Examination 1

CSC315 Programming Language Concepts

09 October 2015

Contents

1 Eric Andow 4

1.1 More questions for discussion and review. 8

2 Emily Andrulis 14

2.1 More questions for discussion and review. 18

3 Sam Caldwell 25

3.1 More questions for discussion and review. 29

4 Dot Carmichael 37

4.1 More questions for discussion and review. 40

5 Sam Chalkley 48

5.1 More questions for discussion and review. 52

6 Nicci Geiger 58

6.1 More questions for discussion and review. 62

7 Brian Hixson-Simeral 69

7.1 More questions for discussion and review. 72

8 Tuli Leota 80

8.1 More questions for discussion and review. 83

8.2 More questions for discussion and review. 93

9 Aeint Thet Ngon 99

9.1 More questions for discussion and review. 103

10 Huong Nguyen 110

10.1 More questions for discussion and review. 114

10.2 More questions for discussion and review. 121

11 Spencer Rudnick 123

11.1 More questions for discussion and review. 127

12 Kat Sayrs **135**
12.1 More questions for discussion and review. 139

13 Cameron Seebach **145**
13.1 More questions for discussion and review. 149

14 Joe Sterchele **155**
14.1 More questions for discussion and review. 158

15 Htut Khine Win **163**
15.1 More questions for discussion and review. 167

Chapter 1

Eric Andow

1. 1. Most programming languages require the use of brackets to enclose the index in a reference to an element of an array.

   (a) Identify a language the requires the use of parentheses to enclose the index in a reference to an element of an array.

   (b) Why did the designers of the language choose parentheses rather than brackets?

   (a) Ada is one example that uses parentheses to reference array elements.

   (b) In the case of Ada, the designers chose to use a uniform style for function calls and array references because, ultimately, both represent a mapping to a specific point in memory.

2. 2. What is the relationship between a lexeme and a token?

   A lexeme is an association that the compiler uses to understand your code. So, a number is understood as an integer literal and an equal sign is understood as an equivalence operator. The lexemes are then grouped into categories, called tokens. While a lexeme appears verbatim in your code, a token is a name that only the compiler uses, like 'equal_sign' or 'mult_op'.

3. 3.

   (a) What kind of symbols are found at the internal nodes of a parse tree?

(b) What kind of symbols are found at the leaves of a parse tree?

(a) The internal nodes of a parse tree are non-terminal symbols, which branch out into other symbols.

(b) At the leaves of the tree lie terminal symbols, each of which represents an atom of the original code.

4. 4. One of the most significant contributions from the developers of ALGOL 60 also limited the success of that language. What was that contribution?

ALGOL 60 heralded the first use of the BNF formalism. BNF stands for Backus-Naur form, one of the primary ways of describing the syntax of programming languages. Unfortunately, when it was introduced, BNF was seen as too complicated. The slow shift towards accepting BNF would come too late for ALGOL 60 to ever gain a foothold among users.

5. 5. What problem were the creators of Common LISP trying to solve?

As the Reagan era commenced, Americans were facing tougher problems than they knew how to deal with. Namely, they had too many dialects of Lisp. Like having too many dialects of any language, this caused widespread strife and factionalism among those who were left with no way to communicate. To bring their colleagues back to the table of brotherhood, a few brave developers created Common Lisp, a language which included features of several popular Lisp dialects. In this way, the developers gave the many dialects a unified way to communicate.

6. 6. What is an ambiguous context free grammar?

Grammar?? But seriously, an ambiguous grammar is a sort of context-free grammar (one for which a parse tree can be generated) which can produce more than one valid parse tree for some input.

7. 7. Contrast the complexity of algorithms that can parse strings that conform to the most general kinds of context free grammars and the complexity of the algorithms that can parse strings that conform to the grammars of programming languages?

The most general kinds of context-free grammar require parsers that are of complexity O(n cubed), because they often make mistakes and must rebuild parts of the parse tree. Any commercially viable algorithm (the kind used to parse programming languages for compilers) will only operate on a subset of the grammars, but it will be of complexity O(n).

8. 8. Java represents characters with Unicode. It is the first widely used programming language with this feature. What is the significance of this feature?

Most languages support ASCII, a character set which only includes the english alphabet, plus special characters. UCS-2 is a standard developed by the Unicode Consortium which supports most character and number systems from around the world. This means that programmers in Thailand don't have to write custom Java to interpret their input; it just works.

9. 9. How does the binary coded decimal type differ from the floating point type?

A binary coded decimal represents a set of decimal digits to store a decimal number precisely. In contrast, a floating-point number stores a less-precise value and an exponent. This means that floating-point numbers have a much wider range, and take up less memory, but they are less precise (1.0 - 1) might give you something like (0.000000001).

10. 10. Identify a user-defined ordinal type in the Java programming language.

In 2004, Java added enumeration types, which map a set of discrete values onto a subset of the integers. The user can match any set of labels invisibly to a set of distinct integers.

11. 11. Mathematicians and programmers might have different ideas about the precedence of Boolean operators. Explain.

    For some reason, C-based languages give the AND operator precedence over the OR operator. However, mathematicians will remind you over and over again that the two operators should have equal precedence. They invented Boolean algebra, for crying out loud!

12. 12. Programmers should use `===` rather than `==` to test the equality of the values of two expressions in JavaScript. Why?

    In JavaScript, == means test whether the two sides evaluate to the same thing. So, (5+1) == (9-3) returns true. However, this isn't very useful if we want to match the exact expression, for example we want to find a difference between '7' and 7. For those purposes, we use ===, which does not affect the operands before comparing them.

13. 13. Describe a hazard of allowing short-circuited evaluation of expressions and side effects in expressions at the same time.

    Short-circuited evaluations can evaluate only the left-hand side, so if there are side-effects on the right-hand side (as in 'latest ¿ bestSoFar —— index++ ¿ size), they might not happen in some cases. When the program decides on side effects at runtime, this can be really difficult to track.

14. 14. Briefly describe the three steps in the mark-sweep algorithm for garbage collection.

    Only run this when you are out of available memory. 1. Mark everything to look like garbage. 2. For each pointer, mark what it points to as not-garbage. 3. All cells that are still marked as garbage are reclaimed.

15. 15. What led Yukihiro Matsumoto to create the Ruby programming language?

    Matz was unhappy with the neat languages that everyone liked, like Perl and Python. While they allowed you to use objects if you wanted, Matz thought that this was too weak. He wanted a language that would scale without the inelegance of primitive types or functions. Therefore, he built Ruby in the model of Smalltalk: a purely object-oriented language, where even primitive operators were methods and could be re-written.

16. 16. What did Microsoft aim to achieve with its development of the `C#` language?

