**THE UNIVERSITY OF THE WEST INDIES** 

**PROPOSAL FOR REVISED UNDERGRADUATE COURSE**

**Campus and Faculty: St. Augustine Campus, Faculty of Science and Technology School, Department, or Centre: Department of Computing and Information Technology**

**Course Code and Title: COMP 1601 Computer Programming I**

**Semester and Level: Semester 1 Level 1**

**Pre-requisites: None**

**Co-requisites:**

**Anti-requisites: None**

**Course Type: Core**

**Credits: 3**

**Projected Enrolment: 100**

**Projected Start Date: January 2020**

**Mode of Delivery: Face-to-Face** ☑ **Blended** ❑ **Online** ❑

**1. Course Description**

This course uses an appropriate programming language as a tool to teach fundamental programming concepts. The main concepts covered are sequence, selection and repetition logic, character and string manipulation, functions, and a basic introduction to arrays and their applications.

**2. Rationale**

This course equips students to solve problems on computer based systems. It identifies what type of problems can be solved by such systems and which cannot. It guides students on methods of developing structured algorithms. The focus on this course is problem description and presentation using either flowcharting or pseudocode tools. The selected programming language is used as a vehicle to show the basics of programming algorithms.

**3. Course Aims**

**4. Course Learning Outcomes**

Upon the successful completion of this course, the student will be able to:

1. Discuss the importance of algorithms in the problem-solving process.

2. Discuss how a problem may be solved by multiple algorithms, each with different properties.

3. Create algorithms for solving simple problems.

4. Use a programming language to implement, test, and debug algorithms for solving simple problems.

5. Analyse and explain the behaviour of simple programs involving the fundamental programming constructs variables, expressions, assignments, I/O, control constructs, functions, parameter passing, and recursion.

6. Identify and describe uses of primitive data types.

7. Write programs that use primitive data types.

8. Modify and expand short programs that use standard conditional and iterative control structures and functions.

9. Design, implement, test, and debug a program that uses each of the following fundamental programming constructs: basic computation, simple I/O, standard conditional and iterative structures, the definition of functions, and parameter passing.

10. Write a program that uses file I/O to provide persistence across multiple executions.

11. Choose appropriate conditional and iteration constructs for a given programming task.

12. Describe the concept of recursion and give examples of its use.

13. Identify the base case and the general case of a recursively-defined problem.

**5. Course Content/Syllabus**

The following main topics are covered in this course:

1. The concept and properties of algorithms  
2. Informal comparison of algorithm efficiency (e.g., operation counts)  
3. The role of algorithms in the problem-solving process  
4. Basic syntax and semantics of a higher-level language   
5. Variables and primitive data types (e.g., numbers, characters, Booleans, strings)  
6. Expressions and assignments  
7. Conditional and iterative control structures  
8. Simple I/O including file I/O  
9. Functions and parameter passing  
10. One dimensional arrays  
11. The concept of recursion

**6. Teaching Methods**

|  |  |
| --- | --- |
| **Teaching Method** | **Description** |
| Interactive Lectures | Live lectures delivered twice weekly |
| Online Activities (myElearning) | Quizzes |
| Laboratory Work |  |
| Flipped Classroom | Readings from text book.Video lectueres by the author of the essential textbook |
| Inquiry-based Learning | Interactive tutorial sessions - students solve problems on the board |
| Problem-based Learning | Apply programming concepts to real world problems |
| Groupwork |  |

**7. Contact and credits hours:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Type** | **Duration**  **(Number of weeks)** | **Contact Hours**  **(Weekly)** | **Credit Hours**  **(Weekly)** |
| Lecture | 13 weeks | 2 hours | 2 |
| Lab Tutorials | 13 weeks | 2 hours | 1 |
| Total: | 13 weeks | 4 hours | 3 |

