Please complete this quiz after reviewing the lecture materials. You may refer to the lecture materials (anything directly linked from the current lecture page) while taking the quiz but you may not talk to other students or search the web. Please use your middlebury.edu account.

1. Email *

2. Full name *

3. What is the late policy for homeworks? *
   - Late work will never be accepted
   - Late work will not be accepted, but you have one 24h extension during the term
   - Late work will not be accepted, but you have two 24h extensions during the term
   - 10% off for each day late

4. What is the number one rule about the Honor Code (in any of your courses)? *
   - You may never do a web search for any of your assignments
   - You may not discuss the assignments with other students
   - If you are uncertain about how the Honor Code applies, ask your professor

5. We will cover four _________ in this course *
   - programming languages
   - programming paradigms
   - formal models of computation

6. Which linguistic hypothesis was covered in this lecture? *
   - The Church-Turing thesis
   - Chomsky’s theory of universal grammar
   - The Sapir-Whorf hypothesis

7. What do you need to add to the end of the course page URL to access the reading materials? *
   - text
   - book
   - textbook
   - reading

---

This content is neither created nor endorsed by Google.
CS 313 F21 - Quiz 02 - Syntax and Semantics

Please complete this quiz after reviewing the lecture materials. You may refer to the lecture materials (anything directly linked from the current lecture page) while taking the quiz but you may not talk to other students or search the web. Please use your middlebury.edu account.

* Required

1. Email *

2. Full name *

3. What are Syntax and Semantics? *
   - Syntax is spelling, Semantics is grammar
   - Syntax is grammar, Semantics is spelling
   - Syntax is meaning, Semantics is grammar/spelling
   - Syntax is grammar/spelling, Semantics is meaning

4. Valid syntax for a programming language is usually specified using *
   - regular expressions
   - a context-free grammar
   - a clear description in English

5. Algorithmic checking of syntax / semantics *
   - Correct syntax of a program can be checked algorithmically
   - Correct semantics of a program can be checked algorithmically
   - Both of the first two statements are true
   - Neither of the first two statements are true

6. In the following BNF rule, what are the terminals? <stmt> ::= <variable> = <expr> ; *
   - <stmt>
   - <variable> <expr>
   - = ;

7. What is (1 - 2) * 3 in prefix? *
   - - 1 2 * 3
   - * 3 - 1 2
   - 1 2 - 3 *
   - * - 1 2 3

This content is neither created nor endorsed by Google.
CS 313 Quiz 3

Match 10 questions to 10 answers (one-to-one mapping)

_  First language aimed at liberal arts students
1 - Grace Hopper

_  First functional language, based on lambda calculus
2 - COBOL

_  Which language? Multiply A by B giving C
3 - C

_  Considered to be the first computer programmer
4 - ALGOL

_  Developed for US Department of Defense, 1975-1985
5 - FORTRAN

_  Used Backus-Naur form for publication of language description in 1960
6 - John von Neumann

_  Developed the first compiler in 1952; helped design COBOL
7 - LISP

_  Designed in 1972 for coding UNIX routines
8 - BASIC

_  Oldest successful high-level language, designed for efficiently translating mathematical formulas
9 - Ada

_  20th century mathematician, pioneer of the modern digital computer architecture
10 - Ada Byron Lovelace
1. Email *

2. What is the first and last word in a valid Pascal program? *
   
   Mark only one oval.
   - begin ...
   - begin ...
   - program ...
   - program ...