    Microsoft wanted all the cool cats to use their .Net framework, so that they could obtain the power of huge market share. To achieve their dreams, they forged `C#` a weapon meant to combine the power of Java and C++ into a new general purpose programming language. They intended this great power to attract the greedy race of men, who would swarm in countless legions under the banner of .Net.

1.1 More questions for discussion and review.

1. The design of which machine influenced the design of the control statements in FORTRAN?

2. How many different kinds of control statements must the designer of a programming language include in a language?

3. What is the one question that applies in the design of all statements that allow selection or iteration?

4. What is an advantage of requiring that the **then** and **else** clauses of an **if** statement be compound statements?

5. How does the **switch** statement in C# differ from the **switch** statement in Java?

6. Distinguish between 2 statements in Ruby that correspond to Java's **switch** statement.

7. Features of a programming language sometimes persist longer than a feature of computing hardware that inspired and supported that part of the language's design. Similarly, features of hardware sometimes persist longer than some parts of a language's design that were created to take advantage of that feature in hardware.

   Give examples.

8. Who most famously warned of the dangers of using the **goto** statement? What did Donald Knuth have to say about the use of the **goto** statement?

9. Describes Ada's **for** loop. Are there some kinds of iteration that might be easier in Ada than in Java? Easier in Java than in Ada?

10. What does it mean to say that the guarded commands of Ada are non-deterministic?

11. The header files in a C program contain function prototypes. What is a function prototype?

12. Every method in a Ruby program belongs to a class. A programmer can place a definition of a method inside the definition of a class or outside of the definition of any class that the programmer writes. To which class does the method belong in the second case?

13. Distinguish between positional and keyword parameters.

14. Ruby blocks are closures. What does that mean?

15. What is a pure function?

16. Some languages give programmers means to define both functions and procedures. Java doe not. Is that a serious limitation?

17. Declarations of formal parameters in an Ada procedure can include, in addition to the names and types of the parameters, reserved words that do not appear in declarations in Java programs. What is the purpose of those reserved words?

18. The C language imposes a constraint upon programmers who want to pass a multidimensional array to a function. What is the constraint? How did the design of the Java programming language eliminate that constraint for programmers who use that language?

19. An activation record contains a return address, a dynamic link, parameters, and local variables.

    (a) To what does the return address point?

    (b) To what does the dynamic link point?

20. The stack will contain multiple activation records for a single subprogram under what circumstances?

21. How (or why?) does the LIFO protocol apply to calls to and returns from subprograms?

22. Which important development in computer architecture has changed the way that the stack is used in some systems for facilitating calls to and returns from subprograms?

23. A dynamic chain contains a history of what?

24. Which two numbers are needed to compute the address of a local variable in a subprogram?

25. How does a Ruby module differ from a class?

26. Memory for variables can be allocated on the heap and on the stack. In which place or places is memory allocated for objects in C++? in Java?

27. What problems were solved by the addition of genericity to Java?

28. What is the purpose of the static chain?

29. What is a singleton?

30. What are the two parts of the definition of an abstract data type?

1. IBM 704 system

2. While it is possible to use only one (GOTO), the minimum number of control statements in a language which does not use GOTO is two. One for choosing between control flow paths, and one for logically controlled iterations.

3. Requiring compound statements helps increase the readability and writability for programmers when using nested selector statements, that otherwise can get very messy and complicated.

4. In Java, the switch statement does not allow case expressions anywhere except the top level in the body of the switch. C# allows the execution of more than one segment. In C#, the control expressions as well as case statements can be strings.

5. Case expressions are the Ruby equivalent of Java's **switch** statement. One is semantically similar to nested if statements with case - when - then. The other is with boolean expressions being evaluated one at a time from top to bottom. The value of this case expression is equivalent to the value of the **then** statement corresponding to the first *true* **when** statement.

   (a) The IBM 704 influenced the design of control statements which are still used today, and prompted the development of Fortran.

(b) Most of the popular languages of the past 50 years have been designed around the von Neumann architecture. These are called imperative languages. In a von Neumann computer, both data and programs are stored in the same memory. The CPU, which executes instructions, is separate from the memory. Therefore, instructions and data must be transmitted, or piped, from memory to the CPU. Results of operations in the CPU must be moved back to memory. Nearly all digital computers built since the 1940s have been based on the von Neumann architecture.

(c) The Algol standard used several different syntaxes which, among other things, allowed Europeans to use a comma to denote a decimal point, while Americans could continue to use a period.

6. The **register** keyword in C is a hint to the compiler that a variable will be used repeatedly, and so it should be stored in the CPU rather than in memory. However, modern compilers are far better at optimization than programmers, so this keyword is outdated and unnecessary.

7. Edsger Dijkstra noted The goto statement as it stands is just too primitive; it is too much an invitation to make a mess of ones program. Donald Knuth argued there were occasions when the efficiency of the goto outweighed its harm to readability.

8. The for loop looks like this

   ```
   for variable in [reverse] discrete_range loop    end loop;
   ```

   Adas for loop can use any ordinal type variable for its counter. Arrays with ordinal type subscripts can be conveniently processed. (261) For loops in Java are more flexible - can have infinite loops, change loop variable inside body, etc. but is potentially more confusing to read.

9. Guarded commands in Ada are nondeterministically chosen for execution when more than one of the statements are evaluated to true. This means that if there are three guarded statements and two of the three evaluate to true, then each time the program will use one of the two statements. It will not always use the one that appears first nor the one that appears last, but rather it will choose between them non-deterministically or randomly.

10. A function prototype is a function declaration that gives the function's name and type signature, but does not specify the function body. It is also referred to as a function interface at times. In other languages these are uncommon because subprograms do not need declarations since they do no need to be defined before they are called.

11. If a method is defined outside of the definition of any class that the programmer writes then the method belongs to the root object, **Object**.

12. Positional parameters are bound based on the order in which they are given to the function. Keyword parameters are when the name of the formal parameter to which an actual parameter is to be bound is specified with the actual parameter in a call. This allows them to be input in any order.

13. A closure is an anonymous function which can be passed as a parameter. Also, Ruby doesn't nest scope, so variables defined in a method are not accessible outside of the method (say, in the class which defines them).

14. A **pure function** is a **function** where the return value is only determined by its input values, without observable side effects.

15. It does not seem to be a serious limitation, it just means that performing procedures in Java is more verbose because it requires that the programmer instantiate the class with any needed methods.

16. Ada allows the programmer to specify *in* mode, *out* mode, and *inout* mode for each formal parameter. This means that they can receive data from the actual parameter, they can send data to the actual parameter, or they can do both.

17. In C, programmers are required to pass the length of an array to a method. Specifying the length of an array (like in C) is unnecessary because Java arrays store a pointer to an address containing their length.

