|  |  |
| --- | --- |
| For Academic Affairs and Research Use Only | |
| CIP Code: |  |
| Degree Code: |  |

**New Course Proposal Form**

**[ ] Undergraduate Curriculum Council**

**[X] Graduate Council**

|  |
| --- |
| **[X] New Course or [ ]Experimental Course (1-time offering) (Check one box)** |

Signed paper copies of proposals submitted for consideration are no longer required. Please type approver name and enter date of approval.

Email completed proposals to [curriculum@astate.edu](mailto:curriculum@astate.edu) for inclusion in curriculum committee agenda.

|  |  |
| --- | --- |
| Edward Hammerand 9/24/2017 **Department Curriculum Committee Chair** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Enter date…  **COPE Chair (if applicable)** |
| Hung-Chi Su 9/24/2017 **Department Chair:** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Enter date…  **Head of Unit (If applicable)** |
| David F. Gilmore 10/6/2017 **College Curriculum Committee Chair** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Enter date…  **Undergraduate Curriculum Council Chair** |
| Anne A. Grippo 10/6/2017 **College Dean** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Enter date…  **Graduate Curriculum Committee Chair** |
| |  |  | | --- | --- | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ | Enter date |   **General Education Committee Chair (If applicable)** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Enter date…  **Vice Chancellor for Academic Affairs** |

1. Contact Person (Name, Email Address, Phone Number)

Dr. Hai Jiang, hjiang@astate.edu, (870)680-8164

2. Proposed Starting Term and Bulletin Year

Spring, 2018

3. Proposed Course Prefix and Number (Confirm that number chosen has not been used before. For variable credit courses, indicate variable range. *Proposed number for experimental course is 9*. )

CS 6123

4. Course Title – if title is more than 30 characters (including spaces), provide short title to be used on transcripts. Title cannot have any symbols (e.g. slash, colon, semi-colon, apostrophe, dash, and parenthesis). Please indicate if this course will have variable titles (e.g. independent study, thesis, special topics).

Software Security

5. Brief course description (40 words or fewer) as it should appear in the bulletin.

Study of security issues in the software development process, including security management, secure software development lifecycle, language security, and web application security.

6. Prerequisites and major restrictions. (Indicate all prerequisites. If this course is restricted to a specific major, which major. If a student does not have the prerequisites or does not have the appropriate major, the student will not be allowed to register).

1. **Yes** Are there any prerequisites?
   1. If yes, which ones?

CS3113 or “B” or better in CS5032, and CS3233

* 1. Why or why not?

The material covered by the course requires understanding of advanced concepts in data structures (CS3113 or CS5032) and Operating Systems (CS3233).

1. **No** Is this course restricted to a specific major?
   1. If yes, which major? Enter text...

7. Course frequency(e.g. Fall, Spring, Summer). *Not applicable to Graduate courses.*

Enter text...

8. Will this course be lecture only, lab only, lecture and lab, activity, dissertation, experiential learning, independent study, internship, performance, practicum, recitation, seminar, special problems, special topics, studio, student exchange, occupational learning credit, or course for fee purpose only (e.g. an exam)? Please choose one.

Lecture only

9. What is the grade type (i.e. standard letter, credit/no credit, pass/fail, no grade, developmental, or other [please elaborate])

Standard letter

10. **No** Is this course dual listed (undergraduate/graduate)?

11. **No** Is this course cross listed?

*(If it is, all course entries must be identical including course descriptions. Submit appropriate documentation for requested changes. It is important to check the course description of an existing course when adding a new cross listed course.)*

**11.1** – If yes, please list the prefix and course number of cross listed course.

Enter text...

**11.2** – **Yes / No** Are these courses offered for equivalent credit?

Please explain. Enter text...

12. **Yes** Is this course in support of a new program?

a. If yes, what program?

Cyber Security Certificate in Computer Science

13. **No** Does this course replace a course being deleted?

a. If yes, what course?

Enter text...

14. **No** Will this course be equivalent to a deleted course?

a. If yes, which course?

Enter text...

15. **Yes** Has it been confirmed that this course number is available for use?

*If no: Contact Registrar’s Office for assistance.*

16. **No** Does this course affect another program?

If yes, provide confirmation of acceptance/approval of changes from the Dean, Department Head, and/or Program Director whose area this affects.

Enter text...

**Course Details**

17. Outline (The course outline should be topical by weeks and should be sufficient in detail to allow for judgment of the content of the course.)