3. Which is NOT a valid comment in Pascal? *
   
   Mark only one oval.
   - /* comment */
   - (* comment *)
   - { comment }

4. Is Pascal case sensitive? *
   
   Mark only one oval.
   - yes: temp, Temp, TEMP are all considered different
   - no: capitalization of letters does not matter

5. What is the assignment operator in Pascal? *
   
   Mark only one oval.
   - =
   - :=
   - ::=  
   - ==

6. Which is NOT a basic data type in Pascal? *
   
   Mark only one oval.
   - integer
   - real
   - double
   - char
   - Boolean

7. Which is NOT a valid if statement in Pascal? *
   
   Mark only one oval.
   - if x < 3 then y := 1;
   - if (x < 3) then y := 1;
   - if x < 3 then y := 1 else y := 0;
   - if x < 3 then y := 1; else y := 0;

8. Which is NOT a valid loop in Pascal? *
   
   Mark only one oval.
   - repeat .... until ...
   - loop .... while ...
   - while .... do ...
   - for .... do ....

9. What is the difference between a procedure and a function? *
   
   Mark only one oval.
   - none (two names for the same thing)
   - a function has a return value, a procedure doesn't
   - only functions can be recursive

10. Which of the following is NOT a valid parameter list? *
    
    Mark only one oval.
    - procedure p(x : integer; y : real);
    - procedure p(x, y : integer);
    - procedure p(x : integer, var y : integer);
    - procedure p(x, y : var);

11. How do you return the value 7 from a function named f? *
    
    Mark only one oval.
    - return 7;
    - retweet 7;
    - f := 7;
1. Which is NOT a Pascal data type? *
Mark only one oval.
- record
- dictionary
- pointer
- subrange
- array
- enumeration

2. What is the role of the semicolon in Pascal? *
Mark only one oval.
- statement terminator (required after each statement)
- statement separator (used to separate multiple statements)

3. The following is a legal Pascal type declaration: type lr = (left, right); a = array[lr] of integer; *
Mark only one oval.
- True
- False

4. Suppose a procedure “doit” takes no parameters. How would we call this function? *
Mark only one oval.
- doit();
- doit;

5. Why is it generally a good idea to pass arrays by reference (precede the parameter with “var”)? *
Mark only one oval.
- it is faster, since no copy is made of the array
- it allows the called function to modify the (caller’s) array values
- both of the above

6. How can we increment an integer variable i? *
Mark only one oval.
- inc(i);
- i += 1;
- i++; 
- i := 1;

7. Pascal allows defining nested procedures / functions *
Mark only one oval.
- True
- False

8. Which symbol denotes a pointer type in Pascal? *
Mark only one oval.
- ^
- &
- *

9. Which symbol denotes “not equal” in Pascal? *
Mark only one oval.
- ≠
- ±
- ≠

10. Which symbol denotes “not equal” in Pascal? *
Mark only one oval.
- ≠
- ±
- ≠

11. What is the null pointer in Pascal? *
Mark only one oval.
- null
- nil
- 0
1. Email *

2. How do we declare p to be a pointer to an integer? *
   Mark only one oval.
   - var p : *integer;
   - var ^p : integer;
   - var p^ : integer;
   - this is not possible in standard Pascal
   * 1 point

3. How do we dereference a pointer p to print the value it points to? *
   Mark only one oval.
   - writeln(^p);
   - writeln(p^);
   - writeln(*p);
   - this is not possible in standard Pascal
   * 1 point

4. Address-of operator *
   How do we get a pointer p to hold the address of an integer variable x?  
   Mark only one oval.
   - p := x^;
   - p := &x;
   - p := @x;
   - this is not possible in standard Pascal
   * 1 point

5. In a program using a pointer p, the statement "p := nil;" can never create garbage (i.e. inaccessible memory cells) *
   Mark only one oval.
   - true
   - false
   * 1 point

6. In a program using a pointer p, the statement "new(p);" can never create garbage (i.e. inaccessible memory cells) *
   Mark only one oval.
   - true
   - false
   * 1 point

7. Unlike in Java (which has a garbage collector), in Pascal it is the responsibility of the programmer to dispose of dynamically allocated memory cells that are no longer used *
   Mark only one oval.
   - true
   - false