18. The return address usually consists of a pointer to the instruction following the call in the code segment of the calling program unit. The dynamic link points to the base of the activation record instance of the caller.

19. In a recursive program there can be multiple activation records (although they will be incomplete) for a single subprogram.

20. LIFO allows for subprograms to be nested within one another and for subprograms to be used as parameters within other subprograms. (417)

21. RISC (reduced instruction set computing) machines have parameters passed in registers in their compilers because RISC machines have more registers than CISC (complex instruction set computing) machines. Chapter 10 assumes parameters are passed in the stack though, as they had been in CISC machines.

22. A dynamic chain represents the history of how the execution got to its current position, which is always in the subprogram code whose activation record instance is on top of the stack.

23. To compute the address of a local variable you need the (chain_offset, local_offset) pair.

24. Modules are unlike classes in that they cannot be instantiated or subclassed and do not define variables. Methods that are defined in a module include the modules name in their names.

25. In C++, variables can be allocated to the heap either by making them static or by allocating memory with the keyword new. However, variables which are initialized during the execution of a function are allocated to the stack. Java behaves the same way.

26. Generics allow a type or method to operate on objects of various types while providing compile-time type safety.

27. The static chain is a path of pointers which go from each function to its parent. They allow child subprograms to use variables which are local to their parent, grandparent, or farther up, without needing to copy those variables to the call stack.

28. A singleton is a class which provides a global access point to a single instance. This is useful for tasks which only need one point of access, like a file system. The two pieces are:

    (a) A type definition which allows program units to declare variables of the type but hides the representation of objects of the type.

    (b) A set of operations for manipulating objects of the type.

Chapter 2

Emily Andrulis

1. Most programming languages require the use of brackets to enclose the index in a reference to an element of an array.

   (a) Identify a language the requires the use of parentheses to enclose the index in a reference to an element of an array.

   (b) Why did the designers of the language choose parentheses rather than brackets?

   (a) Ada uses parentheses to enclose the index in a reference to an element of an array.

   (b) Ada was designed this way so that array references and function calls in expressions would both use the same syntax. They wanted this uniformity because they are both mappings, and should therefore have the same syntax.

2. What is the relationship between a lexeme and a token?

   A token is a category of its lexemes. An identifier is a token that can have many lexemes (or instances), such as index or count, but some tokens only have a single possible lexeme, i.e. = is the only lexeme for equalSign token.

3. (a) What kind of symbols are found at the internal nodes of a parse tree?

(b) What kind of symbols are found at the leaves of a parse tree?

(a) Internal nodes of a parse tree have a nonterminal symbol, such as id in angle brackets.

(b) Leaves of a parse tree have terminal symbols, such as A.

4. One of the most significant contributions from the developers of ALGOL 60 also limited the success of that language. What was that contribution?

ALGOL 60 was the first programming language to use BNF, or even to

formally describe the syntax of the langauge at all. Even though nowadays BNF is widely used and recognized as an acceptable way to document programming language syntax, at the time BNF was not easily accepted since most viewed it as strange and complicated.

5. What problem were the creators of Common LISP trying to solve?

Mainly, the creaters of LISP were trying to create a language that could do list processing. In particular, they needed a language that could support recursion, conditional expressions, and dynamic storage allocation or implicit deallocation. The creaters were John McCarthy and Marvin Minsky from MIT.

6. What is an ambiguous context free grammar?

An ambiguous context free grammar is one where there are two or more distinct parse trees possible for a single statement. This commonly occurs when a grammar allows parse trees to grow both from the left and the right, instead of restricting it to one side.

7. Contrast the complexity of algorithms that can parse strings that conform to the most general kinds of context free grammars and the complexity

of the algorithms that can parse strings that conform to the grammars of programming languages?

Attribute grammars are used to describe more of the structure of a programming language than can be described with a context-free grammar. Rules of static semantics are described in attribute grammars, whereas context free grammars do not take these into consideration. Therefore, algorithms that parses strings for the grammars of programming languages are more complicated because they must also account for static and dynamic semantics. However, parsing algorithms for context-free grammars must account for a large general set of grammars, and their complexity is usually measured as $O(n^3)$. Even though programming languages are complicated, they can use parsing alogirthms that are less general and do not fit all grammars, so long as they do fit the grammar for that language. Therefore, commercial compliers have complexity of O(n) typically, which makes them less complex algorithms than those of the general context-free grammar parsing algorithms.

8. Java represents characters with Unicode. It is the first widely used programming language with this feature. What is the significance of this feature?

This feature is significant because as opposed to its predeccessor, ASCII, the Unicode character set includes characters from most of the world's natural languages. Therefore, Unicode is needed to faciliate international communication with code. Also, Unicode encapsulates ASCII, and the first 128 characters of Unicode are the same as those from ASCII.

9. How does the binary coded decimal type differ from the floating point type?

Decimal types are able to precisely store decimal numbers, within a restricted range, which cannot be done with floating point types. Although this allows more accuracy in arithmatic operations, it is also mildy wasteful with memory since they are stored one or two digits per byte.

10. Identify a user-defined ordinal type in the Java programming language.

Enum is the class of user-defined ordinal types in Java.

11. Mathematicians and programmers might have different ideas about the precedence of Boolean operators. Explain.

In mathematics the OR and AND operators must have equal precedence, but in programming languages, specifically those that are C-based, a higher precedence is assigned to the AND operator over OR.

12. Programmers should use `===` rather than `==` to test the equality of the values of two expressions in JavaScript. Why?

A double equals allows coercion first, whereas the triple equals does not.

13. Describe a hazard of allowing short-circuited evaluation of expressions and side effects in expressions at the same time.

If a language allows short-circuited evaluation and side effects of an expression at the same time, then it is possible for a piece of the expression to not be evaluated, and therefore not have it's side effect come into play. This can be a serious problem if the evaluation of the side effect was neccessary for the program's correctness.

14. Briefly describe the three steps in the mark-sweep algorithm for garbage collection.

Mark-sweep starts with all cells in the heap having their indicators set to indicate that they are garbage. Next, each point in the program is traced into the heap, and all reachable cells are marked instead as not being garbage. Finally, all cells in the heap that were not marked as still being in use are returned to the list of available space.

15. What led Yukihiro Matsumoto to create the Ruby programming language?

He was unsatisfied with the Perl and Python languages, and wanted another language that was purely object oriented in that it did not support non-object primitive types nor functions instead of method calls.

16. What did Microsoft aim to achieve with its development of the C# language?

Microsoft wanted to create a language for component-based software development, specifically for such development in the .NET framework that they had already established.

