social computing, and sustainability. Dr. Zhan's research program has secured over $3 million in funding for the next 5 years.

3. The department has strengths in software engineering, program languages, compilers, and accessibility. These areas are especially requested by local industry, and have a significant impact on graduates' employability. Three faculty members are working in these concentrations. Notably, Dr. Andreas Steffik recently received the White House Champion of Change Award from the Executive Office of the President.

4. Three faculty members are in the area of cyber-security and computer forensics. As with software engineering, this area is in high demand by local industry as well as nationally. One member, Dr. Hal Berghel, is a regular contributor to IEEE Computer and an ACM Fellow – one of the highest honors in the area of Computer Science.

5. Dr. Fatma Nazos conducts research in human-computer interface and machine learning — another highly requested area. Courses fulfill an important component listed in the ACM curriculum. Dr. Nazos' work has been funded through the NASA Mentor-Protege Program, the U.S. National Park Service, and the U.S. Forest Service.

III. External Demand for Program

A. Stakeholders

1. Who are the main local and regional stakeholders of your educational programs, i.e., employers and entities benefiting from these programs, hiring the graduates, or admitting them to graduate and/or professional programs?

   The market for Master's level jobs is national (rather than local); after graduation many alumni relocate to California and Texas, or the East Coast — especially our international students. Demand for CS graduates by local gaming companies is also healthy, but it is imperative that Nevada further diversify its industrial base. Academic job searches (for CS Ph.D. graduates) are usually national as well.

Local employers include:

- Zappos
- NVEnergy
- Southwest Gas
- National Security Technologies (NSTec)
- Department of Motor Vehicles
- International Game Technology (IGT)
- Scientific Gaming
- MGM
- Sands
- UNLV, Office of OIT

2. What are specific stakeholder needs for graduates?

Input from the Industrial Advisory Board (IAD) shows that local stakeholders have programming and software development skills as their top priority.

B. Needs for Graduates and Future Plans

1. What are the anticipated needs for program graduates over the next 3-5 years? Please cite sources of information.

The IEEE Computer Society is the largest professional society in computer science. According to its IEEE CS 2022 Report (https://www.computer.org/cms/ComputingNow/2022Report.pdf), there are 23 potential technologies that could change the landscape of computer science and industry by the year 2022. Among them, the following areas are actively researched and taught in our department:

- Cybersecurity — Rank 1
- Sustainability — Rank 3
- Multicore Computing — Rank 9
- Networking — Rank 11
- High-Performance Computing — Rank 13
- Cloud Computing — Rank 14
- Big Data and Analytics — Rank 18
- Machine Learning and Intelligent Systems — Rank 19
- Computer Vision and Pattern Recognition — Rank 20
- Computational Biology and Bioinformatics — Rank 21

On a micro-level alumni surveys indicate that there is a need to incorporate emerging programming languages such as Python, Ruby on Rails, or Google-Go.

2. What changes to the program will those require?

- Change in Information Technology is fast-paced. It must be noted that graduate education in computer science should provide a strong foundation that can stand the test of time and on which future CS professionals can build, not solely vocational training in a contemporary set of skills.

- Adjustments to individual courses, and revisions to the curriculum are routinely undertaken. Faculty members adapt their courses to keep up with the fast pace of change and the resulting shift in employer expectations.

- Currently, course offerings also include independent study courses (CS 690) and special topics courses (CS 789). These can be used to include contemporary topics in the curriculum.

- The course CS 140 - Computing Languages - has been used to offer specialized programming language training (for example, "Ruby on Rails" is currently offered).

- With the shift towards renewable energy sources, new challenges exist in designing an effective, dependable and secure power infrastructure. Nevada with an abundance of sunshine, various other natural energy resources and a low population density, could take a leadership position. One of the major research themes pursued by the university in an interdisciplinary manner is the theme 'sustainability" and the Center for Information Technology and the Center for Energy Research have begun collaborations that benefit from the synergies that currently exit. New CS courses and a faculty line in the area of green computing would benefit all engineering departments and the UNLV community as a whole. A faculty line in this area will be a catalyst for fresh algorithmic approaches across different disciplines.

C. Success of Graduates

1. What steps does the department take to facilitate the success of graduates (e.g., internships, career fairs, employment talks, etc.)?

   The department strongly encourages students to take internships. The College of Engineering Placement Office operated by a full time administrative line (Ms. Marian Mason) is in charge of assisting our students with internships and job search. Students also take internship from out-of-state organizations given the lack of diversification of the Nevada economy.

   The department has been hosting a career fair, called TechConnect, successfully in collaboration with the Industrial Advisory Board (IAD) for nearly a decade. Company representatives from both technical and management sides are invited to various classes to introduce their work and job skill needs.

   The department has provided off-campus research experiences and conference participation at the major conferences, which has provided networking opportunities. For example, one of our Ph.D. students took part in extensive NSF funded research experiences at the University of Electro-Communications in Japan and one of our recent Master's students was sponsored by the department to participate in the prestigious European Symposium on Algorithms held in Slovania.

2. Discuss the placements of recent graduates:
   It is hard to track the careers of our graduates, since careers in IT are highly dynamic. As mentioned above the market for Master's level jobs is national rather than local. Below are examples of the placement of recent graduates.

   - Ashkaan Kouhpaenejad - Decision Ready Solutions
   - Erik. H. Tribou - Vegas.com
   - George Oprean - VMWare

- Mikolaj Slomka - Officer School in the U.S. Navy
- Chris Jones and Jason Hurt - IBM
- Matthew Sowders - Amazon.com
- Michael Jones - GES
- Yoga Jaideep Darapuneni - KPMG US
- Vivek Gudibande - Scientific Games
- Sabbir Manandhar - TransPerfect/Worldlingo
- Ryan Carr - Barclaycard
- Revanth Pamballa - Intel
- Akshat Pampanwar, IBM Watson Center
- James Oravec - Allegiant Travel Company
- Rushikesh Veni - Amazon
- Rohit Raj - UNLV
- Megha Sairam Darapuneni – Perficient Inc.
-

3. If the department or program does not have placement information on graduates, what is the plan to implement gathering that information?

   Placement information is currently kept through the individual advisors. The department is in the process of forming a graduate placement committee and will keep a centralized database of placement history.

4. Do placements match stakeholder needs as identified above in A of this section? In the main, yes.

5. If not, please explain.

6. Does the program assess whether the graduates are meeting employer's needs?[1]

   Beyond the usual program assessment (the most recent assessment report is in the appendix) the department meets with the Industrial Advisory Board (IAD) twice a year to discuss how needs are fulfilled by the department. The department also conducts M.S. alumni interviews to gauge what the perception is once graduates have entered the job market. Recent interviews indicate that both employers and alumni want opportunities to be involved in real-world projects, to have offerings in certain contemporary programming languages, to have an emphasis on cyber-security and big data. The department also conducts alumni surveys for the B.S. and B.A. programs as part of the ABET accreditation activities. The feedback from M.S. graduates is very similar to that of the B.S. and B.A. surveys. The department is responsive in its course offerings and hiring decisions.

6. If not, what will the program do to get this assessment in place and by what date?[2]

   The data base mentioned in item 3 above will be online in Spring 2018 and contain data for graduates who graduated during the 2016/2017 academic year.

8. Additional Comments

[1] This is a new question to respond to recently implemented program review enhancements by the NSHE. (3/16)
[2] If the program has no employer expectations information, there must be a plan to put such a program in place and it has to be stated. (3/16)

IV. Program Resources

A. Faculty Time

1. Faculty and GA Resources

	Fall 2014	Spring 2015	Fall 2015	Spring 2016
Number of Full Time Faculty	15	17	18	18
Number of State-Supported GA lines	18	18	18	20
Number of PTIs	4	4	4	4
Number of FIRS & Visiting	0	0	0	0

	Fall 2014	Spring 2015	Fall 2015	Spring 2016
Percent of Classes Taught by Full Time Faculty	67%	73%	74%	78%
Percent of Classes Taught by Number of State-Supported GA lines	5%	2.5%	0%	2%
Percent of Classes Taught by Number of PTIs	28%	24.5%	26%	20%
Percent of Classes Taught by Number of FIRS & Visiting	0%	0%	0%	0%

	Fall 2014	Spring 2015	Fall 2015	Spring 2016
Student Credit Hours Taught by Full Time Faculty	2559	2491	3168	3213
Student Credit Hours Taught by Number of State-Supported GA lines	234	87	0	120
Student Credit Hours Taught by Number of PTIs	1602	1371	1521	1362
Student Credit Hours Taught by Number of FIRS & Visiting	0	0	0	0

2. For other non-major courses – e.g., upper division for the college or university, estimate the unit's resources allocated to them:
   N/A

B. General Education

1. If your program or unit offers General Education courses, please estimate what proportion of the unit's resources allocated to this area:
   It is difficult to ascertain an exact figure, as this information is not collected by the Dean's office. Numerous sections of "CS 115: Introduction to Computers" are taught. Our best estimate is 7%.

2. Does the combined load from A and B affect your unit's ability to offer courses for its major? If so, please describe:
   No. Through the utilization of PTIs the unit's ability to offer courses for its major is sustained.

C. Budget

1. Please fill in the table with three years of financial expenditures to be used to respond to questions 2 and 3 below.

Budget category	FY 13-14	FY 14-15	FY 15-16
State Operating (2101)	$ 21,110	$ 22,946	$ 22,946
Student Fees	N/A	$ 14,940	$ 16,560
Indirect Cost Recovery	$ 1,946	$ 8,970	$ 17,932
Self-supporting	$ $20,350	$ 59,738	$ 59,455
Total Allocations	$ 43,406	$ 106,311	$ 114,876
Number of Graduate Assistantships (including GAs on grants)	16	22	26

2. Are these resources sufficient to meet the degree program's instructional and scholarship needs?
   To maintain quality these resources are not sufficient and the state operating budget for the department needs to be increased. Currently, course enrollments exceed routinely exceed 50 students per course. More PTIs are urgently needed so that graduate research faculty can focus more on teaching upper-division and graduate doctoral level courses, and Ph.D. student advising. Focus should be on developing much needed expanding the body of courses in the areas of big data, cyber-security, and sustainability.

   In order to achieve top-tier ranking the number of Ph.D. graduates must substantially increase from current levels. The number of state funded Graduate Assistantships should increase by at least 50%. As the Ph.D. program is growing, more funding for conference travel is required. Part of this could come from an increase in overhead.

3. If not, approximately what line items and amounts would be needed?
   A conservative estimate would be a 25% state operating fund increase. This estimate is valid even if a modest increase in overhead (line-item self-supporting) occurs.

D. Other Funding and Resources

1. Is funding from other sources sufficient to assist the program in achieving its outcomes? Other sources to be considered include: differential tuition, grants and contracts, endowment income, and one-time gifts for student scholarships, other one-time gifts.

   No.
   - Differential and lab-fees are used to hire lab-monitors and tutors in programming sessions. This line-item is also used to equip and maintain computer science labs used by students. Thus, there is enormous contention for limited funds.
   - On average, only 3 GA's are supported from external grants (NSF, DoD, NIH).

   Conclusion: In order to grow the Ph.D. program, additional funds will be required.

2. If not, which funding streams could most reasonably be increased to help the program attain its outcomes?

Additional funding will allow us to attract and recruit students, nurture these students, and contribute to the top-tier mission of the university. Additional funding must be secured in state operating funds as well as by higher levels of external funding.

In light of the needs identified in Section "B. Needs for Graduates and Future Plans" three new faculty lines are needed in

- Sustainability and green computing
- Big Data Analysis
- Cybersecurity.