8. Given the definition "procedure q(var x : real); begin x:=0 end;" the call "q(t);" can change t’s value (where t is a variable) *
   Mark only one oval.
   - true
   - false
   * 1 point

9. A procedure can have some parameters that are passed by value, and others that are passed by reference *
   Mark only one oval.
   - true
   - false
   * 1 point

10. Suppose we implemented our own version of Pascal’s function "inc(x)" to increment an integer variable. How would we declare the parameter x? *
    Mark only one oval.
    - pass by reference (with "var")
    - pass by value (no "var")
    * 1 point

11. The "inc(x)" function (where x is an int) could also be implemented in Java *
    Mark only one oval.
    - true
    - false
    * 1 point

The respondent’s email (null) was recorded on submission of this form.  
* Required
CS 313 Parameter Passing Quiz

1. Consider the following procedure definition and statements (in pseudo-code):

   ```pseudo
   procedure proc1(p, q) {
     p := 1;
     q := 2;
   }
   i := 0;   a[0] := 0;   a[1] := 0;
   proc1(i, a[i]);
   ```

   For each of the following parameter-passing mechanisms, list the values of i, a[0], and a[1] after the above code has been executed.

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>i</td>
<td>a[0]</td>
<td>a[1]</td>
</tr>
<tr>
<td>-----</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>pass-by-value:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>pass-by-reference:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>pass-by-name / macro expansion:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

2. Consider the following procedure definition and statements (in pseudo-code):

   ```pseudo
   var c, d;      // global variables
   procedure proc2(p, q) {
     c := p * 2;
     q := c - 1;
   }
   c := 3;   d := 2;
   proc2(c, d);
   ```

   For each of the following parameter-passing mechanisms, list the values of c and d after the above code has been executed.

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>c</td>
<td>d</td>
</tr>
<tr>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>pass-by-value:</td>
<td></td>
</tr>
<tr>
<td>pass-by-reference:</td>
<td></td>
</tr>
<tr>
<td>pass-by-value-result:</td>
<td></td>
</tr>
</tbody>
</table>
1. Email *

2. Which is NOT a C data type? *
   - int
   - double
   - struct
   - subrange
   - pointer

3. What is the role of the semicolon in C? *
   - statement terminator (required after each statement)
   - statement separator (used to separate multiple statements)

4. Suppose a void function "doit" takes no parameters. How would we call this function in C? *
   - doit();
   - doit;

5. Pass by reference *
   - C does not have pass-by-reference parameters, but a similar effect can be achieved using pointers and the "address of" operator &
   - C has pass-by-reference parameters prefixed by '&'

6. "if ((x = y) == z)" is a valid statement in C *
   - True
   - False

7. How do sizeof(int *) and sizeof(char *) relate? *
   - sizeof(int *) is larger
   - sizeof(char *) is larger
   - they are equal

8. Pointer arithmetic: what is printed by the following C code? char *s = "quiz";
   printf("%c", *(s + 3)); *
   - i
   - z
   - 0 (the number zero)
   - nothing – this code won’t compile

9. C does not have Booleans, it uses integers instead *
   - True
   - False

10. What C function can be used to get user input (like Pascal’s readln)? *
    - input
    - scanf
    - fread

11. C has both "while (...) {...}" and "do {...} while (...)" loops *
    - True
    - False
CS 313 - Quiz 09 - C structs, stack frame, survey

Please complete this quiz after reviewing the lecture materials and videos. You may refer to the lecture materials (anything directly linked from the current lecture page) while taking the quiz but you may not talk to other students or search the web.

The respondent's email (null) was recorded on submission of this form.

* Required

1. Email *

2. Where are local variables and parameters stored? *
   - In global memory
   - In the stack
   - In the heap

3. Where does "malloc" allocate memory? *
   - In global memory
   - In the stack
   - In the heap

Matching
For each of the following 5 items, select the matching (equal / most similar) term from the 7 options given. Each option is used at most once.