2.1 More questions for discussion and review.

1. The design of which machine influenced the design of the control statements in FORTRAN?

Fortran was designed by the architects of the IBM 704, and the control statements were directly related to machine language instructions, instead of language design requirements.

2. How many different kinds of control statements must the designer of a programming language include in a language?

A programming language can be designed with as few as two control statements. There must be one to choose between two control flow paths and one to handle logically controlled iterations.

3. What is the one question that applies in the design of all statements that allow selection or iteration?

The one question is this: Should the control structure have multiple entries? This asks whether or not the execution of the code segments must always begin with the first statement in the segment. Most believe that multiple entries decrease readability more than they add flexibility and functionality, and they are only possible in languages that include goto and statement labels. Java, for example, does not include these.

4. What is an advantage of requiring that the **then** and **else** clauses of an **if** statement be compound statements?

Requiring compound statements helps increase the readability and writability for programmers when using nested selector statements, that otherwise can get very messy and complicated.

5. How does the **switch** statement in C# differ from the **switch** statement in Java?

C# switch statements do not allow the implicit execution of more than one segment, and they do this by requiring each case statement to end in a break or goto. Also, in C# the control expression and the case statements can be strings in C#.

6. Distinguish between 2 statements in Ruby that correspond to Java's **switch** statement.

Ruby has both a case expression form that is similar to the switch in Java, and a case expression that is semantically similar to a list of neted if statements. It's form is as follows:

case

when BooleanExpression **then** expression

when BooleanExpression **then** expression

when BooleanExpression **then** expression

[**else** expression]

end

7. Features of a programming language sometimes persist longer than a feature of computing hardware that inspired and supported that part of the language's design. Similarly, features of hardware sometimes persist longer than some parts of a language's design that were created to take advantage of that feature in hardware.

Give examples.

An example of this would be how the IBM 704 influenced the design of control statements.

8. Who most famously warned of the dangers of using the **goto** statement? What did Donald Knuth have to say about the use of the **goto** statement?

   Edsger Dijkstra famously warned about the dangers of including goto statements in programming languages in an expose he wrote in 1968. Donald Knuth argued for keeping the use of goto available, stating that there were certain times when the efficiency of using a goto was more significant that the harm it caused to the readability of the program.

9. Describes Ada's **for** loop. Are there some kinds of iteration that might be easier in Ada than in Java? Easier in Java than in Ada?

   Ada's for loop limits the scope of the loop variable to the loop body. After the loop body, loop variables are not defined and their values are not relevant. Even if a variable with the same name is defined before the loop, a loop variable of the same name can be created without changing the other variable at all. It might be easier to iterate through ranges of numbers in Ada because reusing variables won't mess them up later outside the loop. However, Java would be better if you want to iterate through something and keep the loop variable to use later on.

10. What does it mean to say that the guarded commands of Ada are nondeterministic?

    Guarded commands in Ada are nondeterministically chosen for execution when more than one of the statements are evaluated to true. This means that if there are three guarded statements and two of the three evaluate to true, then each time the program will use one of the two statements. It will not always use the one that appears first nor the one that appears last, but rather it will choose between them nondeterministically or randomly at times.

11. The header files in a C program contain function prototypes. What is a function prototype?

A function prototype is a function declaration taht gives the function's name and type signature, but does not specify the function body. It is also referred to as a function interface at times. In other languages theses are uncommon because subprograms do not need declarations since they do no need to be defined before they are called.

12. Every method in a Ruby program belongs to a class. A programmer can place a definition of a method inside the definition of a class or outside of the definition of any class that the programmer writes. To which class does the method belong in the second case?

If a method is defined outside of the definition of any class that the programmer writes then the method belongs to the root object, **Object**.

13. Distinguish between positional and keyword parameters.

Positional parameters are bound based on the order in which they are given to the function. Keyword parameters are when the name of the formal parameter to twhich an actual parameter is to be bound is specified with the actual parameter in a call. This allows them to be input in any order

14. Ruby blocks are closures. What does that mean?

A closure is a nested subprogram and its referencing environment, which together allow the subprogram to be called from anywhere in the program. This means that Ruby blocks can be called from anywhere in the program.

15. What is a pure function?

A pure function is one that is well based on the mathematical model in that it produces no side effects and it modifies neither the parameters not any variables defined outside the function.

16. Some languages give programmers means to define both functions and procedures. Java does not. Is that a serious limitation?

Java does not allow procedures, but like most programming languages that do not, it also allows functions to return values of void, which in effect makes it behave very much like a procedure anyways. Therefore, it doesn't really limit the language to exclude procedures.

17. Declarations of formal parameters in an Ada procedure can include, in addition to the names and types of the parameters, reserved words that do not appear in declarations in Java programs. What is the purpose of those reserved words?

    Ada allows the programmer to specify in mode, out mode, or inout mode on each parameter. These specify whether or not the parameter can receive data from the corresponding actual parameter (in), transmit data to the actual parameter (out), or do both (inout).

18. The C language imposes a constraint upon programmers who want to pass a multidimensional array to a function. What is the constraint? How did the design of the Java programming language eliminate that constraint for programmers who use that language?

    In C, the compiler must be able to build the mapping function for the array while seeing only the text of the subprogram. This means that the mapping function at least needs the number of columns, but not neccessarily the number of rows. This means though that it cannot input matrices with different numbers of columns. In Java, arrays are objects that inherit the length constant. This allows the formal parameter for an array to be passed with two sets of empty brackets. Since the arrays have their own unique length values, the matrix can have rows of different lengths.

19. An activation record contains a return address, a dynamic link, parameters, and local variables.

    (a) To what does the return address point?

    (b) To what does the dynamic link point?

    The return address points to the instruction following the call in the code segment of the calling program unit. The dynamic link points to the base of the activiation record instance of the caller.

20. The stack will contain multiple activation records for a single subprogram under what circumstances?

With recursion there is the possibility of multiple simultaneous activations of a subprogram, so there is more than one instance of that subprogram at one time.

21. How (or why?) does the LIFO protocol apply to calls to and returns from subprograms?

Last in first out protocol applies to calls to and returns from subprograms because calls to subprograms get put on the stack and returns from subprograms use the last call in the stack as the first thing returned.

22. Which important development in computer architecture has changed the way that the stack is used in some systems for facilitating calls to and returns from subprograms?

RISC (reduced instruction set computing) machines have parameters passed in registers in their compilers because RISC machines have more registers than CISC (complex instruction set computing) machines. Chapter 10 assumes parameters are passed in the stack though, as they had been in CISC machines.

23. A dynamic chain contains a history of what?

A dynamic chain contains the history of all subprogram activation records, but in the reverse order of when they were activated. (LIFO stack)

24. Which two numbers are needed to compute the address of a local variable in a subprogram?

To compute the address of a local variable one would need both the chain offset and the local offset as a pair. (Found on page 457.)