3. Has any new donor revenue been generated since the last program review?

   - The department secured GA support from NSTec ($200K 2010-2016). These funds were used for GA stipends in the area of cybersecurity.
   - Secured a fellowship from JT3, which was used for Ph.D. student Guymond Hall.
   - A modest amount of scholarship money was raised through TechConnect activities (see below).

4. Has the unit engaged in fundraising activities to support the program over the last 5 years?
   Most fundraising takes place at the college level. However, the department did engage in the following activities:
   - Tech Connect Event by the Industrial Advisory Board: Once a year for the last 7 years (Spring semester).
   - JT3 collaboration activities.

5. What has been the result of these fundraising activities?

   The activities have resulted in a scholarship fund in the amount of $12,000 for the last 4 years.

6. Review the space data for your department and comment on its amount and quality. These data will need to be accessed by an individual with Archibus® access.

   CS Department / Administrative office space = 982.54 sq ft
   CS Faculty office space = 3372.27 sq ft
   CS Labs space = 3853.46 sq ft

   Space for CS is entirely inadequate. Conservative estimates show that at this time – not factoring increases in Ph.D. student numbers – at least 7000 sq ft in Lab space is needed. Though space is made available in the Science and Engineering building on a per-project basis, there is a lack of permanent laboratory space, especially for state-funded graduate students. The lack of space is impacting work with students in a negative way. Student projects routinely are conducted in faculty offices. Minimal or no work space is afforded to sabbatical visitors.

7. Is the quality and quantity of available consumable materials and supplies (e.g., office supplies or lab supplies) adequate and if not, explain why not:

   At this time, it is barely adequate. If enrollment increases at current levels than an operational budget increase is needed within two years.

8. Is the quality and quantity of available technology resources, such as computers adequate and if not, explain why not:
   No. Though basic desktop computers are provided out of university funds, faculty purchase laptop computers or wide-screen displays out of pocket. There are no funds for software licenses (e.g. Mathematica) neither for faculty nor graduate students.

9. Is the quality and quantity of available equipment (other than computing) adequate and if not, explain why not:

   Classrooms have good technical facilities. However, smart boards are increasingly common and should be installed in faculty offices and laboratories.

10. Is the quality and quantity of available library and information resources adequate and if not, explain why not:

    Yes, though the library did try to cancel a number of important journals, the Faculty Senate prevented this.

11. Staffing
    a. Are available department staff resources sufficient to attain the program's outcomes?

       No. Currently there are two secretarial staff members and one network professional, which is adequate for current levels. To facilitate growth in the graduate programs, one more staff member is needed. To maintain support for five laboratories, there is an urgent need for one more system administrator.

    b. If not, what additional staff resources are needed and how would they be funded?

       The additional staff members should be funded by the state under top-tier initiatives.

12. Additional Comments

V. Size of Program

1. Below are headcount, course enrollment, and degrees conferred data from Decision Support.

Headcount

Computer Science MSCS (CSCMSCS)

Academic Level - Beginning of Term	Fall 2011 Prelim	Spring 2012 Prelim	Fall 2012 Prelim	Spring 2013 Prelim	Fall 2013 Prelim	Spring 2014 Prelim	Fall 2014 Prelim	Spring 2015 Prelim	Fall 2015 Prelim	Spring 2016 Prelim
GR (MS)	45	40	43	45	45	36	41	36	44	44

Computer Science PhD (CSCPHD)

Academic Level - Beginning of Term	Fall 2011 Prelim	Spring 2012 Prelim	Fall 2012 Prelim	Spring 2013 Prelim	Fall 2013 Prelim	Spring 2014 Prelim	Fall 2014 Prelim	Spring 2015 Prelim	Fall 2015 Prelim	Spring 2016 Prelim
PHD	3	4	4	3	5	5	9	11	11	12

Course Enrollments

Dept Name of Course	Subject	Course Number Level	Fall 2011 Prelim	Spring 2012 Prelim	Fall 2012 Prelim	Spring 2013 Prelim	Fall 2013 Prelim	Spring 2014 Prelim	Fall 2014 Prelim	Spring 2015 Prelim	Fall 2015 Prelim	Spring 2016 Prelim
Computer Science	CS	100-Level	424	462	517	517	613	540	690	648	704	689
		200-Level	138	163	156	164	184	226	251	239	273	297
		300-Level	203	136	202	139	190	159	220	180	245	215
		400-Level	93	150	147	212	192	190	193	193	268	262
		600-Level	39	31	45	56	40	23	66	21	57	40
		700-Level	50	76	62	63	55	71	47	87	58	92
	ITE	400-Level								2	2	8
		600-Level									6	

Department	Computer Science
Academic Career	GRAD
Academic Plan Description	Computer Science MSCS (CSMSCS)
Degree	MSCS
Degree Description	Master of Science in Computer

Academic Year - July to June	Degree Count
2004-05	18
2005-06	16
2006-07	21
2007-08	19
2008-09	14
2009-10	8
2010-11	18
2011-12	11
2012-13	19
2013-14	14
2014-15	13
2015-16	17

Department	Computer Science
Academic Career	GRAD
Academic Plan Description	Computer Science PHD (CSCPHD)
Degree	PHD
Degree Description	Doctor of Philosophy

Academic Year - July to June	Degree Count
2004-05	0
2005-06	1
2006-07	0
2007-08	0
2008-09	0
2009-10	0
2010-11	1
2011-12	0
2012-13	1
2013-14	0
2014-15	0
2015-16	0

2. Discuss the headcounts from the last five years, i.e., are the trends in line with projections in your unit's strategic plan?
   Yes.

3. If not, why not?

4. Does your program's enrollment trend differ from national trends?

   Yes.

   M.S.: Computer Science degrees are on the increase nationwide. For example, the number of Master's degrees in Computer Science (inside engineering) rose by 32.2 percent from 2014 to 2015 according to the latest ASEE report. The number of M.S. degrees in the CS department is stable.

   Ph.D.: The degree count is so small that a comparison with national trends cannot be made.

4. If yes, please discuss the reasons:

   M.S.: Stagnant GA funding levels.

   Ph.D.:

   i. For many years (before 2010) there has been a lack of recruitment activities. Currently there are 18 Ph.D. students due to more recent recruitment efforts. It is imperative that these efforts be enhanced. The B.S. to Ph.D. track also represents an improvement.
   ii. Until 2012 the comprehensive examination was much stricter than at peer institution. The rationale was that "UNLV as a second-tier institution needed to prove what rigor and high standard it had." In 2012 these requirements were brought in line with best practices at other institutions.

6. Additional Comments

VI. Retention, Progression, Completion

A. Major Course Offerings

1. Are enough courses offered to meet enrollment demands? There are enough courses to meet the demands of students. However, it is not always possible to fill advanced or doctoral level elective courses, which presents challenges for a subgroup of effected students.

2. How many major courses have been added or eliminated in the last 5 years?
   __3_ Added __0__ Eliminated

   CS649 (Computer and Network Forensics) is a companion course for CS449.
   CS795 (Directed Research) and CS798 (Dissertation Proposal) are created to help students spend more time on their research after they pass the Comprehensive Examination.

   Special topics courses/independent study courses were also taught:

   - Big Data Analytics

- Deep Learning
- Machine Learning
- Scheduling
- Online Competitive Algorithms
- Algorithmic Problems in Power Management

3. Why were the actions taken?

   In response to the fast-changing nature of the body of knowledge in computer science and information technology.

4. After reviewing the program, what additional actions should be taken to improve retention, progression, and completion?

   - As described above more faculty lines are needed to strengthen the program and cover emerging areas.
   - The overall GA situation must be improved, both in the number of lines and also the GA stipend itself. The stipend for Ph.D. students should be increased substantially. The B.S. to Ph.D. track should be utilized more frequently, by identifying suitable undergraduate students.
   - Lab space is inadequate, as described above.

5. Are there any courses that students routinely have difficulty getting enrolled in, that slow progression and/or graduation? If so, please identify them:

   Not at this time. Also, the recent addition of CS795 (Directed Research) and CS798 (Dissertation Proposal) give opportunities for individualized progression.

6. If last question was answered yes, what steps can be taken to reduce "bottle-necks" in these courses. Please indicate *both* financially-based and non-financially-based solutions.
   N/A

7. Can any changes in sequencing of courses be made to facilitate graduations?

   No suggestions at this time.

B. Graduation Rates

Program graduation numbers and rates are summarized below.

New Masters Students Graduating in Less than Six Years (Computer Science MSCS - CSCMSCS)

Fall 2006 - Fall 2013 Cohorts

Cohort Term	#	Graduated in... 2 Years	%	3 Years	%	4 Years	%	5 Years	%	6 Years	%	7 Years	%	8 Years	%
Fall 2006	18	8	44.4	10	55.6	11	61.1								
Fall 2007	14	7	50.0	11	78.6	11	78.6	0	0.0	0	0.0	0	0.0		
Fall 2008	12	3	25.0	6	50.0	7	58.3	-	-	-	-	-	-		
Fall 2009	14	8	57.1	12	85.7	12	85.7	1	50.0	1	50.0	1	0.0		
Fall 2010	17	5	29.4	8	47.1	10	58.8	-	-	-	-	-	-		
Fall 2011	11	6	54.5	10	90.9	10	90.9	0	0.0	0	0.0	*1/3*	*33.33*		
Fall 2012	12	7	58.3	7	58.3	*61/86*	*70.93*	1	100.0	*1/4*	*25.00*				
Fall 2013	13	7	53.8	*64/98*	*65.31*			*2/5*	*40.00*						
Combined Cohort		*51/111*	*45.95*												

New Doctoral Students Graduating within Eight Years (Computer Science PhD - CSCPHD)

Fall 2004 - Fall 2012 Cohorts

Cohort Term	#	3 Years	%	4 Years	%	5 Years	%	6 Years	%
Fall 2004	1	0	0.0	0	0.0	0	0.0	0	0.0
Fall 2005	0	-	-	-	-	-	-	-	-
Fall 2006	2	0	0.0	0	0.0	0	0.0	1	50.0
Fall 2007	0	-	-	-	-	-	-	-	-
Fall 2008	1	0	0.0	0	0.0	0	33.3	0	0.0
Fall 2009	1	1	100.0	1	100.0	1	100.0	1	100.0
Fall 2010	0	-	-	-	-	-	-	*2/5*	*40.00*
Fall 2011	1	0	0.0	0	0.0%	*1/5*	*20.00*		
Fall 2012	0	-	-	*1/6*	*16.67*				
Combined Cohort		*1/6*	*16.67*						

*Doctoral graduation rates include only doctoral degree conferrals.

Using the data in the tables above, please answer these questions:

1. Are trends in 6-year cohort graduation close to the University's goals (UNLV's undergrad goal is 50%)?

   The M.S. rate is 65%, which is considered close to the university goal. Ph.D. 8-year rate cannot be ascertained at this point.

2. If not, what is being done to reach the goal?
   N/A.
3. Discuss how and why the graduation rate is changing.

It is hard to see any continuing trend. The 2-year graduate rate has been nearly constant at around 55% in recent years.

4. Additional Comments

VII. Relationship to Other Programs

1. What relationship does your program have to other programs (such as transfers, collaborations, partnerships) in the NSHE system?

   Students can take 6 credits in the M.S. and Ph.D. programs outside CS. Typically these are taken in the Engineering College in Electrical or Mechanical Engineering, sometimes in the Sciences or in the Business School.

2. What the relationship does this program have to other programs at UNLV (e.g., collaborations, partnerships, affiliated faculty, General Education requirements, etc.)?