4. activation record *
   - access link
   - control link
   - struct
   - stack frame
   - *(a+i)
   - (*a).i
   - (&a)+i

5. record *
   - access link
   - control link
   - struct
   - stack frame
   - *(a+i)
   - (*a).i
   - (&a)+i

6. old FP (frame pointer) *
   - access link
   - control link
   - struct
   - stack frame
   - *(a+i)
   - (*a).i
   - (&a)+i

7. a[i] *
   - access link
   - control link
   - struct
   - stack frame
   - *(a+i)
   - (*a).i
   - (&a)+i

Brief course survey
Please answer the questions below to give me some feedback about the course so far. (All answers will receive full points once I manually assign them.)

9. Overall *
   - How is the course going so far? (1 - terrible, 2 - bad, 3 - ok, 4 - good, 5 - great)
   - Mark only one oval.

   1 2 3 4 5
terrible bad ok good great

10. Pace *
    - How is the pace? (1 - much too slow, 2 - slow, 3 - good, 4 - fast, 5 - much too fast)
    - Mark only one oval.

    1 2 3 4 5
too slow slow good fast too fast
11. **Workload** *
   How is the workload? (1 - very low, 2 - low, 3 - good, 4 - high, 5 - very high)
   Mark only one oval.

<table>
<thead>
<tr>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
</table>
   low | | | | | high |

12. **Is there anything I should change about the course?**
   (optional) – Please let me know if there are aspects of the course that need improving.

---

This content is neither created nor endorsed by Google.

Google Forms
1. Email *

2. What analogy does Steve Jobs use to explain OO programming? *
   - doing Laundry in San Francisco
   - buying Ben and Jerry’s in Vermont
   - building a spaceship in Cupertino

3. Which is NOT an OO language? *
   - Simula
   - LISP
   - Smalltalk
   - C++
   - Java

4. What’s the difference between method and message? *
   - method is a function, message is a procedure
   - method is a procedure definition, message is a procedure call
   - none — they are synonyms

5. Inheritance - which is NOT true? *
   - If A is a subclass of B, then B is a superclass of A
   - a subclass can override (redefine) a method declared in a superclass
   - subclasses can add instance variables to those defined in a superclass
   - subclasses can remove instance variables defined in a superclass

6. What does the Smalltalk expression 5 - 2 * 2 evaluate to? *
   - 1
   - 5
   - 6
   - 9

The respondent’s email (null) was recorded on submission of this form.
* Required

Please complete this quiz after reviewing the lecture materials and videos. You may refer to the lecture materials (anything directly linked from the current lecture page) while taking the quiz but you may not talk to other students or search the web.

This content is neither created nor endorsed by Google.
CS 313 - Quiz 11 - Smalltalk

Please complete this quiz after reviewing the lecture materials and videos. You may refer to the lecture materials (anything directly linked from the current lecture page) while taking the quiz but you may not talk to other students or search the web.

The respondent’s email (null) was recorded on submission of this form. * Required

1. Email *

2. What does this code do: #(2 4 6 8) at: 2 put: 3 *
   Mark only one oval.
   - It changes the array to #(3 4 6 8)
   - It changes the array to #(2 3 6 8)
   - It changes the array to #(2 4 3 8)
   - It throws an error

3. Smalltalk has unary, binary, and keyword messages. What is their order of precedence (highest to lowest)? *
   Mark only one oval.
   - unary, binary, keyword
   - unary, keyword, binary
   - keyword, unary, binary
   - It throws an error

4. What is a block? *
   Mark only one oval.
   - a keyword message with one argument
   - the value returned from a method
   - a group of Smalltalk statements, to be evaluated later

5. How does one write an if-statement in Smalltalk? *
   Mark only one oval.
   - by sending block(s) to a Boolean object via a keyword message
   - by sending Boolean value(s) to a block
   - using if:elsif:

6. What does this code evaluate to? s := 0. (2 to: 4) do: [:x | s := s + x]. s *
   Mark only one oval.
   - 0
   - 4
   - 6
   - 9
   - nil
   - it throws an error

This content is neither created nor endorsed by Google.