25. How does a Ruby module differ from a class?

Modules are different from classes in that they cannot be instantiated or subclassed and they do not define variables. Also, methods defined in a module will always have the module's name in their names, i.e. MyModule.myMethod() would be a method in module MyModule.

26. Memory for variables can be allocated on the heap and on the stack. In which place or places is memory allocated for objects in C++? in Java?

In C++ memory for variables is allocated on the heap by making them "static" or by allocating memory with the keyword "new". Variables that are initialized during the execution of a function though are allocated to the stack instead. Java is another C-based language and therefore works the same way.

27. What problems were solved by the addition of genericity to Java?

By adding genericity to Java this allows a type or method to operate on objects of various types while still providing compile-time type safety.

28. What is the purpose of the static chain?

A static chain is a path of pointer which go from each function to its parent. They allow child subprograms to use variables which are local to their parent or other ancestors without needing to copy those variables to the call stack.

29. What is a singleton?

A singleton is a class which provides a single global access point to a single instance. This is helpful when there are tasks that need only one point of access, i.e. a file system.

30. What are the two parts of the definition of an abstract data type?

An abstract data type is both a set of operations for manipulating objects of that type, and also a definition which allows program units to declare variables of that type while hiding the representation of objects of that type.

Chapter 3

Sam Caldwell

1. Most programming languages require the use of brackets to enclose the index in a reference to an element of an array.

   (a) Identify a language the requires the use of parentheses to enclose the index in a reference to an element of an array.

   (b) Why did the designers of the language choose parentheses rather than brackets?

   (a) Ada

   (b) The designers of Ada specifically chose parentheses to enclose subscripts so there would be uniformity between array references and function calls in expressions, in spite of potential readability problems

2. What is the relationship between a lexeme and a token?

   A token represents functional groups of lexemes, which are representations of the lowest level of syntactic units

3. (a) What kind of symbols are found at the internal nodes of a parse tree?

   (b) What kind of symbols are found at the leaves of a parse tree?

(a) internal nodes have non-terminal symbols

(b) leaf nodes have terminal symbols

4. One of the most significant contributions from the developers of ALGOL 60 also limited the success of that language. What was that contribution?

   ”Ironically, one of the most important contributions to computer science associated with ALGOL 60, BNF, was also a factor in its lack of acceptance. Although BNF is now considered a simple and elegant means of syntax descrip- tion, in 1960 it seemed strange and complicated.” pg. 57 (10th edition)

5. What problem were the creators of Common LISP trying to solve?

   The developers of Common LISP were trying to solve issues with portability among programs written in various dialects.

6. What is an ambiguous context free grammar?

   ”A grammar that generates a sentential form for which there are two or more distinct parse trees is said to be ambiguous”

7. Contrast the complexity of algorithms that can parse strings that conform to the most general kinds of context free grammars and the complexity of the algorithms that can parse strings that conform to the grammars of programming languages?

   Parsing algorithms for unambiguous grammars are complicated and inneficient. In fact, ” the complexity of such algorithms is $O(n^3)$”(pg. 180). On the other hand, the algorithms used for context free grammars are closer to the level of $O(n)$, which means the time they take is linearly related to the length of the string to be parsed. This is vastly more efficient than $O(n^3)$ algorithms.

8. Java represents characters with Unicode. It is the first widely used programming language with this feature. What is the significance of this feature?

   The previously used ASCII was becoming obsolete with the globalization of business and the need for computers to communicate around the world. Java quickly becomes a global coding language based on it’s acceptance of unicode - which includes (among other things) the Cryllic alphabet.

9. How does the binary coded decimal type differ from the floating point type?

   ”Decimal types have the advantage of being able to precisely store decimal values, at least those within a restricted range, which cannot be done with floating-point”

10. Identify a user-defined ordinal type in the Java programming language.

    ”There are two user-defined ordinal types that have been supported by programming languages: enumeration and subrange.”

11. Mathematicians and programmers might have different ideas about the precedence of Boolean operators. Explain.

    Because in programming, arithmetic expressions can be the operands of relational expressions, and relational expressions can be the operands of Boolean expressions. For example the ’=’ sign - in mathmatics is signifies that two sides of an equation are the same value, wheras in programming it signifies the changing of value of a variable (or other things).

12. Programmers should use `===` rather than `==` to test the equality of the values of two expressions in JavaScript. Why?

Because the '==' operator uses coercions to achieve equality, '===' is testing for it.

13. Describe a hazard of allowing short-circuited evaluation of expressions and side effects in expressions at the same time.

"Suppose that short-circuit evaluation is used on an expression and part of the expres- sion that contains a side effect is not evaluated; then the side effect will occur only in complete evaluations of the whole expression."

14. Briefly describe the three steps in the mark-sweep algorithm for garbage collection.

First, all cells in the heap have their indicators set to indicate they are garbage. Second, Every pointer in the program is traced into the heap, and all reachable cells are marked as not being garbage Third, All cells in the heap that have not been specifically marked as still being used are returned to the list of available space

15. What led Yukihiro Matsumoto to create the Ruby programming language?

"The motivation for Ruby was dissatisfaction of its designer with Perl and Python. Although both Perl and Python support object-oriented programming,14 nei- ther is a pure object-oriented language, at least in the sense that each has primi- tive (nonobject) types and each supports functions."(pg. 100)

16. What did Microsoft aim to achieve with its development of the `C#` language?

"The purpose of `C#` is to provide a language for component-based software development, specifically for such development in the .NET Framework.

In this environment, components from a variety of languages can be easily com- bined to form systems"

3.1 More questions for discussion and review.

1. The design of which machine influenced the design of the control statements in FORTRAN?

   IBM 704

2. How many different kinds of control statements must the designer of a programming language include in a language?

   It's possible to use one if you are using the (GOTO) statement, but otherwise the minimum is two control statements.

3. What is the one question that applies in the design of all statements that allow selection or iteration?

   Whether or not the control structure should have multiple entries.

4. What is an advantage of requiring that the **then** and **else** clauses of an **if** statement be compound statements?

   Compound statements increase readability and writability when using nested loops and selector statements. Otherwise it would get pretty ugly pretty fast...

5. How does the **switch** statement in C# differ from the **switch** statement in Java?

   C# allows execution of more than one segment, also the control expressions can be strings as well as case statements. In Java, the switch statement does not allow case expressions anywhere except the top level of the switch.

6. Distinguish between 2 statements in Ruby that correspond to Java's **switch** statement.

Case expressions are the Ruby equivalent of Java's **switch** statement. One is semantically similar to nested if statements with case - when - then. The other is with boolean expressions being evaluated one at a time from top to bottom. The value of this case expression is equivalent to the value of the **then** statement corresponding to the first *true* **when** statement.