   The Center for Information Technology and Algorithms is an interdisciplinary center with directors from Computer Science (Dr. Wolfgang Bein) and Electrical Engineering (Dr. Shahram Latifi). Opportunities for joint student research projects exit. Dr. Fatma Nazos is both a Computer Science professor and a senior resident scholar of information technology with The Lincy Institute, a research institute organized to conduct and support research that focuses on improving Nevada's health, education, and social services. Computer Science is essential in nearly all engineering and science disciplines as well as in business. Big data research has significant overlap in health science, information security, social computing, and sustainability.

3. Additional Comments

VIII. Impact

1. What impact has this program had or will have in the following areas:
   a. University

      It has been said that the language of science is no longer mathematics but rather algorithmics. As such computer science provides foundations to other disciplines similar to endeavors such as philosophy or mathematics. In an engineering sense, computer science is versatile and offers a great inter-disciplinary research opportunity. The scholarly work contributes tremendously to the Top Tier initiatives under way. Students routinely participate in research in other departments and the university IT-related work.

   b. Community

      Our students are participating in internships and part-time jobs at local companies while at school, and many join these companies after graduation. They participate in hackathons, maker space, and tutoring, and provide their advanced skills to the community. They have a great impact in enhancing the technology levels in the region, and thus computer science contributes to the goal of diversifying the economy of Southern Nevada.

c. Field

Our faculty members are well established in their own research fields. Journal publications in CS have a high citation count. For example, in the area of computer science theory, Dr. Lawrence Larmore has 2364 citations with an h-index of 25; in cybersecurity Dr. Hal Berghel (who is an ACM fellow) is a regular contributor to IEEE Computer; in the area of big data science, Dr. Justin Zhan currently oversees a multi-million-dollar grant portfolio; and in the area of accessibility, Dr. Andreas Steffik received the White House Champion of Change Award from the Executive Office of the President. Currently, UNLV provides the only institution of higher learning in Nevada to offer a comprehensive immersion in computer science theory.

2. What are the benefits to the institution of offering this program?

Computer Science is essential to nearly all engineering and science disciplines, as well as in business. Computer Science jobs are growing twice as fast as the national average. Among all STEM jobs in US, computer science jobs account for 62% (source: US Bureau of Labor Statistics). Currently, there is still a shortage of CS professionals in southern Nevada, and CS professionals must be imported from other states. As such, a larger computer science program will greatly contribute to student job placement and will greatly impact in the local economy.

3. Are there examples of the integration of teaching, research, & service that you would like to highlight (e.g., faculty mentoring leading to student presentations at conferences, service learning classes, community service activities involving students, or other student activities and/or achievements that you think are noteworthy)?

Students can work with faculty on funded research leading to presentations at top-tier conferences and joint papers in high impact journals. The department has provided off-campus research experiences and conference participation at major conferences, which provides networking opportunities. For example, one of our Ph.D. students took part in NSF funded research at the University of Electro-Communications in Japan, and one of our recent Master's students was sponsored by the department to participate in the prestigious European Symposia on Algorithms held recently in Slovania.

4. Additional Comments

IX. Productivity

1. Please provide an indication of faculty productivity appropriate for your unit:

In 2016, the department had 8 journal publications and 23 publications in peer-reviewed conferences. More than half of these were co-authored with students in the graduate program. In the same year research expenditures were $321,028, which is trending up from previous years. (In 2016 the dollar amount of proposals submitted was $6,778,182.)

The normal teaching load is two courses per semester. A number of faculty receive an extra course reduction for exceptional research work or grant proposal development. Also, administrative responsibilities may reduce the teaching load.

2. Additional Comments

X. Quality

A. Admission and graduation requirements

1. Please provide program admission requirements as Appendix 2 from the current UNLV catalog:

   Both MS and Ph.D. admission requirements are attached in the Appendix 2.

2. Are there any updates that need to be made to the catalog and if so, what are they?
   The comprehensive examination has been significantly changed. The department used to give three two-hour examinations covering six core areas and one two-hour examination covering two application areas. Now, we give six examinations (30 minutes each) covering six core areas.

3. How many full-time advisors are available at the college level?
   There is sufficient advising available to our graduate students:
   - One full-time graduate coordinator with course reduction
   - Any graduate faculty member is available by appointment

B. Outcomes and Assessment

1. Student Learning Outcomes and Program Assessment Plans and Reports by program concentration are listed at http://provost.unlv.edu/Assessment/plans.html. Please attach the most recent assessment report as Appendix 3.
   Leaning Outcomes M.S.
   A. Acquire in-depth knowledge of specialized areas and advanced topics in computer science.
   B. Independently analyze, design, and implement an innovative computer application or research project.
   C. Prepare a final project or research report on the solution to a computer-related problem.
   D. Present the results of their research orally.

   Learning Outcomes Ph.D.:
   A. Exhibit a breadth of knowledge in the areas of algorithms, programming languages and compilers, theory, operating systems, and computer architecture.
   B. Exhibit a depth of knowledge in at least one specialized area of computer science.
   C. Conduct a thorough literature survey on a research topic.
   D. Prepare a research paper that is publishable.
   E. Present the research results in an oral defense.

   The recent assessment reports are attached.

2. Describe specific program changes made based on the program's evaluation of its assessment reports:
   We have added 3 elective courses: CS649 (Computer and Network Forensics), CS795 (Directed Research), and CS798 (Dissertation Proposal).

3. Has the program revised its curriculum such as changing prerequisites, adding or eliminating required or elective courses, or co-curricular experiences for the degree(s) in the last 5 years?
   We have added 3 elective courses: CS649 (Computer and Network Forensics), CS795 (Directed Research), and CS798 (Dissertation Proposal).

The comprehensive examination has been significantly changed. We used to give three 2-hour examinations covering 6 core areas and one 2-hour examination covering two application areas. Now, we give 6 half-hour examinations covering 6 core areas.

a. If yes, what changes were made and why?
CS649 (Computer and Network Forensics) is a companion course for CS449. CS795 (Directed Research) and CS798 (Dissertation Proposal) were created to help students spend more time on their research after passing their Comprehensive Examination.

4. Has the program revised course content or instructional approaches (pedagogy, technology) in the last 5 years?
New emphasis on big data and cyber-security was placed throughout the curriculum.

   a. If yes, what changes were made and why?

5. Describe any other changes made in the last 5 years (for example, advising) based on assessment reports.
No significant changes.

6. List and describe two specific improvements in student learning outcomes and why they represent forward movement. [3]

   We have simplified the Ph.D. Comprehensive Examination with fewer subject areas. We added CS795 and CS798 courses to help students spend more time on research.

7. Additional Comments

XI. Conclusions, Self-Assessment

A. Faculty Review of self-study

1. On what date did the program and/or department faculty review this self-study?
03/07/17-03/13/17. About half the graduate faculty provided input.

2. What were the results of the faculty review?
The faculty was content overall with the self-study, except that the number of Ph.D. students should increase further. Faculty agreed that recruitment efforts to attract Ph.D. students should be strengthened. One faculty member suggested including more subjects in the qualifying exam, in order to cover areas that students are interesting in working towards their Ph.D.

   Faculty also saw the need for more space as crucial to the program. As mentioned above, currently there is not even enough space to accommodate an academic visitor for a few days. The university is raising funds for a new Engineering building; clearly this could solve the problem.

   The faculty saw the need to further integrate the new concentrations of big data and cyber security into our programs, and to increase the number of faculty members working in these areas. Furthermore, the area of sustainability should be nurtured as a new emerging area in the department.

   On faculty member suggested to eliminate the project option and require a thesis for all M.S. students.

[3] This is a new question to respond to recently implemented program review enhancements by the NSHE. (3/16)

3. What are the top 3 priorities and/or needs for the future development of the program?
   i. Attract highly motivated graduate students, increase salaries for graduate students, both for the M.S. and the Ph.D. program.
   ii. Enhance strength of the program in Cybersecurity and Big Data, as well as the emerging area of Sustainability.
   iii. Increase the number of faculty, along with laboratory and office space. Also state operating funds will have to increase significantly to pursue the agendas laid out in this document,

4. What are the strengths of the program?
   - Core faculty, especially in CS Theory, have a long record of research published in top-tier conferences and journals.
   - Concentration in software engineering.
   - Cybersecurity and big data research with recent external funding successes.
   - Resources for STEM education.

5. What are the challenges facing the program?

   The most pressing need is to increase the pool of Ph.D. students through the various means mentioned. Furthermore, increased state funding and overhead from external funding could make higher student stipends possible. More faculty are needed as a number of core courses now routinely have enrollments of more than fifty students.

6. What recent additions, corrections, or other changes have been made to the program that reflect changes or developments in the field?
   - The Ph.D. comprehensive exam has been made less stringent, because the level was beyond best practices of other peer institutions.
   - The department has actively promoted CS participation in UNLV's Hackathons. This was embedded as a practice session in CS 645 "Internet Security" this Spring 2017. There will be an effort to enhance CS 445 by including a Certified Ethical Hacker component. In fact, local industry has offered to defray the certification cost.
   - In response to the Alumni comments regarding internships and job interview preparedness TechConnect has been established as an annual student job networking event attracting student majors from Computer Science, Management Information Services, Engineering, Mathematics and Science students in a forum promoting the availability of technology career opportunities.

B. Other comments

1. Is there anything else you would like to discuss about the program?

The NSHE also requires that any action steps identified based on the review of the program and the status of the action steps be ready for consideration at the December board meeting the year the program review is completed. You will be contacted about this after the external review has been completed.

NEXT STEPS:

A. Create an executive summary of this self-study, using the template provided, that is ***no more than 2 pages long***.
B. Email the self-study and the executive summary to:
   - Chair of the Faculty Senate Program Review Committee found here: http://facultysenate.unlv.edu/committees/program-review or the Chair of the Graduate College Program Review Committee found here: http://www.unlv.edu/graduatecollege/program-review-committee
   - Gail Griffin, gail.griffin@unlv.edu, 702-895-0482.

Doctor of Philosophy - Computer Science

Plan Description

The Ph.D. degree is awarded to a candidate who has demonstrated breadth of knowledge in computer science in general and has displayed depth of knowledge in the area of specialty as well as the ability to make original contributions to the body of knowledge in this field.
For more information about your program, including your graduate program handbook and learning outcomes, please visit the Degree Directory.

Plan Admission Requirements

Applications available on the UNLV Graduate College website.

Applicants for admission to the Ph.D. program in computer science must meet the following:

1. A GPA of 3.70 (on a 4.00 scale) or higher in post-baccalaureate course work is required for admission. Students entering with a bachelor's degree must have a GPA of 3.5 or higher for the courses at the 200-level or above.
2. Students are expected to have a master's degree in computer science before applying to the Ph.D. program. On rare occasions, an unusually capable student may be admitted to work directly for the Ph.D. degree without having a master's degree.
3. At least three letters of recommendation (preferably from academic sources) attesting to the applicant's professional competence and academic potential are required.
4. A personal statement of purpose, which should be as specific as possible and should include the applicant's objectives and area(s) of interest, is required.
5. A minimum score of 315 on the general test of the Graduate Record Examination (GRE) is required. Official score reports from the last five years are acceptable.
6. All domestic and international applicants must review and follow the Graduate College Admission and Registration Requirements.

Students are accepted into a degree program as described in the Graduate Catalog. The faculty and corresponding sub-disciplines and sub-plans within the described programs are subject to change at any time.

Plan Requirements

See Subplan Requirements below.

Subplan 1 Requirements: Post-Master's Track

Total Credits Required: 48

Course Requirements

Required Courses – Credits: 30

Complete 30 credits of 600- or 700- level Computer Science (CS) courses.