Week 1: Overview

Week 2: Security Management and Risk Assessment

Week 3: Security Controls, Plans and Procedures

Week 4: Human Resource Security

Week 5: Legal and Ethical Aspects

Week 6: Threat Modeling

Week 7: Secure Software Development Lifecycle

Week 8: Secure Software Development Lifecycle

Week 9: Firewalls and Intrusion Prevention Systems

Week 10: C Language Security

Week 11: C Language Security

Week 12: Penetration Testing

Week 13: Web Application Security

Week 14: Web Application Security

18. Special features (e.g. labs, exhibits, site visitations, etc.)

N/A

19. Department staffing and classroom/lab resources

Enter text...

1. Will this require additional faculty, supplies, etc.?

No. The course has been in the rotation for several semesters as a special topics subject. Its addition to the bulletin as a regular course will have no impact on department staffing or resources.

20. **No** Does this course require course fees?

*If yes: please attach the New Program Tuition and Fees form, which is available from the UCC website.*

**Course Justification**

21. Justification for course being included in program. Must include:

a. Academic rationale and goals for the course (skills or level of knowledge students can be expected to attain)

Since software security is a central and critical aspect of a secure system, software design, development, and analysis are top priority issues in cyber security. This new course will prepare students to be security programmers and software security analysts. Such professionals are in high demand in industry

b. How does the course fit with the mission established by the department for the curriculum? If course is mandated by an accrediting or certifying agency, include the directive.

One part of the mission is to maintain the curriculum with updated technologies. The course addresses the department’s ongoing need to add curriculum reflecting new concepts and technologies in computer science. While the department curriculum has included data security for some time, the rapid growth in security issues in recent years brings great attention and challenges to computer science academia and industry in the developing field of software security. The proposed course will address security issues during software design and the development process, strengthening the department’s curriculum.

c. Student population served.

Graduate.

d. Rationale for the level of the course (lower, upper, or graduate).

Students should have comprehensive understanding about computer systems as well as programming skills.

**Assessment**

**University Outcomes**

22. Please indicate the university-level student learning outcomes for which this new course will contribute. Check all that apply.

|  |  |  |
| --- | --- | --- |
| * 1. **[ ]** Global Awareness | * 1. **[X ]** Thinking Critically | * 1. **[X ]** Information Literacy |

**Relationship with Current Program-Level Assessment Process**

23. What is/are the intended program-level learning outcome/s for students enrolled in this course? Where will this course fit into an already existing program assessment process?

1. M.S. Computer Science graduate students should have a deeper understanding of the theory and application of algorithms, programming languages, and computer processes.
2. M.S. Computer Science graduate students should have the ability to apply advanced analysis techniques to problem identification and solution in computing applications.
3. M.S. Computer Science graduate students should have the ability to apply advanced implementation techniques to problem identification and solution in computing applications.

The course will be assessed along with other graduate courses on the same schedule.

24. Considering the indicated program-level learning outcome/s (from question #23), please fill out the following table to show how and where this course fits into the program’s continuous improvement assessment process.

*For further assistance, please see the ‘Expanded Instructions’ document available on the UCC - Forms website for guidance, or contact the Office of Assessment at 870-972-2989.*

|  |  |
| --- | --- |
| **Program-Level Outcome 1 (from question #23)** | M.S. Computer Science graduate students should have a deeper understanding of the theory and application of algorithms, programming languages, and computer processes. |
| Assessment Measure | Comprehensive examinations and employer surveys |
| Assessment  Timetable | Comprehensive exams will be conducted each semester, reviewed annually, and reported on every three years; employer surveys will be conducted each fall and reported on every four years. |
| Who is responsible for assessing and reporting on the results? | Department assessment committee |

|  |  |
| --- | --- |
| **Program-Level Outcome 2 (from question #23)** | M.S. Computer Science graduate students should have the ability to apply advanced analysis techniques to problem identification and solution in computing applications. |
| Assessment Measure | Comprehensive examinations and employer surveys |
| Assessment  Timetable | Comprehensive exams will be conducted each semester, reviewed annually, and reported on every three years; employer surveys will be conducted each fall and reported on every four years. |
| Who is responsible for assessing and reporting on the results? | Department assessment committee. |

|  |  |
| --- | --- |
| **Program-Level Outcome 3 (from question #23)** | M.S. Computer Science graduate students should have the ability to apply advanced implementation techniques to problem identification and solution in computing applications. |
| Assessment Measure | Comprehensive examinations and employer surveys |
| Assessment  Timetable | Comprehensive exams will be conducted each semester, reviewed annually, and reported on every three years; employer surveys will be conducted each fall and reported on every four years. |
| Who is responsible for assessing and reporting on the results? | Department assessment committee. |