7. Features of a programming language sometimes persist longer than a feature of computing hardware that inspired and supported that part of the language's design. Similarly, features of hardware sometimes persist longer than some parts of a language's design that were created to take advantage of that feature in hardware.

(a) The IBM 704 influenced the design of control statements which are still used today, and prompted the development of Fortran.

(b) The best example of a feature that has persisted is the Von Neumann Architecture. In a von Neumann computer, both data and programs are stored in the same memory. The CPU, which executes instructions, is separate from the memory. Therefore, instructions and data must be transmitted, or piped, from memory to the CPU. Results of operations in the CPU must be moved back to memory. Nearly all digital computers built since the 1940s have been based on the von Neumann architecture.

(c) The Algol standard used several different syntaxes which, among other things, allowed Europeans to use a comma to denote a decimal point, while Americans could continue to use a period.

(d) The **register** keyword in C is a hint to the compiler that a variable will be used repeatedly, and so it should be stored in the CPU rather than in memory. However, modern compilers are far better at optimization than programmers, so this keyword is outdated and unnecessary.

8. Who most famously warned of the dangers of using the **goto** statement? What did Donald Knuth have to say about the use of the **goto** statement?

Edsger Dijkstra noted The goto statement as it stands is just too primitive; it is too much an invitation to make a mess of ones program. Donald Knuth argued there were occasions when the efficiency of the goto outweighed its harm to readability.

9. Describes Ada's **for** loop. Are there some kinds of iteration that might be easier in Ada than in Java? Easier in Java than in Ada?

The for loop looks like this

```
for variable in [reverse] discrete_range loop    end loop;
```

Adas for loop can use any ordinal type variable for its counter. Arrays with ordinal type subscripts can be conveniently processed. (261) For loops in Java are more flexible - can have infinite loops, change loop variable inside body, etc. but is potentially more confusing to read.

10. What does it mean to say that the guarded commands of Ada are nondeterministic?

Guarded commands in Ada are nondeterministically chosen for execution when more than one of the statements are evaluated to true. This means that if there are three guarded statements and two of the three evaluate to true, then each time the program will use one of the two statements. It will not always use the one that appears first nor the one that appears last, but rather it will choose between them non-deterministically or randomly.

11. The header files in a C program contain function prototypes. What is a function prototype?

A function prototype is a function declaration that gives the function's name and type signature, but does not specify the function body. It is also referred to as a function interface at times. In other languages these are uncommon because subprograms do not need declarations since they do no need to be defined before they are called.

12. Every method in a Ruby program belongs to a class. A programmer can place a definition of a method inside the definition of a class or outside of the definition of any class that the programmer writes. To which class does the method belong in the second case?

    If a method is defined outside of the definition of any class that the programmer writes then the method belongs to the root object, **Object**.

13. Distinguish between positional and keyword parameters.

    Positional parameters are bound based on the order in which they are given to the function. Keyword parameters are when the name of the formal parameter to which an actual parameter is to be bound is specified with the actual parameter in a call. This allows them to be input in any order.

14. Ruby blocks are closures. What does that mean?

    A closure is an anonymous function which can be passed as a parameter. Also, Ruby doesn't nest scope, so variables defined in a method are not accessible outside of the method (say, in the class which defines them).

15. What is a pure function?

    A **pure function** is a **function** where the return value is only determined by its input values, without observable side effects.

16. Some languages give programmers means to define both functions and procedures. Java does not. Is that a serious limitation?

It does not seem to be a serious limitation, it just means that performing procedures in Java is more verbose because it requires that the programmer instantiate the class with any needed methods.

17. Declarations of formal parameters in an Ada procedure can include, in addition to the names and types of the parameters, reserved words that do not appear in declarations in Java programs. What is the purpose of those reserved words?

Ada allows the programmer to specify *in* mode, *out* mode, and *inout* mode for each formal parameter. This means that they can receive data from the actual parameter, they can send data to the actual parameter, or they can do both.

18. The C language imposes a constraint upon programmers who want to pass a multidimensional array to a function. What is the constraint? How did the design of the Java programming language eliminate that constraint for programmers who use that language?

In C, programmers are required to pass the length of an array to a method. Specifying the length of an array (like in C) is unnecessary because Java arrays store a pointer to an address containing their length.

19. An activation record contains a return address, a dynamic link, parameters, and local variables.

    (a) To what does the return address point?

    (b) To what does the dynamic link point?

The return address usually consists of a pointer to the instruction following the call in the code segment of the calling program unit. The dynamic link points to the base of the activation record instance of the caller.

20. The stack will contain multiple activation records for a single subprogram under what circumstances?

    In a recursive program there can be multiple activation records (although they will be incomplete) for a single subprogram.

21. How (or why?) does the LIFO protocol apply to calls to and returns from subprograms?

    LIFO allows for subprograms to be nested within one another and for subprograms to be used as parameters within other subprograms.

22. Which important development in computer architecture has changed the way that the stack is used in some systems for facilitating calls to and returns from subprograms?

    RISC (reduced instruction set computing) machines have parameters passed in registers in their compilers because RISC machines have more registers than CISC (complex instruction set computing) machines. Chapter 10 assumes parameters are passed in the stack though, as they had been in CISC machines.

23. A dynamic chain contains a history of what?

    A dynamic chain represents the dynamic history of how the execution got to its current position, which is always in the subprogram code whose activation record instance is on top of the stack.

24. Which two numbers are needed to compute the address of a local variable in a subprogram?

    To compute the address of a local variable you need the (chain_offset, local_offset) pair.

25. How does a Ruby module differ from a class?

Modules are unlike classes in that they cannot be instantiated or subclassed and do not define variables. Methods that are defined in a module include the modules name in their names.

26. Memory for variables can be allocated on the heap and on the stack. In which place or places is memory allocated for objects in C++? in Java?

In C++, variables can be allocated to the heap either by making them static or by allocating memory with the keyword new. However, variables which are initialized during the execution of a function are allocated to the stack. Java behaves the same way.

27. What problems were solved by the addition of genericity to Java?

Generics allow a type or method to operate on objects of various types while providing compile-time type safety.

28. What is the purpose of the static chain?

The static chain is a path of pointers which go from each function to its parent. They allow child subprograms to use variables which are local to their parent, grandparent, or farther up, without needing to copy those variables to the call stack.

29. What is a singleton?

A singleton is a class which provides a global access point to a single instance. This is useful for tasks which only need one point of access, like a file system.

30. What are the two parts of the definition of an abstract data type?

A type definition which allows program units to declare variables of the type but hides the representation of objects of the type. 2. A set of operations for manipulating objects of the type.