Dissertation – Credits: 18

CS 799 - Dissertation Research

Degree Requirements

1. A student entering the Ph.D. program with a master's degree in computer science is required to take at least 48 credits of coursework.
2. At least 24 credits must be in computer science (excluding dissertation).
3. A minimum of 12 credits of 700-level Computer Science courses (excluding dissertation)
4. A maximum of 12 credits of 600-level Computer Science courses.
5. A maximum of 6 credits of 600/700 level non-Computer Science courses (with departmental approval).
6. Satisfactorily pass a written comprehensive examination within the first four semesters.
    a. The written comprehensive examination will be given twice a year. The comprehensives will assess the student's breadth of knowledge through two

examinations covering the six Core Areas listed below and another examination in two other areas of his or her choice.

b. Core Areas:
    i. Automata and formal languages; Algorithms and data structures
    ii. Programming languages; Compiler construction
    iii. Computer architecture; Operating systems
c. Application Areas:
    i. Artificial intelligence
    ii. Computer graphics and image processing
    iii. Computer simulation and networks
    iv. Data base systems
    v. Software engineering and reliability
    vi. Document analysis
    vii. Networks and distributed computing
    viii. Geometric applications
d. The level of the examination is that of 600-level and 700-level courses in each area. A syllabus will be published well in advance of the exams listing the topics to be covered in each exam. Students are expected to take the comprehensive examination within two years of entering the Ph.D. program. All Ph.D. students are urged to take this examination as early as possible. Preference is given in the allocation of student financial support to those who have passed the comprehensive examination. The comprehensive examination may be attempted at most twice. Students who do not pass the comprehensive examination the first time must retake the examination at the next scheduled offering. Failure to pass the comprehensive examination after two attempts will normally lead to dismissal from the Ph.D. program. After passing the comprehensive examination, a research topic of mutual interest to the student and his/her proposed committee is selected. At this point, the student formally begins his/her research study.

7. The qualifying examination is an oral examination designed to test the depth of the student's knowledge in his or her area of research specialization.
    a. It must be taken before either:
        i. Two years after passing the comprehensive examination or
        ii. Four years after entering the Ph.D. program.
    b. It generally focuses on his/her dissertation proposal. The main purpose of this exam is to evaluate the technical merits and feasibility of the student's proposal for his/her Ph.D. dissertation.
    c. The student's Ph.D. committee must conduct the examination. This committee consists of five faculty members of whom one must be from outside the school of computer science. The student's advisor is the chairperson of this committee. Please see Graduate College policy for committee appointment guidelines.
    d. The student must prepare a dissertation proposal before taking this examination. The student's advisor should have already approved this proposal. This proposal must be given to the Ph.D. committee members at least two weeks before the date of the qualifying exam. The proposal must contain a discussion of the background literature on the problem area, description of the specific topic of research proposal approach, feasibility arguments, the

objective of the research project, and a list of references.
    e. The student begins the exam with a presentation of the dissertation proposal. The remaining time is used for discussion and asking questions to determine if the student has sufficient depth of knowledge to carry out the proposed research.
    f. The examination cannot be taken more than twice. After successful completion of the qualifying examination, the student is advanced to candidacy for the doctoral degree.
8. Satisfactorily pass a dissertation proposal defense by the end of year 4.
9. The candidate must prepare a dissertation on his or her research. The doctoral dissertation should represent a significant original research contribution to the field of computer science and be publishable in a recognized refereed journal.
10. After completion of the dissertation, the candidate must pass a final oral defense of his/her dissertation. The candidate must make the final changes, if any, in the dissertation within three months from the date of the oral defense. A candidate can defend the dissertation no more than twice. Each member of the committee must approve the final dissertation.
11. Maintain a satisfactory rate of progress and a yearly progress report must be submitted. To maintain satisfactory progress in the Ph.D. program a student must:
    a. Pass the comprehensive examination within 2.50 years of entering the Ph.D. program.
    b. Maintain a minimum grade point average required by the College of Engineering.
    c. Pass the qualifying examination within four years of entering the Ph.D. program.
    d. Maintain satisfactory progress towards research.
    e. Students who enter the Ph.D. program with a master's degree must complete all requirements for the Ph.D. degree within six years. Those who enter the Ph.D. program with a bachelor's degree must complete all requirements for the Ph.D. degree within eight years. If these requirements are not met, the department may place the student on academic probation or drop him/her from the Ph.D. program.

Graduation Requirements

1. The student must submit all required forms to the Graduate College and then apply for graduation up to two semesters prior to completing his/her degree requirements.
2. The student must submit and successfully defend his/her dissertation by the posted deadline. The defense must be advertised and is open to the public.
3. The student must submit his/her approved, properly formatted hard-copy dissertation to the Graduate College, and submit the approved electronic version to ProQuest by the posted deadline.

Subplan 2 Requirements: Post-Bachelor's - No Master's/Ph.D. Only Track

Total Credits Required: 72

Course Requirements

Required Courses – Credits: 54

Complete 54 credits of 600- or 700- level Computer Science (CS) courses.

Dissertation – Credits: 18

CS 799 - Dissertation Research

Degree Requirements

1. Complete a minimum of 72 credits of coursework with a minimum GPA of 3.00.
2. At least 42 credits must be in computer science (excluding dissertation).

3. A minimum of 24 credits of 700-level Computer Science courses (excluding dissertation)
4. A maximum of 12 credits of 600-level Computer Science courses.
5. A maximum of 6 credits of 600/700 level non-Computer Science courses (with departmental approval).
6. Satisfactorily pass a written comprehensive examination within the first four semesters.
    a. The written comprehensive examination will be given twice a year. The comprehensives will assess the student's breadth of knowledge through two examinations covering the six Core Areas listed below and another examination in two other areas of his or her choice.
    b. Core Areas:
        i. Automata and formal languages; Algorithms and data structures
        ii. Programming languages; Compiler construction
        iii. Computer architecture; Operating systems
    c. Application Areas:
        i. Artificial intelligence
        ii. Computer graphics and image processing
        iii. Computer simulation and networks
        iv. Data base systems
        v. Software engineering and reliability
        vi. Document analysis
        vii. Networks and distributed computing
        viii. Geometric applications
    d. The level of the examination is that of 600-level and 700-level courses in each area. A syllabus will be published well in advance of the exams listing the topics to be covered in each exam. Students are expected to take the comprehensive examination within two years of entering the Ph.D. program. All Ph.D. students are urged to take this examination as early as possible. Preference is given in the allocation of student financial support to those who have passed the comprehensive examination. The comprehensive examination may be attempted at most twice. Students who do not pass the comprehensive examination the first time must retake the examination at the next scheduled offering. Failure to pass the comprehensive examination after two attempts will normally lead to dismissal from the Ph.D. program. After passing the comprehensive examination, a research topic of mutual interest to the student and his/her proposed committee is selected. At this point, the student formally begins his/her research study.
7. The qualifying examination is an oral examination designed to test the depth of the student's knowledge in his or her area of research specialization.
    a. It must be taken before either:
        i. Two years after passing the comprehensive examination or
        ii. Four years after entering the Ph.D. program.
    b. It generally focuses on his/her dissertation proposal. The main purpose of this exam is to evaluate the technical merits and feasibility of the student's proposal for his/her Ph.D. dissertation.
    c. The student's Ph.D. committee must conduct the examination. This committee consists of five faculty members of whom one must be from outside the school of computer science. The student's advisor is the chairperson of this committee. Please see Graduate College policy for committee appointment guidelines.

d. The student must prepare a dissertation proposal before taking this examination. The student's advisor should have already approved this proposal. This proposal must be given to the Ph.D. committee members at least two weeks before the date of the qualifying exam. The proposal must contain a discussion of the background literature on the problem area, description of the specific topic of research proposal approach, feasibility arguments, the objective of the research project, and a list of references.
e. The student begins the exam with a presentation of the dissertation proposal. The remaining time is used for discussion and asking questions to determine if the student has sufficient depth of knowledge to carry out the proposed research.
f. The examination cannot be taken more than twice. After successful completion of the qualifying examination, the student is advanced to candidacy for the doctoral degree.

8. Satisfactorily pass a dissertation proposal defense by the end of year 4.
9. The candidate must prepare a dissertation on his or her research. The doctoral dissertation should represent a significant original research contribution to the field of computer science and be publishable in a recognized refereed journal.
10. After completion of the dissertation, the candidate must pass a final oral defense of his/her dissertation. The candidate must make the final changes, if any, in the dissertation within three months from the date of the oral defense. A candidate can defend the dissertation no more than twice. Each member of the committee must approve the final dissertation.
11. Maintain a satisfactory rate of progress and a yearly progress report must be submitted.

To maintain satisfactory progress in the Ph.D. program a student must:

a. Pass the comprehensive examination within 2.50 years of entering the Ph.D. program.
b. Maintain a minimum grade point average required by the College of Engineering.
c. Pass the qualifying examination within four years of entering the Ph.D. program.
d. Maintain satisfactory progress towards research.
e. Students who enter the Ph.D. program with a master's degree must complete all requirements for the Ph.D. degree within six years. Those who enter the Ph.D. program with a bachelor's degree must complete all requirements for the Ph.D. degree within eight years. If these requirements are not met, the department may place the student on academic probation or drop him/her from the Ph.D. program.

Graduation Requirements

1. The student must submit all required forms to the Graduate College and then apply for graduation up to two semesters prior to completing his/her degree requirements.
2. The student must submit and successfully defend his/her dissertation by the posted deadline. The defense must be advertised and is open to the public.
3. The student must submit his/her approved, properly formatted hard-copy dissertation to the Graduate College, and submit the approved electronic version to ProQuest by the posted deadline.

Subplan 3 Requirements: Post-Bachelor's - Thesis Track

Total Credits Required: 72

Course Requirements

Required Master's Courses – Credits: 24

Complete 24 credits of 600- or 700- level Computer Science (CS) courses.

Thesis – Credits: 6

CS 791 - Thesis

After successfully completing the requirements above, students are eligible to earn the Master of Science – Computer Science.

Required Doctoral Courses – Credits: 24

Complete 24 credits of 600- or 700- level Computer Science (CS) courses

Dissertation – Credits: 18

CS 799 - Dissertation Research

Degree Requirements

1. Complete a minimum of 72 credits of coursework with a minimum GPA of 3.00.
2. At least 42 credits must be in computer science (excluding thesis & dissertation).
3. A minimum of 24 credits of 700-level Computer Science courses (excluding thesis & dissertation)
4. A maximum of 12 credits of 600-level Computer Science courses.
5. A maximum of 6 credits of 600/700 level non-Computer Science courses (with departmental approval).
6. The student must submit a thesis conforming to the specifications of the Graduate College and pass a final oral examination covering the thesis and relevant course work.
7. Satisfactorily pass a written comprehensive examination within the first four semesters.
    a. The written comprehensive examination will be given twice a year. The comprehensives will assess the student's breadth of knowledge through two examinations covering the six Core Areas listed below and another examination in two other areas of his or her choice.
    b. Core Areas:
        i. Automata and formal languages; Algorithms and data structures
        ii. Programming languages; Compiler construction
        iii. Computer architecture; Operating systems
    c. Application Areas:
        i. Artificial intelligence
        ii. Computer graphics and image processing
        iii. Computer simulation and networks
        iv. Data base systems
        v. Software engineering and reliability
        vi. Document analysis
        vii. Networks and distributed computing
        viii. Geometric applications
    d. The level of the examination is that of 600-level and 700-level courses in each area. A syllabus will be published well in advance of the exams listing the topics to be covered in each exam. Students are expected to take the comprehensive examination within two years of entering the Ph.D. program. All Ph.D. students are urged to take this examination as early as possible. Preference is given in the allocation of student financial support to those who have passed the comprehensive examination. The comprehensive examination may be attempted at most twice. Students who do not pass the comprehensive examination the first time must retake the examination at the next scheduled offering. Failure to pass the comprehensive examination after two attempts will normally lead to dismissal from the Ph.D. program. After passing the comprehensive examination, a research topic of mutual interest to the student and his/her proposed committee is

selected. At this point, the student formally begins his/her research study.

