**WID Course Proposal Form**

Please read the *WID Course Proposal Guidelines* before filling in this form. The form should be submitted to the Director of the Writing in the Disciplines program, preferably by email. The current WID Director is Prof. Chris Sten ([csten@gwu.edu](mailto:csten@gwu.edu)). The University Writing Advisory Committee (UWAC) will evaluate your form and communicate with you through the WID Director.

### Part A: Administrative

<table>
<thead>
<tr>
<th><strong>Name:</strong></th>
<th>Valentina Harizanov</th>
<th><strong>Department:</strong></th>
<th>Mathematics</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Email:</strong></td>
<td><a href="mailto:harizanv@gwu.edu">harizanv@gwu.edu</a></td>
<td><strong>Phone:</strong></td>
<td>202-994-6595</td>
</tr>
<tr>
<td><strong>Course number:</strong></td>
<td>Math 103</td>
<td><strong>Number of credits:</strong></td>
<td>3</td>
</tr>
<tr>
<td><strong>Prerequisites:</strong></td>
<td>Math 32 or permission of instructor</td>
<td><strong>Associated labs, if any:</strong></td>
<td>none</td>
</tr>
<tr>
<td><strong>Title:</strong></td>
<td>Computability Theory</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Course offered in (circle all that apply):**  
- Fall  
- Odd Years  
- Spring  
- Summer

**Course cross-listed with (if any):**  
N/A

**Class size:**  
20

**Number of seats for WID students:**  
10

**Number of GTA's supporting each section:**  
none

### Part B: Course Details

**B1. Brief course/catalog description:**

*Math 103: Computability Theory*

The unlimited register machine as a model of an idealized computer. Computable functions, Church’s thesis. Effective enumerability. Unsolvability of the halting problem and other theoretical limitations on what computers can do.

**B2. Is this a new course? Note: new courses need to be approved through regular channels in each school (for example, the curriculum committee in CCAS). UWAC only evaluates the content relevant to WID.**
B3. For each assignment involving writing, describe the type of writing, approximate due date (in weeks counting from the start of the semester), length (in pages), whether it’s a revision of a previous draft, and whether it will be peer-reviewed. Use fewer or more rows in table as appropriate or any other format to describe the same information. The second column should indicate the type of writing or intended audience, for example: journal article, review, expository article, report, short story, among others - see FAQ 6 in the guidelines for a more complete list.

Assignment #1: Devise URM-programs for computing functions on natural numbers and solve more abstract problems related to the power of the unlimited register machines.
   (i) Write mathematical proofs using the convention of the “programming” URM-language, 3 pages, due week 2.
   (ii) First revision, 3 pages, peer reviewed, due week 3.

Assignment #2: Show recusiveness and partial recursiveness of important functions and relations in mathematics and cryptography, using composition, primitive recursion and minimalization.
   (i) Write stylized proofs using the symbolic languages with the first-order quantifiers and relation and function symbols, 3 pages, due week 4.
   (ii) First revision, 3 pages, peer reviewed, due week 5.

Assignment #3: Develop computability theory using Church-Turing’s Thesis, Kleene’s Fixed Point Theorem and Parameter Theorem.
   (i) Write sketches of solutions and short essays establishing the main ideas, 2 pages, due week 6.
   (ii) First revision (detailed solutions and essays establishing exact arguments), 4 pages, due week 7, peer reviewed.
   (iii) Second revision (polished solutions and complete arguments), 4 pages, due week 8.

Assignment #4: Develop the properties of computably enumerable sets and the theory of undecidable problems.
   (i) Write sketches of solutions and short essays establishing the main ideas, 2 pages, due week 9.
   (ii) First revision (detailed solutions and essays establishing exact arguments), 4 pages, due week 10, peer reviewed.
   (iii) Second revision (polished solutions and complete arguments), 4 pages, due week 11.

Assignment #5: Final assignment that reviews the most important concepts, facts and methods.
   (i) Write sketches of solutions and short essays, and a biographical sketch of a selected computability theorist, 6 pages, due week 12.
   (ii) First revision, 6 pages, due week 14, peer reviewed.
   (iii) Second revision, 6 pages, due during finals.

B4. Total number of pages to be revised once:

20 pages

B5. Total number of pages to be revised twice:

14 pages

B6. If your course assigns significantly fewer pages than recommended (see guidelines, especially FAQ’s 1-6) or offers fewer opportunities for draft revision, explain why it should be considered a WID course.
B7. Percentage of the course grade devoted to these writing assignments:
   75%

B8. How do the assignments address the conventions of writing in your discipline?

Computability theory is the mathematical foundation of computer science. The assignments reflect writing in this unique area in which English (or any other natural language) is interwoven with several formal languages such as computer languages of algorithms, and symbolic languages of logic and mathematics. The assignments emphasize the intuitive correctness of the ideas in the first drafts, and precision, exactness and clarity in the later drafts. The final versions should present accurate, complete and polished arguments.

B9. Have you taken or are you planning to take a WID workshop (see FAQ 16)? When?

   Yes, in fall 2005.

B10. Instructional librarians can provide research instruction to students in WID courses (see FAQ 18). Do you anticipate needing this service for your WID course?

   No.