Chapter 4

Dot Carmichael

1. Most programming languages require the use of brackets to enclose the index in a reference to an element of an array.

   (a) Identify a language the requires the use of parentheses to enclose the index in a reference to an element of an array.

   (b) Why did the designers of the language choose parentheses rather than brackets?

   (a) Ada uses parentheses to enclose the index.

   (b) The programmers used parentheses instead of brackets to provide uniformity between arrays and functions, since they both use mapping.

2. What is the relationship between a lexeme and a token?

   A token is a category of one or more lexemes within a language.

3. (a) What kind of symbols are found at the internal nodes of a parse tree?

   (b) What kind of symbols are found at the leaves of a parse tree?

   (a) Non-terminal symbols.

(b) Terminal symbols.

4. One of the most significant contributions from the developers of ALGOL 60 also limited the success of that language. What was that contribution?

Backus-Naur Form syntax description

5. What problem were the creators of Common LISP trying to solve?

The problem was a lack of portability between programs written in different dialects of LISP.

6. What is an ambiguous context free grammar?

An ambiguous context free grammar can generate sentences with more than one leftmost and rightmost derivation.

7. Contrast the complexity of algorithms that can parse strings that conform to the most general kinds of context free grammars and the complexity of the algorithms that can parse strings that conform to the grammars of programming languages?

Algorithms for parsing any unambiguous grammars are highly inefficient, with a complexity of O(n**3), whereas algorithms for programming languages are complexity O(n).

8. Java represents characters with Unicode. It is the first widely used programming language with this feature. What is the significance of this feature?

Unicode allows programs to be written which can communicate with other countries whose languages cannot be represented by the basic english alphabet.

9. How does the binary coded decimal type differ from the floating point type?

   Floating point numbers are an approximation of the number using fractions and exponents, stored in binary which can make them even less accurate. Binary coded decimals precisely store decimal values at the cost of taking up more storage than floating point numbers.

10. Identify a user-defined ordinal type in the Java programming language.

    Enumeration

11. Mathematicians and programmers might have different ideas about the precedence of Boolean operators. Explain.

    When Boolean operators evaluate to 0 and 1, they can be used in equations with multiple relational operators to produce unexpected results. (1 ¡ 2 ¡= 1) -¿ ((true/1) ¡= 1) -¿ true

12. Programmers should use `===` rather than `==` to test the equality of the values of two expressions in JavaScript. Why?

    With +==+, strings can be coerced into numbers to check for equality. With +===+, no coercion will take place, so strings and numbers will not resolve as equal.

13. Describe a hazard of allowing short-circuited evaluation of expressions and side effects in expressions at the same time.

    If there is a side effect in the right hand side of the expression and it short-circuits on the left, then the side effect will not occur.

14. Briefly describe the three steps in the mark-sweep algorithm for garbage collection.

1- all cells in heap set indicators to 'garbage' 2- all pointers in the program are traced, and anything which is pointed to is changed to 'not garbage' 3- all cells still marked 'garbage' are reverted to available space

15. What led Yukihiro Matsumoto to create the Ruby programming language?

Matsumoto was dissatisfied with Perl and Python; specifically, that they were not pure object-oriented languages because they had primitive types and functions.

16. What did Microsoft aim to achieve with its development of the C# language?

Microsoft aimed to provide a language for development of component-based software in the .NET framework.

4.1 More questions for discussion and review.

1. The design of which machine influenced the design of the control statements in FORTRAN?

IBM 704 system

2. How many different kinds of control statements must the designer of a programming language include in a language?

While it is possible to use only one (GOTO), the minimum number of control statements in a language which does not use GOTO is two. One for choosing between control flow paths, and one for logically controlled iterations.

3. What is the one question that applies in the design of all statements that allow selection or iteration?

Should the control structure have multiple entries?

4. What is an advantage of requiring that the **then** and **else** clauses of an **if** statement be compound statements?

Requiring compound statements helps increase the readability and writability for programmers when using nested selector statements, that otherwise can get very messy and complicated.

5. How does the **switch** statement in C# differ from the **switch** statement in Java?

In Java, the switch statement does not allow case expressions anywhere except the top level in the body of the switch. C# allows the execution of more than one segment. In C#, the control expressions as well as case statements can be strings.

6. Distinguish between 2 statements in Ruby that correspond to Java's **switch** statement.

Case expressions are the Ruby equivalent of Java's **switch** statement. One is semantically similar to nested if statements with case - when - then. The other is with boolean expressions being evaluated one at a time from top to bottom. The value of this case expression is equivalent to the value of the **then** statement corresponding to the first *true* **when** statement.

7. Features of a programming language sometimes persist longer than a feature of computing hardware that inspired and supported that part of the

language's design. Similarly, features of hardware sometimes persist longer than some parts of a language's design that were created to take advantage of that feature in hardware.

Give examples.

(a) The IBM 704 influenced the design of control statements which are still used today, and prompted the development of Fortran.

(b) The Algol standard used several different syntaxes which, among other things, allowed Europeans to use a comma to denote a decimal point, while Americans could continue to use a period.

(c) Most of the popular languages of the past 50 years have been designed around the von Neumann architecture. These are called imperative languages. In a von Neumann computer, both data and programs are stored in the same memory. The CPU, which executes instructions, is separate from the memory. Therefore, instructions and data must be transmitted, or piped, from memory to the CPU. Results of operations in the CPU must be moved back to memory. Nearly all digital computers built since the 1940s have been based on the von Neumann architecture.

(d) The **register** keyword in C is a hint to the compiler that a variable will be used repeatedly, and so it should be stored in the CPU rather than in memory. However, modern compilers are far better at optimization than programmers, so this keyword is outdated and unnecessary.

8. Who most famously warned of the dangers of using the **goto** statement? What did Donald Knuth have to say about the use of the **goto** statement?

Edsger Dijkstra noted The goto statement as it stands is just too primitive; it is too much an invitation to make a mess of ones program. Donald Knuth argued there were occasions when the efficiency of the goto outweighed its harm to readability.

9. Describes Ada's **for** loop. Are there some kinds of iteration that might be easier in Ada than in Java? Easier in Java than in Ada?

The for loop looks like this:

```
for variable in [reverse] discrete\_range loop     end loop;
```

Adas for loop can use any ordinal type variable for its counter. Arrays with ordinal type subscripts can be conveniently processed.

10. What does it mean to say that the guarded commands of Ada are nondeterministic?

    Guarded commands in Ada are nondeterministically chosen for execution when more than one of the statements are evaluated to true. This means that if there are three guarded statements and two of the three evaluate to true, then each time the program will use one of the two statements. It will not always use the one that appears first nor the one that appears last, but rather it will choose between them non-deterministically or randomly.