Google Forms
1. Email *

2. Which of the following Smalltalk expressions yields the largest result? *
   - 7 + 9 sqrt max: 5
   - (7 + 9) sqrt max: 5
   - 7 + (9 sqrt max: 5)

3. Suppose b is a block with no parameters that prints 'Hello World' to the Transcript. How do we run the block? *
   - b doIt
   - b run
   - b value

4. Short-circuiting: Which of the following expressions properly ensures a valid array index > 0? *
   - i > 0 and: (a at: i) = 7
   - [i > 0] and: (a at: i) = 7
   - i > 0 and: [(a at: i) = 7]

5. Which variable names indicate the correct types for the do: method? *
   - aBlock do: aBoolean
   - aBlock do: anInteger
   - aCollection do: aBlock

6. How can we write the body of the "max: x" method? *
   - x > self ifTrue: [^ x] ifFalse: [^ self]
   - self > x ifTrue: [^ self] * x
   - either of the first two options will work
   - neither of the first two options will work
CS 313 - Quiz 13 - Smalltalk 3

Please complete this quiz after reviewing the lecture materials and videos. You may refer to the lecture materials (anything directly linked from the current lecture page) while taking the quiz but you may not talk to other students or search the web.

Implementing new classes
Here are some questions about our first example of defining a new class

1. Email *

2. What data structure did we implement? *
   Mark only one oval.
   ○ Stack
   ○ Queue
   ○ Vector
   ○ Linked List

3. What is the name of the instance variable? *
   Mark only one oval.
   ○ a
   ○ c
   ○ q
   ○ data
   ○ self

4. What is the type of the instance variable? *
   Mark only one oval.
   ○ Array
   ○ OrderedCollection
   ○ Boolean

5. What does " ^ super new init " do? *
   Mark only one oval.
   ○ #1: It returns an instance of the superclass
   ○ #2: It sends the "init" message to the superclass
   ○ #3: It sends the "new" message to the superclass
   ○ #4: It sends the "init" message to the new instance, then returns it
   ○ #2 followed by #1
   ○ #3 followed by #4

Survey question
(any answer will earn a point :)

6. What's your preference for "flipped classroom" (video classes with questions answered in class) vs. in-person classes? *
   Mark only one oval.
   ○ Prefer flipped classroom (like the first 3 weeks)
   ○ Prefer a mix, e.g. one in-person class per week (like we did this week)
   ○ Prefer all in-person classes
   ○ Other: ____________________________

* Required
CS 313 - Quiz 14 - Smalltalk 4

Please complete this quiz after reviewing the lecture materials and videos. You may refer to the lecture materials (anything directly linked from the current lecture page) while taking the quiz but you may not talk to other students or search the web.

The respondent's email (null) was recorded on submission of this form.

* Required

1. Email *

2. In our IntList implementation, what is the total number of objects representing a list containing 3 integers? *
   - Mark only one oval.
     - 1
     - 3
     - 4
     - 6

3. In the IntListElt class, what is "val"? *
   - Mark only one oval.
     - an instance variable
     - an instance method
     - both

4. What are "setters" and "getters"? *
   - Mark only one oval.
     - instance methods providing access to instance variables
     - class methods to access superclass and subclass
     - special types of blocks

5. In Smalltalk, all instance variables are private *
   - Mark only one oval.
     - True
     - False

6. In the "do:" method in the IntListElt class, what does the statement "aBlock value: val" accomplish? *
   - Mark only one oval.
     - It calls aBlock recursively on the rest of the list
     - It evaluates aBlock and stores the result in val
     - It evaluates aBlock with val as the parameter
     - It checks if aBlock == val

This content is neither created nor endorsed by Google.