**Course-Level Outcomes**

25. What are the course-level outcomes for students enrolled in this course and the associated assessment measures?

|  |  |
| --- | --- |
| **Outcome 1** | Students will become familiar with the requirements for secure programming. |
| Which learning activities are responsible for this outcome? | In-class discussion and illustrations  Demonstration of analysis results in presentations |
| Assessment Measure | Course presentations, exams and projects |

|  |  |
| --- | --- |
| **Outcome 2** | Students will master the corresponding analysis skills. |
| Which learning activities are responsible for this outcome? | Accomplish related literature reviews  Demonstrate the analysis results in presentations and exams |
| Assessment Measure | Course exams and projects |

|  |  |
| --- | --- |
| **Outcome 3** | Students will get experience which will develop implementation skills. |
| Which learning activities are responsible for this outcome? | In-class discussion and illustrations  Accomplish related literature reviews  Conduct effective projects |
| Assessment Measure | Course homework, presentations, exams and projects |

*(Repeat if needed for additional outcomes)*

**Bulletin Changes**

|  |
| --- |
| **Instructions** |
| **Please visit** [**http://www.astate.edu/a/registrar/students/bulletins/index.dot**](http://www.astate.edu/a/registrar/students/bulletins/index.dot) **and select the most recent version of the bulletin. Copy and paste all bulletin pages this proposal affects below. Follow the following guidelines for indicating necessary changes.**  **\*Please note: Courses are often listed in multiple sections of the bulletin. To ensure that all affected sections have been located, please search the bulletin (ctrl+F) for the appropriate courses before submission of this form.**  - Deleted courses/credit hours should be marked with a red strike-through (~~red strikethrough~~)  - New credit hours and text changes should be listed in blue using enlarged font (blue using enlarged font).  - Any new courses should be listed in blue bold italics using enlarged font (***blue bold italics using enlarged font***)  *You can easily apply any of these changes by selecting the example text in the instructions above, double-clicking the ‘format painter’ icon 🡪 , and selecting the text you would like to apply the change to.*  *Please visit* [*https://youtu.be/yjdL2n4lZm4*](https://youtu.be/yjdL2n4lZm4) *for more detailed instructions.* |

**2017-2018 Graduate Bulletin**

**Page 375**

**CS 5713. Analysis of Algorithms** The analysis of space and time requirements of algorithms. Worst-case and average case studies. Greedy algorithms and divide-and-conquer algorithms examined. Tractable and intractable algorithms surveyed. Prerequisites: CS 3113 or “B” or better in CS 5032, and MATH 2214.

**CS 5723. Automata Theory** Study formal languages and equivalent models of computation. Finite state automata and regular expressions. Push-down automata and context free grammars. Pumping lemmas and closure properties. Turing Machines. Prerequisite: CS 3113 or “B” or better in CS 5032.

**CS 5823. Scripting Languages** Examination of scripting languages compared to conventional programming languages and construction of domain-specific solutions for common problems in GUI, networking, and web programming. Prerequisite: CS 3113 or “B” or better in CS 5032.

**CS 583V. Internship** Supervised work experience participating in application system development in a business/manufacturing environment. Grade earned will be pass or fail. Prerequisites: Permission of the Computer Science faculty, CS 3113 or “B” or better in CS 5032, and either CS 3123 or CS 5113.

***CS 6123. Software Security* Study of security issues in the software development process, including security management, secure software development lifecycle, language security, and web application security*. Prerequisite: CS 3113 or “B” or better in CS 5032, and CS3233.***

**CS 6213. Parallel Processing** Parallel processing and supercomputer architecture with emphasis on efficient utilization of resources. Prerequisite: CS 3223, or “B” or better in CS 5032 and permission of professor.

**CS 6313. Data Security** Methods for protection, security, and privacy of data; access controls, authentication, cryptographic controls, information flow controls, security kernels. Security of data in networks. Prerequisite: CS 3233 or “B” or better in CS 5032.   
  
**CS 6413. Solid Modeling** Examination of advanced modeling techniques with emphasis on radiosity. Techniques for rapid interactive display of a complex three-dimensional environment will be developed. Prerequisite: CS 3113 or “B” or better in CS 5032 or CS 5423.