8. The qualifying examination is an oral examination designed to test the depth of the student's knowledge in his or her area of research specialization.
    a. It must be taken before either:
        i. Two years after passing the comprehensive examination or
        ii. Four years after entering the Ph.D. program.
    b. It generally focuses on his/her dissertation proposal. The main purpose of this exam is to evaluate the technical merits and feasibility of the student's proposal for his/her Ph.D. dissertation.
    c. The student's Ph.D. committee must conduct the examination. This committee consists of five faculty members of whom one must be from outside the school of computer science. The student's advisor is the chairperson of this committee. Please see Graduate College policy for committee appointment guidelines.
    d. The student must prepare a dissertation proposal before taking this examination. The student's advisor should have already approved this proposal. This proposal must be given to the Ph.D. committee members at least two weeks before the date of the qualifying exam. The proposal must contain a discussion of the background literature on the problem area, description of the specific topic of research proposal approach, feasibility arguments, the objective of the research project, and a list of references.
    e. The student begins the exam with a presentation of the dissertation proposal. The remaining time is used for discussion and asking questions to determine if the student has sufficient depth of knowledge to carry out the proposed research.
    f. The examination cannot be taken more than twice. After successful completion of the qualifying examination, the student is advanced to candidacy for the doctoral degree.
9. Satisfactorily pass a dissertation proposal defense by the end of year 4.
10. The candidate must prepare a dissertation on his or her research. The doctoral dissertation should represent a significant original research contribution to the field of computer science and be publishable in a recognized refereed journal.
11. After completion of the dissertation, the candidate must pass a final oral defense of his/her dissertation. The candidate must make the final changes, if any, in the dissertation within three months from the date of the oral defense. A candidate can defend the dissertation no more than twice. Each member of the committee must approve the final dissertation.
12. Maintain a satisfactory rate of progress and a yearly progress report must be submitted. To maintain satisfactory progress in the Ph.D. program a student must:
    a. Pass the comprehensive examination within 2.50 years of entering the Ph.D. program.
    b. Maintain a minimum grade point average required by the College of Engineering.
    c. Pass the qualifying examination within four years of entering the Ph.D. program.
    d. Maintain satisfactory progress towards research.
    e. Students who enter the Ph.D. program with a master's degree must complete all requirements for the Ph.D. degree within six years. Those who enter the Ph.D. program with a bachelor's degree must complete all requirements for the

Ph.D. degree within eight years. If these requirements are not met, the department may place the student on academic probation or drop him/her from the Ph.D. program.

Graduation Requirements

1. The student must submit all required forms to the Graduate College and then apply for graduation up to two semesters prior to completing his/her master's degree requirements.
2. The student must submit and successfully defend his/her thesis by the posted deadline. The defense must be advertised and is open to the public.
3. The student must submit his/her approved, properly formatted hard-copy thesis to the Graduate College, and submit the approved electronic version to ProQuest by the posted deadline.
4. The student must submit all required forms to the Graduate College and then apply for graduation up to two semesters prior to completing his/her doctoral degree requirements.
5. The student must submit and successfully defend his/her dissertation by the posted deadline. The defense must be advertised and is open to the public.
6. The student must submit his/her approved, properly formatted hard-copy dissertation to the Graduate College, and submit the approved electronic version to ProQuest by the posted deadline.

Subplan 4 Requirements: Post-Bachelor's - Project Track

Total Credits Required: 72

Course Requirements

Required Master's Courses – Credits: 27

Complete 27 credits of 600- or 700- level Computer Science (CS) courses.

Project – Credits: 3

CS 790 - Master's Project

After successfully completing the requirements above, students are eligible to earn the Master of Science – Computer Science.

Required Doctoral Courses – Credits: 24

Complete 24 credits of 600- or 700- level Computer Science (CS) courses

Dissertation – Credits: 18

CS 799 - Dissertation Research

Degree Requirements

1. Complete a minimum of 72 credits of coursework with a minimum GPA of 3.00.
2. At least 42 credits must be in computer science (excluding dissertation).
3. A minimum of 24 credits of 700-level Computer Science courses (excluding dissertation)
4. A maximum of 12 credits of 600-level Computer Science courses.
5. A maximum of 6 credits of 600/700 level non-Computer Science courses (with departmental approval).
6. The student must complete a computer science project and a report approved by his/her advisor and pass a final oral examination over the project and relevant course work.
7. Satisfactorily pass a written comprehensive examination within the first four semesters.
    a. The written comprehensive examination will be given twice a year. The comprehensives will assess the student's breadth of knowledge through two examinations covering the six Core Areas listed below and another examination in two other areas of his or her choice.
    b. Core Areas:
        i. Automata and formal languages; Algorithms and data structures

ii. Programming languages; Compiler construction
iii. Computer architecture; Operating systems

c. Application Areas:
i. Artificial intelligence
ii. Computer graphics and image processing
iii. Computer simulation and networks
iv. Data base systems
v. Software engineering and reliability
vi. Document analysis
vii. Networks and distributed computing
viii. Geometric applications

d. The level of the examination is that of 600-level and 700-level courses in each area. A syllabus will be published well in advance of the exams listing the topics to be covered in each exam. Students are expected to take the comprehensive examination within two years of entering the Ph.D. program. All Ph.D. students are urged to take this examination as early as possible. Preference is given in the allocation of student financial support to those who have passed the comprehensive examination. The comprehensive examination may be attempted at most twice. Students who do not pass the comprehensive examination the first time must retake the examination at the next scheduled offering. Failure to pass the comprehensive examination after two attempts will normally lead to dismissal from the Ph.D. program. After passing the comprehensive examination, a research topic of mutual interest to the student and his/her proposed committee is selected. At this point, the student formally begins his/her research study.

8. The qualifying examination is an oral examination designed to test the depth of the student's knowledge in his or her area of research specialization.
a. It must be taken before either:
i. Two years after passing the comprehensive examination or
ii. Four years after entering the Ph.D. program.
b. It generally focuses on his/her dissertation proposal. The main purpose of this exam is to evaluate the technical merits and feasibility of the student's proposal for his/her Ph.D. dissertation.
c. The student's Ph.D. committee must conduct the examination. This committee consists of five faculty members of whom one must be from outside the school of computer science. The student's advisor is the chairperson of this committee. Please see Graduate College policy for committee appointment guidelines.
d. The student must prepare a dissertation proposal before taking this examination. The student's advisor should have already approved this proposal. This proposal must be given to the Ph.D. committee members at least two weeks before the date of the qualifying exam. The proposal must contain a discussion of the background literature on the problem area, description of the specific topic of research proposal approach, feasibility arguments, the objective of the research project, and a list of references.
e. The student begins the exam with a presentation of the dissertation proposal. The remaining time is used for discussion and asking questions to determine if the student has sufficient depth of

knowledge to carry out the proposed research.

f. The examination cannot be taken more than twice. After successful completion of the qualifying examination, the student is advanced to candidacy for the doctoral degree.

9. Satisfactorily pass a dissertation proposal defense by the end of year 4.
10. The candidate must prepare a dissertation on his or her research. The doctoral dissertation should represent a significant original research contribution to the field of computer science and be publishable in a recognized refereed journal.
11. After completion of the dissertation, the candidate must pass a final oral defense of his/her dissertation. The candidate must make the final changes, if any, in the dissertation within three months from the date of the oral defense. A candidate can defend the dissertation no more than twice. Each member of the committee must approve the final dissertation.
12. Maintain a satisfactory rate of progress and a yearly progress report must be submitted. To maintain satisfactory progress in the Ph.D. program a student must:
    a. Pass the comprehensive examination within 2.50 years of entering the Ph.D. program.
    b. Maintain a minimum grade point average required by the College of Engineering.
    c. Pass the qualifying examination within four years of entering the Ph.D. program.
    d. Maintain satisfactory progress towards research.
    e. Students who enter the Ph.D. program with a master's degree must complete all requirements for the Ph.D. degree within six years. Those who enter the Ph.D. program with a bachelor's degree must complete all requirements for the Ph.D. degree within eight years. If these requirements are not met, the department may place the student on academic probation or drop him/her from the Ph.D. program.

Graduation Requirements

1. The student must submit all required forms to the Graduate College and then apply for graduation up to two semesters prior to completing his/her degree requirements for both the master's and doctoral portions of the program.
2. The student must successfully complete a master's project.
3. The student must submit and successfully defend his/her dissertation by the posted deadline. The defense must be advertised and is open to the public.
4. The student must submit his/her approved, properly formatted hard-copy dissertation to the Graduate College, and submit the approved electronic version to ProQuest by the posted deadline.

Plan Graduation Requirements

Refer to your subplan for Graduation Requirements.

Master of Science in Computer Science

Plan Description

Our master's program gives you the opportunity to study different areas, including:

- Design and analysis of algorithms
- Operating and distributed systems
- Computer architecture and networking
- Computational geometry and robotics
- Computer graphics and image processing
- Programming languages and compiler construction
- Artificial intelligence and expert systems
- Database design, document analysis, and retrieval
- Software engineering

For more information about your program including your graduate program handbook and learning outcomes please visit the Degree Directory.

Plan Admission Requirements

Applications available on the UNLV Graduate College website.

Applicants must submit the following to the Graduate College:

1. An application and official transcripts of all college level work with a minimum GPA of 3.00.
2. Two letters of recommendation concerning the student's potential for succeeding in the graduate program.
3. Another set of official transcripts.
4. The results of the Graduate Record Examination current to within five years should be sent directly to the school.
5. In addition, applicants must have completed courses and their prerequisites equivalent to our undergraduate Programming Languages CS 326), Operating Systems (CS 370), Discrete Mathematics II (MATH 351), and Statistical Methods I (STAT 411) with an average grade of B or better.
6. The Computer Science Admission Committee may elect to admit an outstanding applicant who has not satisfied all of the background requirements on a conditional basis.
7. All domestic and international applicants must review and follow the Graduate College Admission and Registration Requirements.

The student must complete these requirements before full admission to the program is granted.

Students who have not completed all the following courses (or equivalent courses) as part of their bachelor's degree may be required to complete them as a condition of their admission. If taken as part of their master's degree program, these courses may count toward the 30 credits required.

CS 656 - Automata and Formal Languages
CS 677 - Analysis of Algorithms
CS 660 - Compiler Construction

Students are accepted into a degree program as described in the Graduate Catalog. The faculty and corresponding sub-disciplines and sub-plans within the described programs are subject to change at any time.

Plan Requirements

See Subplan Requirements below.

Subplan 1 Requirements: Thesis Track

Total Credits Required: 30

Course Requirements

Computer Science Courses – Credits: 24

Complete 24 credits of 600- or 700- level Computer Science (CS) courses. Students may complete up to 3 credits outside of CS. Outside credits must be related to the student's research area and be approved by the school graduate committee.

Thesis – Credits: 6

CS 791 - Thesis

Degree Requirements

1. The student must pass at least 30 credits of 600- and 700-level courses with grades of C or better.
2. Students must complete 12 credits of 700-level CS courses (excluding thesis).
3. Courses in which the student earns a grade lower than C cannot be included in his or her program, and the student's total grade point average (GPA) must be 3.00 or higher while in the program. A student whose GPA falls below 3.00 will be placed on academic probation. That student must have an overall GPA of at least 3.00 by the end of two subsequent semesters; otherwise the student will be separated from the graduate program. A student on probation will not be allowed to register for CS 690, CS 790, CS 791, CS 792, CS 799, or equivalent courses in another department.
4. In consultation with his/her advisor, a student will organize a thesis committee of at least three departmental members. In addition, a fourth member from outside the department, known as the Graduate College Representative, must be appointed. An additional committee member may be added at the student and department's discretion. Please see Graduate College policy for committee appointment guidelines.
5. The student must submit a thesis conforming to the specifications of the Graduate College and pass a final oral examination covering the thesis and relevant course work.