**8. Course Assessments Description**

3 Assingments, 3 online quizzes, 2 courswork exams, 1 final exam

**9. Course Assessment Type and Course Learning Outcome Matrix**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Assessment** | **Learning Outcomes** | | | | | | | | | | | | | **Weighting %** | **Assessment**  **Description** | **Duration** |
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 |
| Assingment 1 | X | X | X |  |  |  |  |  |  |  |  |  |  | 5 | Take home problems | 10 days |
| Assingment 2 |  |  | X | X | X |  | X | X |  |  | X |  |  | 6 | Take home problems | 10 days |
| Assingment 3 |  |  |  |  |  |  |  |  |  |  |  |  |  | 6 | Take home problems | 10 days |
| Quiz 1 | X | X | X | X | X |  |  |  |  |  |  |  |  | 1 | Online quiz, multiple choice | 30 min |
| Quiz 2 |  |  | X | X | X | X | X |  |  |  | X |  |  | 1 | Online quiz, multiple choice | 30 min |
| Quiz 3 |  |  |  |  | X | X |  |  | X |  | X |  |  | 1 | Online quiz, multiple choice | 30 min |
| Coursework Exam 1 |  |  | X | X | X |  | X | X |  |  | X |  |  | 10 | Problems & short answer questions | 1.5 hours |
| Coursework Exam 2 |  |  | X | X | X | X | X | X | X | X | X |  |  | 20 | Problems & short answer questions | 1.5 hours |
| Final Examinaiton | X | X | X | X | X | X | X | X | X | X | X |  |  | 50 | Problems & short answer questions | 2 hours |
| Total % |  | | | | | | | | | | | | | 100% |  |  |

**10. Readings/Learning Resources** *(Online and Print)*

*Required/Essential*

Noel Kalicharan. C Programming - A Beginner's Course. 2005. CreateSpace Independent Publishing Platform.

**11. Staffing Requirements:** 1 Lecturer, 1 Tutor, 1 Marker

**12. Projected additional Cost (if any) for Proposed Undergraduate Course: None**

**13. Collaboration/ Consultation**

Academic staff from the Department of Computing and Information Technology were consulted on the proposal and full support was given for the changes to the curriculum.

**14. All relevant BUS Policies are available at: http://uwi.edu/undergraduatestudies/ default.aspx**

Have you taken these policies into account in the design of this Course? **Yes No \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Academic Staff Member / Contact Person Responsible/Coordinator**

Name: Dr. Wayne Goodridge Telephone: 868-662-2002 ext 85381

Email: wayne.goodridge@sta.uwi.edu

**Campus/Faculty/Department**:

**Date of Recommendation by Faculty Board/APAD:**

**Signature: Dean/Director\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Signature: Department Head**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Checked and endorsed by:** ❑ Campus Bursary ❑ CETL ❑ Library ❑ Bookshop ❑ Faculties on other Campuses ❑ OOL ❑ Other \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ NB. Attach supporting documents as appendix e.g. CETL. Library, online checklist

**Course Calendar**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Week** | **Topics** | **Required**  **Readings**  **Learning**  **Resources** | **Learning Activities** | **Assessment** |  |
| **Topic** | **Date** |
| 1. | Security Goals  Passive/Active Threats | Selected  chapters  from  required  text,  Content  posted on  myElearning weekly | Face to  Face  Lectures, Lab  Tutorials |  |  |
| 2. | Shared key Cryptography, Caesar Cipher, Rail Fence, Encryption (DES) | Assignment 1  (Programming Ciphers) | Week 2 |
| 3. | Authentication, Hash  Functions, MACs |  |  |
| 4. | Biometric Authentication, Token Based Authentication, Single SignOn Systems and Key Distribution Centers |  |  |
| 5. | Public key Cryptography, RSA Algorithm |  |  |
| 6. | Digital Signatures, Message Digests, Certificates | Assignment 2  (Biometric Techniques) | Week 6 |
| 7. | Security in Internet Commerce |  |  |
| 8. | Revision | Coursework Exam  (Topics from Week 1-7) | Week 8 |
| 9. | Presentation of Biometric Techniques, VPNs |  |  |
| 10. | Firewalls and Packet Filters, Access Lists |  |  |
| 11. | Application Attacks: Input/ Output Validation, Exception Handling, Sign-On, Access Control, Library Linking, Configuration Management, Session Handling and Program Execution | Group Project  (Security Policies) | Week 11 |
| 12. | Risk Management – Assets & Vulnerabilities & Attacks, Presentation of Application Attacks |  |  |
| 13. | Course Review |  |  |  |  |