Graduation Requirements

1. The student must submit all required forms to the Graduate College and then apply for graduation up to two semesters prior to completing his/her degree requirements.
2. The student must submit and successfully defend his/her thesis by the posted deadline. The defense must be advertised and is open to the public. .
3. The student must submit his/her approved, properly formatted hard-copy thesis to the Graduate College, and submit the approved electronic version to ProQuest by the posted deadline.

Subplan 2 Requirements: Project Track

Total Credits Required: 30

Course Requirements

Computer Science Courses – Credits: 27

Complete 27 credits of 600- or 700- level Computer Science (CS) courses. Students may complete up to 3 credits outside of CS. Outside credits must be related to the student's research area and be approved by the school graduate committee.

Project – Credits: 3

CS 790 - Master's Project

Degree Requirements

1. The student must pass at least 30 credits of 600- and 700-level courses with grades of C or better.
2. Students must complete 15 credits of 700-level CS courses (excluding the project).
3. Courses in which the student earns a grade lower than C cannot be included in his or her program, and the student's total grade point average (GPA) must be 3.00 or higher while in the program. A student whose GPA falls below 3.00 will be placed on academic probation. That student must have an overall GPA of at least 3.00 by the end of two subsequent semesters; otherwise the student will be separated from the graduate program. A student on probation will not be allowed to register for CS 690, CS 790, CS 791, CS 792, CS 799, or equivalent courses in another department.
4. The student must complete a computer science project and a report approved by his/her advisor and pass a final oral examination over the project and relevant course work.

Graduation Requirements

1. The student must submit all required forms to the Graduate College and then apply for graduation up to two semesters prior to completing his/her degree requirements.
2. The student must successfully complete a master's project.

Plan Graduation Requirements

Refer to your subplan for Graduation Requirements.

Computer Science Courses

CS 617 - Introduction to Computer Simulation
Credits 3
Simulation as a tool for the investigation of random phenomena. Emphasis on discrete simulation. Preparation of input for simulation and analysis of results. Use of SIMSCRIPT for discrete simulation. Comparison of discrete and continuous simulation. Simulation problems in several disciplines examined in detail.
Notes This course is crosslisted with CS 417. Credit at the 600-level requries additional work.

CS 620 - Human-Computer Interaction
Credits 3
Overview of human-computer interaction principles, guidelines, methods, and tools. User research, low-fidelity prototyping, participatory design, usability evaluation, visual design, usability principles, and affordances. Graphical user interface implementation, including design patterns, event handling, widget tool kits, languages, and development environments.
Notes This course is crosslisted with CS 420. Credit at the 600-level requires additional work.
Prerequisites Consent of Instructor

CS 641 - Advanced Internet Programming
Credits 2
Advanced Internet programming design and applications including client/server technologies and environment and software, client/server network operating systems, client/server database management systems, data warehousing environments, data mining, basic networking models and protocols, CASE tools, Groupware, Middleware, Internet security, privacy considerations.
Notes This course is crosslisted with CS 441. Credit at the 600-level requires additional work.

CS 641L - Advanced Internet Programming Lab
Credits 1
Helps student develop practical skills and learn to apply industry-wide standards and practices for advanced Internet and Internet 2 applications.
Notes This course is crosslisted with CS 441L. Credit at the 600-level requires additional work.

CS 643 - Information Assurance
Credits 3
Introduction to the principles of information assurance. Security awareness, Survey of information security technologies, cryptography, management and administration techniques necessary to improve information security and respond to a security breach, survey of threats to information security, privacy in computing, legal and ethical issues relating to information security, and case studies.
Same as CS 443

CS 645 - Internet Security
Credits 3
Internet security theory and practice, advanced IP concepts, the concepts of stimulus and response in the context of securing a network, network packet and traffic analysis, internet protocol (IP) vulnerabilities, packet filtering, intrusion detection, internet exploits, exploit signatures, internet forensics, network security investigation.
Notes This course is crosslisted with CS 445. Credit at the 600-level requires additional work.

CS 648 - Computer Security
Credits 3
Overview of computer security, threats, vulnerabilities and controls. Physical security, computer security policies and implementation plans, and computer forensics including penetration testing and investigation. Management issues. Legal, privacy and ethical issues.
Notes This course is crosslisted with CS 448. Credit at the 600-level requires additional work.

CS 649 - Computer and Network Forensics
Credits 3
Basics of Computer Forensics and Network Forensics. How to protect your privacy on the internet: Email, obfuscation, web sites and servers. Encryption, data hiding, and hostile code. Investigating Windows and Unix. File system recovery/analysis and file management in different OSes. Technical and legal issues regarding digital evidence collection and forensics analysis. This course is crosslisted with CS 449. Credit at the 600-level requires additional work.

Prerequisites CS 645 or CS 648

CS 651 - Multimedia Systems Design
Credits 2
Theory and practice of multimedia system design overview. High-level topics include multimedia content and formats, underlying technologies, digital cinematography, scripting, storyboarding, CD-ROM production and online publication, porting multimedia to the Web. Emphasis on the design process and the seamless integration of content in an interactive environment.
Notes This course is crosslisted with CS 451. Credit at the 600-level requires additional work.

CS 651L - Multimedia Systems Design Lab
Credits 1
Helps student develop practical skills and learn to apply industry-wide standards and practices for the design of multimedia systems.
Notes This course is crosslisted with CS 451L. Credit at the 600-level requires additional work.

CS 656 - Automata and Formal Languages
Credits 3
Regular expressions. Regular, context-free, and unrestricted grammars. Finite and pushdown automata. Turing machines and the halting problem; introduction to decidability.
Notes This course is crosslisted with CS 456. Credit at the 600-level requires additional work.

CS 657 - Database Management Systems
Credits 3
Concepts and structures necessary for design and implementation of a database management system. Survey of current database management systems and use of a DBMS.
Notes This course is crosslisted with CS 457. Credit at the 600-level requires additional work.

CS 660 - Compiler Construction
Credits 3
Current methods in the design and implementation of compilers. Construction of the components of an actual compiler as a term project.
Notes This course is crosslisted with CS 460. Credit at the 600-level requires additional work.

CS 663 - Computer Architecture
Credits 3
Introduction to computer architecture. Topics include

basic computer organization concepts; history and taxonomy of computer architectures; language and software influences on architecture; instruction set design; stack, array, data flow, and database machines; multiprocessor and network architectures; and fault tolerant designs.
Notes This course is crosslisted with CS 463. Credit at the 600-level requires additional work.

CS 665 - Computer Networks I
Credits 3
An introduction to the design and implementation of computer communication networks, their protocols and applications. It covers the technologies and standards in data transmission, telecommunication networks, network architectures, networking hardware, wireless networks, and the basis of the Internet including UDP and TCP as well as a number of application protocols.
Notes This course is crosslisted with CS 465. Credit at the 600-level requires additional work.
Prerequisites CS 370

CS 666 - Computer Networks II
Credits 3
Explores advanced topics in computer networks, the protocols, algorithms, hardware, and performance issues, especially in TCP/IP networks. Details of IP routing algorithms, quality of service, protocol implementation issues, router architecture and types, various TCP versions and their performance, the related telecommunication networks, and wireless technologies are discussed.
Notes This course is crosslisted with CS 466. Credit at the 600-level requires additional work.
Prerequisites CS 665 or CS 465

CS 669 - Introduction to Digital Image Processing
Credits 3
Background and basics of digital image processing. Topics include: the human visual system, image representation, sampling, image mathematics, and geometry, image enhancement, smoothing and sharpening, the fast Fourier transform, and a survey of image restoration methods.
Notes This course is crosslisted with CS 469. Credit at the 600-level requires additional work.
Prerequisites MATH 365 and STAT 411 and CS 117 or CS 135

CS 670 - Networks and Distributed Systems
Credits 3
Explores protocols and experiments with creating and implementing new protocols. In addition, students will be introduced to concepts such as deadlocks in networks/distributed applications, communication in distributed systems (among other RPC/RMI and the client server model in more detail), synchronization, reliability, transparency, and atomicity/transaction semantics.
Notes This course is crosslisted with CS 470. Credit at the 600-level requires additional work.

CS 671 - Program Derivation
Credits 3
Introduction to the formal derivation of computer programs from program specifications. Review of the logical and notational prerequisites needed for formal derivation. Guarded commands and the predicate transformer WP. Developing loops from invariants. Program development via sequence of refinements.
Notes This course is crosslisted with CS 471. Credit at the 600-level requires additional work.

CS 672 - Software Product Design and Development I
Credits 3
Current techniques in software design presented with emphasis on architecture first development. Introduction to the processes involved in development. Practice architectural design through a series of homework problems. Students work in teams to prepare the architecture for a software product.
Notes This course is crosslisted with CS 672. Credit at the 600-level requires additional work.
Prerequisites CS 326 and CS 370

CS 673 - Software Product Design II
Credits 3
Synthesis (term project) course to involve students, working in teams, in all of the activities necessary to define, model, implement, test, document, and deliver a program product. Students practice Object-Oriented and Component Based development and utilize UML and CASE tools to model the product and document the process.
Notes This course is crosslisted with CS 473. Credit

at the 600-level requires additional work.
Prerequisites CS 672 or CS 472

CS 674 - Decision Environments for Software Product Development
Credits 3
Term project course to involve students, working in teams, with all of the activities and tools necessary to measure progress and monitor the development of a software product. Students utilize CASE tools for planning, for requirements management, for configuration management, for change management, and for product and process measurement for a product development project.
Notes This course is crosslisted with CS 474. Credit at the 600-level requires additional work.
Prerequisites CS 672 or CS 472

CS 677 - Analysis of Algorithms
Credits 3
Analysis of the time and space complexity of algorithms. Techniques for efficient algorithm design and effect of structure choice on efficiency. Fast algorithms for problems such as set, graph and matrix manipulations, pattern matching, sorting, and storage organization. Exponential time problems and introduction to NP-completeness.
Notes This course is crosslisted with CS 477. Credit at the 600-level requires additional work.
Prerequisites CS 302 and MATH 351

CS 680 - Computer Graphics
Credits 3
Graphics hardware, software and applications. Data structures for graphics, graphics languages, computer-aided design, and three-dimensional graphics.
Notes This course is crosslisted with CS 480. Credit at the 600-level requires additional work.
Prerequisites CS 302 and MATH 365

CS 682 - Artificial Intelligence
Credits 3
Survey of current artificial intelligence technologies: game playing, theorem-proving, natural language processing, pattern recognition, and heuristic programming.
Notes This course is cross listed with CS 482. Credit at the 600 level requires additional work.
Prerequisites CS 302 and PHIL 422

CS 689 - Advanced Computer Science Topics
Credits 3
Undergraduate-level course in advanced topics of computer science, depending upon the interest of faculty and students.
Notes This course is crosslisted with CS 489. Credit at the 600-level requires additional work.

CS 690 - Independent Study
Credits 1-3
Library research and reports on topics of computer science interest. May be repeated for credit with the consent of the Department of Computer Science
Notes This course is crosslisted with CS 490. Credit at the 600-level requires additional work.

CS 715 - Advanced Analysis of Algorithms
Credits 3
Analysis of the complexity and correctness of asymptotically efficient algorithms, including set partitioning, matrix multiplication, integer multiplication and pattern matching algorithms. The theory of NP-completeness; Cook's theorem and polynomial transformations. Basic NP-complete problems, such as the three-satisfactory, three dimensional matching and Hamiltonian circuit problems. PSPACE-completeness results, such as quantified Boolean formulas.
Prerequisites CS 656 and CS 677

CS 717 - Advanced Computer Simulation
Credits 3
Advanced discrete simulation modeling using SIMSCRIPT 11.5 and SLAM. Advanced continuous simulation using ACSL. Modeling concepts, measuring random phenomena. Passive objects, application of simulation to operating systems and software design in general. Digital- analog solution of linear differential equations, industrial dynamics. Feedback systems.
Prerequisites CS 617

CS 718 - Theory of Computation
Credits 3
Computability of functions and sets in terms of Turing machines and other computational models. Universal Turing machines and examples of unsolvable problems. Introduction to other computational models, such as the lambda-calculus, Post systems, Markov algorithms and recursive

function theory. The Church-Turing thesis and proofs of equivalence between the models.
Prerequisites CS 656

CS 719 - Advanced Automata and Formal Languages
Credits 3
Extensive study of context-sensitive, recursive and recursively enumerable languages, including ambiguity and closure properties: decidable and undecidable properties of the different language classes: the halting problem and Post's correspondence problem; properties of the deterministic context-free languages; LR(k) and LL(k) grammars.
Prerequisites CS 656

CS 733 - Geographic Data Base Systems
Credits 3
Spatial data types and operators: point queries, range queries, translation, rotation, and scaling. Data structures for object representation: arc tree, quadtrees. Commercial data bases vs. spatial data bases: relational, hierarchical, network.
Notes (May not be used to satisfy degree requirements in Computer Science.)
Prerequisites CS 135 or CS 117 or equivalent and STAT 611

CS 740 - Statistical Pattern Recognition
Credits 3
Concepts and formal theoretical structures necessary for design and implementation of a pattern recognition system. Topics include: parametric and non-parametric methods, linear and non-linear classifiers and clustering algorithms.
Prerequisites STA 667, MATH 253 or 265, and CS 302

CS 741 - Structural Pattern Recognition
Credits 3
Survey of advanced pattern recognition techniques. Topics include: graph matching methods, syntactic approaches, neural nets, and context-dependent methods.
Prerequisites CS 656 and CS 677

CS 742 - Document Image Understanding
Credits 3
Survey of document understanding methods and related topics that include: data compression, document exchange standards, layout analysis methods, logical analysis methods, OCR, error correction, and document routing.
Prerequisites CS 740 and CS 669

CS 747 - Cryptography and Information Theory
Credits 3
Cryptography, cryptographic systems, encryption algorithms, cryptographic techniques, access control, lattice model of information flow, flow control mechanisms, inference control mechanisms, mechanisms restricting noise, mechanisms restricting statistics, statistical database models.
Prerequisites CS 370, STAT 411

CS 750 - Computational Algorithms in VLSI
Credits 3
Application and inherent limitations of using VLSI to implement computational algorithms, design and analysis of algorithms for design of VLSI circuits, introduction to VLSI implementation of computational algorithms represented by logic circuits, lower bounds on area and time, systolic arrays and their applications, VLSI layout algorithms, VLSI test generation and simulation.
Prerequisites CS 677

CS 754 - Discrete Optimization
Credits 3
Network optimization problems, use of advanced data structures. Topics may vary and include maximum-flow algorithms, multiterminal maximum flows, minimum cost flows and circulations, matching algorithms, approximation algorithms, and applications. Hamiltonian circuits in dense graphs, disjoint paths, the postman problem, introduction to combinatorial geometry, and linear programming.
Prerequisites CS 677

CS 756 - Formal Semantics
Credits 3
Coverage of formal methods for defining the semantics of programming languages, including the operational, denotation and axiomatic approaches. Proof techniques for verifying properties of programs. Consistent and complementary definitions for a Pascal-like language discussed.
Prerequisites CS 326 and CS 656

CS 758 - Computational Geometry
Credits 3
Geometric searching, point location, range searching, convex hull, Graham's scan, gift wrapping, dynamic convex hull, proximity closest pair, Voronoi diagram, triangulation. Intersection, visibility shortest paths, geometry of rectangles.
Prerequisites CS 677

CS 763 - Advanced Computer Architecture
Credits 3
Advanced study of various current computer architectures. Examples taken from specialized architectures that support modern general-purpose programming, operating systems, artificial intelligence and data bases. SIMD and MIMD parallel architectures.
Prerequisites CS 326 and CS 663

CS 767 - Advanced Computer Graphics
Credits 3
Hidden line elimination algorithms and implementation. Perfect interpolators, cubic and bicubic splines, Kriging, Hermite surfaces, nonperfect interpolators, Bezier curves and surfaces, B-splines, ray tracing algorithms, shading, lightness, motion, moving pictures, two- and three- dimensional fractals. Special topics.
Prerequisites CS 680

CS 768 - Surface Estimation for Computer-Aided Geometric Design
Credits 3
Affine maps, function spaces, the DeCasteljan algorithm, Bernstein polynomials, Bezier surfaces, nonparametric curves, Lagrange polynomials, C continuity, B-spline basis, Frenet frame, G continuity, gamma splines, beta splines, geometric continuity, tensor product interpolants, volume deformations, curvatures.
Prerequisites CS 767

CS 769 - Advanced Data Base Management
Credits 3
Continuation of CS 632, including normalization of relational data bases using functional and multivalued dependencies. Query processing, query interpretation, query optimization, and methods for implementing and optimizing logic queries. Knowledge data bases, distributed data bases and object-oriented data bases.
Prerequisites CS 657

CS 770 - Advanced Operating Systems
Credits 3
Study of the design principles, organization, and performance analysis of large-scale computer operating systems. Particular subjects emphasized include coordination of tasks, solutions of deadlock problems, theories of segmentation and paging, and performance prediction.
Prerequisites CS 370

CS 771 - Concurrent Computation
Credits 3
Study of concurrent programming methods and applications; event spaces; models of concurrency, such as Petri nets, CCS and CSP. Synchronization, data sharing and communication. Concurrency constructs in various programming languages. Scheduling and implementation techniques. Applications of concurrency in operating system design, fault-tolerance, and reliability.
Prerequisites CS 326 and CS 370.

CS 772 - Software Architecture
Credits 3
Survey of advanced techniques for specifying and designing large software systems. System verification. Reliability and project management.
Prerequisites CS 370, CS 672, and CS 660, or consent of instructor.

CS 777 - Parallel Algorithms
Credits 3
Methods for creating and analyzing parallel algorithms. Parallel programming languages and programming models of shared-memory and distributed architectures. Measuring complexity of parallel algorithms. NC-class versus P-class algorithms.
Prerequisites CS 677

CS 778 - Advanced Translation
Credits 3
Formal semantics, automatic compiler generation, attribute grammars. Language issues as they relate to compiler generation.
Prerequisites CS 660

CS 779 - Supercompliers for Parallel and Vector Computers
Credits 3
Dependence analysis, Diophantine equations, the GCD test, the Banerjee test, do-loop normalization, concurrency in loops, vector code generation, control dependence and vectorization, parallel code generation for doall-loops, parallel code generation for doacross-loops, shared memory parallelization, parallelization for distributed memory architectures.
Prerequisites CS 778

CS 780 - Distributed Computing and Algorithms
Credits 3
Methods and algorithms of distributed computing. Topics may include architecture and design goals, formal approaches to distributed computing problems, networks and protocols, models of distributed computing, synchronization and communication, synchronous and asynchronous systems, fault-tolerance and reliability, self-stabilization, distributed algorithms and applications.
Prerequisites CS 370, CS 677

CS 781 - Automated Deduction
Credits 3
Use of computers for forming deductions and proving theorems in symbolic logic covered. Topics include resolution, unification, proof strategies, and equality. Also examines areas of application: problem solving, question answering, program verification, automatic programming and logic programming (Prolog).
Prerequisites CS 682

CS 782 - Expert System Construction
Credits 3
Design, organization, and construction of expert systems. Includes general concepts, characteristics, elements, advantages, and examples of expert systems. Also rule-based knowledge representations, inference techniques, implementation tools and shells, and advanced topics.
Prerequisites CS 682

CS 783 - Genetic Algorithms and Neural Networks
Credits 3
A study of the utility of adaptive methods and their limitations across optimization problems spanning areas of engineering. Topics include genetic algorithms and genetic programming, simulated annealing, tabu search, neural networks, artificial life. Use of software tools for implementations.

CS 785 - Computational Linguistics
Credits 3
Introduction to linguistics and computational linguistics, for natural language. Phonology, morphology, syntax, semantics, and lexicology. Text analysis and processing; construction of lexicons, and indexes and concordances. Introduction to text retrieval, translation, speech understanding and generation.
Prerequisites CS 656

CS 786 - Advanced Computational Linguistics
Credits 3
Advanced study of computational linguistics. Emphasis on cognitive methods in natural language understanding and generation. Pragmatics and discourse.
Prerequisites CS 785

CS 788 - Computational Environmetrics
Credits 3
Applications of sensor networks and pattern recognition to environmental problems. Geometric pattern recognition: metrics for comparing 2-d shapes, signature functions, turning functions. Geometric algorithms in sensor networks. Position based routing, face routing, broadcasting and multi-casting. Interference aware sensor networks. Data gathering and target recognition. Prototype implementation.
Prerequisites Consent of instructor.

CS 789 - Topics in Advanced Computer Science
Credits 3
Graduate-level course in some field of computer science, at advanced level, depending upon the current interest of the staff and the students.
Notes May be repeated with a different subject matter to a maximum of nine credits.
Prerequisites Consent of instructor.

CS 790 - Master's Project
Credits 1 – 3
Notes May be repeated, but only three credits will be applied to the student's program.
Grading S/F grading only.
Prerequisites Consent of instructor.

CS 791 - Thesis
Credits 3 – 6
Notes May be repeated, but only six credits will be applied to the student's program.
Grading S/F grading only.
Prerequisites Consent of instructor.

CS 792 - Research Seminar
Credits 1
Oral presentation of assigned articles.
Notes May be repeated to a maximum of four credits.
Prerequisites Consent of instructor.

CS 799 - Dissertation Research
Credits 1 – 6
Research analysis and writing towards completion of dissertation and subsequent defense.
Notes May be repeated but no more than 18 credits will be allowed in the degree.
Grading S/F grading only.
Prerequisites Graduate standing in Ph.D. program and consent of advisor.

INF 730 - Human Computer Interaction
Credits 3
Covers the fundamental concepts and techniques for design, implementation, and evaluation of human computer interfaces. Topics include Foundations of Human computer interaction, design and implementation techniques for graphical user interfaces, evaluation techniques, and different interface models.
Prerequisites Consent of instructor.

INF 731 - Advanced HCI - Design and Implementation
Credits 3
This course is organized around readings that reinforce the student's knowledge in Human Computer Interaction guidelines, principles, and theories and mainly around projects that allow students to apply theoretical knowledge to the design, implementation, and evaluation of interactive computer systems.
Prerequisites INF 730

INF 732 - Affectively Intelligent Systems
Credits 3
Focuses on computational emotion modeling which spawns from a variety of interest: improving basic understanding of the functional role of emotions in humans; integrating emotion recognition and prediction techniques; synthesizing emotion and expression of emotion to apply to synthetic characters, autonomous software agents or robots; understanding social implications of affective information and communication technology.
Prerequisites INF 700

INF 740 - Digital Media Design, Technology and Representation
Credits 3
Covers principles of design to visualize new media concepts in any medium. Exposes students to new and emerging digital media technologies and applications.
Prerequisites Consent of instructor.

INF 760 - Advanced Theoretical Foundations of Informatics
Credits 3
Advanced course to cover mathematical methods for information modeling, analysis, and manipulation. Requires various research article reading and discussions. Topics include proof techniques, first-order logic, computability theory, complexity theory, model theory, and statistics.
Prerequisites INF 700

INF 770 - Social Foundations of Informatics
Credits 3
Covers the relationships between social systems and information and communication technologies. Focuses on social factors that influence the organization of information technologies in social and organizational systems, and how the human social factors and technological tools mutually contribute to the field of Informatics. Prerequisite: Consent of instructor.

INF 780 - Special Topics in Informatics
Credits 3
Emphasis is on new developments and research in science, humanities, fine arts, and other domain informatics.
Prerequisites INF 700

INF 790 - Informatics Project
Credits 3
Advanced project in informatics.

Notes May be repeated for different project topics, but only three credits will be applied to the student's program.
Prerequisites INF 700 and consent of instructor.

INF 792 - Internship
Credits 3
Supervised internship in business, industry, government, or educational institution providing practical experience to use skills and knowledge acquired in informatics and cognate course work.
Prerequisites INF 700 and consent of instructor.

INF 794 - Research Methods
Credits 3
Examination of research methods including: the scientific method, sampling, statistics, research design, analytical technique, literature review, technical writing, professional ethics, faculty research areas and potential topics for thesis.
Prerequisites INF 700

INF 795 - Independent Study in Informatics
Credits 1-6
Supervised independent work in a topic of Informatics.
Notes May be repeated but no more than 6 credits will be allowed in the degree.
Grading S/F grading only
Prerequisites INF 700 and Instructor consent

INF 797 - Master's Thesis
Credits 1-6
Research analysis and writing towards completion of Master's thesis and subsequent defense.
Notes May be repeated but no more than 6 credits will be allowed in the degree.
Grading S/F grading only
Prerequisites INF 700 and Instructor consent

INF 799 - Dissertation Research
Credits 1 – 6
Research analysis and writing towards completion of dissertation and subsequent defense.
Notes May be repeated but no more than eighteen credits will be allowed in the degree.
Prerequisites Passing the written comprehensive examination.

APPENDIX 2: Admission Requirements

UNLV | Howard R. Hughes College of ENGINEERING

Doctor of Philosophy in Computer Science
Admission to the Major

To enter the Doctor of Philosophy in Computer Science, a student must be admitted to the UNLV Graduate College. One can visit the web page of UNLV Graduate College (www.unlv.edu/graduatecolelge) for Graduate College admission requirements, application and deadlines.

Applicants for admission to the Ph.D. program in computer science must meet the following:

1. A GPA of 3.70 (on a 4.00 scale) or higher in post-baccalaureate course work is required for admission. Students entering with a bachelor's degree must have a GPA of 3.5 or higher for the courses at the 200-level or above.
2. Students are expected to have a master's degree in computer science before applying to the Ph.D. program. On rare occasions, an unusually capable student may be admitted to work directly for the Ph.D. degree without having a master's degree.
3. At least three letters of recommendation (preferably from academic sources) attesting to the applicant's professional competence and academic potential are required.
4. A personal statement of purpose, which should be as specific as possible and should include the applicant's objectives and area(s) of interest, is required.
5. A minimum score of 315 on the general test of the Graduate Record Examination (GRE) is required. Official score reports from the last five years are acceptable.
6. All domestic and international applicants must review and follow the Graduate College Admission and Registration Requirements.

Students are accepted into a degree program as described in the Graduate Catalog. The faculty and corresponding sub-disciplines and sub-plans within the described programs are subject to change at any time.

UNLV | Howard R. Hughes College of ENGINEERING

Master of Science in Computer Science
Admission to the Major

To enter the Master of Science in Computer Science, a student must be admitted to the UNLV Graduate College. One can visit the web page of UNLV Graduate College (www.unlv.edu/graduatecolelge) for Graduate College admission requirements, application and deadlines.

Applicants must submit the following to the Graduate College:

1. An application and official transcripts of all college level work with a minimum GPA of 3.00.
2. Two letters of recommendation concerning the student's potential for succeeding in the graduate program.
3. Another set of official transcripts.
4. The results of the Graduate Record Examination current to within five years should be sent directly to the school.
5. In addition, applicants must have completed courses and their prerequisites equivalent to our undergraduate Programming Languages CS 326), Operating Systems (CS 370), Discrete Mathematics II (MATH 351), and Statistical Methods I (STAT 411) with an average grade of B or better.
6. The Computer Science Admission Committee may elect to admit an outstanding applicant who has not satisfied all of the background requirements on a conditional basis.
7. All domestic and international applicants must review and follow the Graduate College Admission and Registration Requirements.

The student must complete these requirements before full admission to the program is granted.

Students who have not completed all the following courses (or equivalent courses) as part of their bachelor's degree may be required to complete them as a condition of their admission. If taken as part of their master's degree program, these courses may count toward the 30 credits required.

CS 656 - Automata and Formal Languages

CS 677 - Analysis of Algorithms

CS 660 - Compiler Construction

Students are accepted into a degree program as described in the Graduate Catalog. The faculty and corresponding sub-disciplines and sub-plans within the described programs are subject to change at any time.

UNLV

Office of Academic Assessment

Annual Academic Assessment Report Cover Sheet

Assessment reports are due the 1st Wednesday after the Fall Term

Program Information:

Program Assessed	Doctor of Philosophy
Department	Computer Science
College	Engineering
Department Chair	Dr. Laxmi Gewali
Assessment Coordinator	Dr. Ajoy Datta
Date Submitted	2/19/15
Contact Person for This Report	
Name	Ajoy Datta
Phone	702-895-0870
Email	Ajoy.datta@unlv.edu

Please attach a narrative (not to exceed 4 pages, excluding appendices) addressing the following:

- What are the student learning outcomes? Please provide a numbered list.
- Which learning outcomes were assessed?
- How were they assessed? (Programs must use at least one direct assessment of student learning.)
- Undergraduate programs should assess at least one University Undergraduate Learning Outcome (UULO) each year, which may or may not overlap with a program learning outcome.
- Graduate programs should assess at least one outcome related to one of the following graduate level requirements each year:
    - student engagement in research, scholarship, creative expression and/or appropriate high-level professional practice.
    - activities requiring originality, critical analysis and expertise.
    - the development of extensive knowledge in the field under study.

- What was learned from the assessment results?
- How did the program respond to what was learned?

Please limit the narrative portion of your report to no more than four pages. You may attach appendices with data, tables, charts, or other materials as needed. Please explain the relevant conclusions from any appendices in your narrative. . Please contact the Office of Academic Assessment if you have questions or need assistance.

For the PhD degree in computer science we have specified the following student outcomes:

1.- Exhibit a breadth of knowledge in the areas of algorithms, programming languages and compilers, theory, operating systems, and computer architecture.

2.- Exhibit a depth of knowledge in at least one specialized area of computer science.

3.- Conduct a thorough literature survey on a research topic.

The following learning outcomes were assessed:

- Written Dissertation
- Oral defense of the dissertation
- Oral Qualifying examination on Research Topic

They were assessed:

- Written dissertation submitted to a CS faculty committee. The results are collected on how well the dissertation is written and the description of significant research.
- A complete, well-research, well-written dissertation describing a significant research contribution to the field of Computer Science. Must be publishable in a referred journal.
- Students present their dissertation in an oral presentation to the CS faculty and Graduate students. Results are collected on how well the student presents.
- A well-organized oral presentation describing and defending the work done by the student on his/her dissertation. The student must also be able to field and answer questions on the presentation from the audience.
- Student presents his/her topic for dissertation.
- Student will present a dissertation proposal containing a background survey on the problem chosen, a discussion of the research objectives and approach to be used, and a list of references. The proposal will be judged based on its completeness, technical merit, and feasibility.

Assessing Student Engagement in Research

Students research work in Dissertation research was assessed by the student's dissertation committee. The initial dissertation proposal and its presentation were assessed by the committee. Their published articles were also evaluated. Finally, the committee assessed the final dissertation report and the presentation.

Assessment results:

The outcome is definitely above of what was expected. Our number of Ph D students, have triple. The students are performing very well and making good progress.

Program response:

Based on the assessment outcomes, no program changes will be taken at this time. The responsibility is shared by the CS graduate committee. Any changes are subject to a vote by the CS department tenure faculty.
The CS department is alerting and encouraging faculty members to attend workshops on assessment processing both outside and inside UNLV

UNLV

Office of Academic Assessment

Annual Academic Assessment Report Cover Sheet

Assessment reports are due the 1st Wednesday after the Fall Term

Program Information:

Program Assessed	Master of Science
Department	Computer Science
College	Engineering
Department Chair	Dr. Laxmi Gewali
Assessment Coordinator	Dr. Ajoy Datta
Date Submitted	2/19/15
Contact Person for This Report	
Name	Ajoy Datta
Phone	702-895-0870
Email	Ajoy.datta@unlv.edu

Please attach a narrative (not to exceed 4 pages, excluding appendices) addressing the following:

- What are the student learning outcomes? Please provide a numbered list.
- Which learning outcomes were assessed?
- How were they assessed? (Programs must use at least one direct assessment of student learning.)
- Undergraduate programs should assess at least one University Undergraduate Learning Outcome (UULO) each year, which may or may not overlap with a program learning outcome.
- Graduate programs should assess at least one outcome related to one of the following graduate level requirements each year:
    - student engagement in research, scholarship, creative expression and/or appropriate high-level professional practice.
    - activities requiring originality, critical analysis and expertise.
    - the development of extensive knowledge in the field under study.

- What was learned from the assessment results?
- How did the program respond to what was learned?

Please limit the narrative portion of your report to no more than four pages. You may attach appendices with data, tables, charts, or other materials as needed. Please explain the relevant conclusions from any appendices in your narrative. . Please contact the Office of Academic Assessment if you have questions or need assistance.

For the MS degree in computer science we have specified the following student outcomes:

1.- Acquires in-dept knowledge of specialized areas and advanced topics in computer science.
2.- Independently analyze, design, and implement an innovative computer application or research project.
3.- Prepare a final project or research report on the solution to a computer-related problem.
4.- Present the results of their research orally.
The Computer Science master's program gives you the opportunity to study different areas, including:

- Design and analysis of algorithms
- Operating and distributed systems
- Computer architecture and networking
- Computational geometry and robotics
- Computer graphics and image processing
- Programming languages and compiler construction
- Artificial intelligence and expert systems
- Database design, document analysis, and retrieval Software engineering

The following learning outcomes were assessed:

- Prepare a final project or research report on the solution to a computer-related problem.
- Present the results of their research orally

They were assessed:

- Written report submitted to a CS faculty committee. The results are collected on how well the project is written and the new technique is described.
- Students present their Thesis or Project report in an oral presentation to the CS faculty and Graduate students. Results are collected on how well the student presented.

Assessing Student Engagement in Research

Students research work in Independent Study class and Thesis research was assessed by the student's thesis committee. Finally, the committee also assessed the final report and the presentation.

Assessment results:

- Student thesis/project reports and oral examinations indicated the students have met all the desired learning outcomes. The MSCS Program is performing very well.
- Based on the outcomes, no program changes will be taken at this time. The responsibility is shared by the CS graduate committee. Any changes are subject to a vote by the CS department tenure faculty.

Respond to what was learned:

Based on assessments, no recommendations for changes to the MS program were made.
The CS department is encouraging faculty members to attend workshops on assessment processing both outside and inside UNLV.