11. The header files in a C program contain function prototypes. What is a function prototype?

    A function prototype is a function declaration that gives the function's name and type signature, but does not specify the function body. It is also referred to as a function interface at times. In other languages these are uncommon because subprograms do not need declarations since they do no need to be defined before they are called.

12. Every method in a Ruby program belongs to a class. A programmer can place a definition of a method inside the definition of a class or outside of the definition of any class that the programmer writes. To which class does the method belong in the second case?

    If a method is defined outside of the definition of any class that the programmer writes then the method belongs to the root object, **Object**.

13. Distinguish between positional and keyword parameters.

    Positional parameters are bound based on the order in which they are given to the function. Keyword parameters are when the name of the formal parameter to which an actual parameter is to be bound is specified with the actual parameter in a call. This allows them to be input in any order.

14. Ruby blocks are closures. What does that mean?

    A closure is an anonymous function which can be passed as a parameter. Also, Ruby doesn't nest scope, so variables defined in a method are not accessible outside of the method (say, in the class which defines them).

15. What is a pure function?

    A **pure function** is a **function** where the return value is only determined by its input values, without observable side effects.

16. Some languages give programmers means to define both functions and procedures. Java doe not. Is that a serious limitation?

    It does not seem to be a serious limitation, it just means that performing procedures in Java is more verbose because it requires that the programmer instantiate the class with any needed methods.

17. Declarations of formal parameters in an Ada procedure can include, in addition to the names and types of the parameters, reserved words that do not appear in declarations in Java programs. What is the purpose of those reserved words?

    Ada allows the programmer to specify in mode, out mode, and inout mode for each formal parameter. This means that they can receive data from the actual parameter, they can send data to the actual parameter, or they can do both.

18. The C language imposes a constraint upon programmers who want to pass a multidimensional array to a function. What is the constraint? How did

the design of the Java programming language eliminate that constraint for programmers who use that language?

In C, programmers are required to pass the length of an array to a method. Specifying the length of an array (like in C) is unnecessary because Java arrays store a pointer to an address containing their length.

19. An activation record contains a return address, a dynamic link, parameters, and local variables.

    (a) To what does the return address point?

    (b) To what does the dynamic link point?

    (a) The return address usually consists of a pointer to the instruction following the call in the code segment of the calling program unit.

    (b) The dynamic link points to the base of the activation record instance of the caller.

20. The stack will contain multiple activation records for a single subprogram under what circumstances?

In a recursive program there can be multiple activation records (although they will be incomplete) for a single subprogram.

21. How (or why?) does the LIFO protocol apply to calls to and returns from subprograms?

LIFO allows for subprograms to be nested within one another and for subprograms to be used as parameters within other subprograms.

22. Which important development in computer architecture has changed the way that the stack is used in some systems for facilitating calls to and returns from subprograms?

RISC (reduced instruction set computing) machines have parameters passed in registers in their compilers because RISC machines have more registers than CISC (complex instruction set computing) machines. Chapter 10 assumes parameters are passed in the stack though, as they had been in CISC machines.

23. A dynamic chain contains a history of what?

Dynamic chain represents the dynamic history of how execution got to its current position, which is always in the subprogram code whose activation record instance is on top of the stack.

24. Which two numbers are needed to compute the address of a local variable in a subprogram?

To compute the address of a local variable you need the (chain_offset, local_offset) pair.

25. How does a Ruby module differ from a class?

Modules are unlike classes in that they cannot be instantiated or subclassed and do not define variables. Methods that are defined in a module include the modules name in their names.

26. Memory for variables can be allocated on the heap and on the stack. In which place or places is memory allocated for objects in C++? in Java?

In C++, variables can be allocated to the heap either by making them static or by allocating memory with the keyword new. However, variables which are initialized during the execution of a function are allocated to the stack. Java behaves the same way.

27. What problems were solved by the addition of genericity to Java?

Generics allow a type or method to operate on objects of various types while providing compile-time type safety.

28. What is the purpose of the static chain?

The static chain is a path of pointers which go from each function to its parent. They allow child subprograms to use variables which are local to their parent, grandparent, or farther up, without needing to copy those variables to the call stack.

29. What is a singleton?

A singleton is a class which provides a global access point to a single instance. This is useful for tasks which only need one point of access, like a file system.

30. What are the two parts of the definition of an abstract data type?

(a) A type definition which allows program units to declare variables of the type but hides the representation of objects of the type.

(b) A set of operations for manipulating objects of the type.

Chapter 5

Sam Chalkley

1. Most programming languages require the use of brackets to enclose the index in a reference to an element of an array.

   (a) Identify a language the requires the use of parentheses to enclose the index in a reference to an element of an array.

   (b) Why did the designers of the language choose parentheses rather than brackets?

---

the IBM 704 system influenced the design of control statements in FORTRAN

(a) the language that requires the use of parentheses to enclose an index is ada.

(b) the designers chose parentheses to enclose subscripts so there would be uniformity between array references and function calls in expressions.

2. What is the relationship between a lexeme and a token?

---

the lexemes of a programming language include numeric literals operators and special words. these are partitioned into groups. the token is the name that represents each lexeme group. (a token of a language is a category of its lexemes.)

3. (a) What kind of symbols are found at the internal nodes of a parse tree?

(b) What kind of symbols are found at the leaves of a parse tree?

(a) nonterminal symbols are at every internal node

(b) leaves of the parse tree is labeled with a terminal symbol

4. One of the most significant contributions from the developers of ALGOL 60 also limited the success of that language. What was that contribution?

This is what is called static scopping it is a method of binding names to nonlocal variables. the scope of the variable can be statically determined prior to execution.

5. What problem were the creators of Common LISP trying to solve?

it was created to solve lack of portability among programs written in the various dialects

6. What is an ambiguous context free grammar?

a grammarthat generates a sentential form where there are two or more distinct parse trees.

7. Contrast the complexity of algorithms that can parse strings that conform to the most general kinds of context free grammars and the complexity of the algorithms that can parse strings that conform to the grammars of programming languages?

the conplexity for parsing strings that conform to general kinds of context free grammers is $O(n^3)$ which is alot and takes awhile. while parsing strings that conform to grammars have a complexity of $O(n)$ which is much faster. this indicates that it is linear. and much more efficient ************************ Write your answer here.

8. Java represents characters with Unicode. It is the first widely used programming language with this feature. What is the significance of this feature?

   Unicode is standard for character incoding. this allowed computers to communicate with comuters around the world. this helped with globalization of business because of the demand of computers talking to comuters around the world.

9. How does the binary coded decimal type differ from the floating point type?

   uses binary codefor the decimal digits (much like strings.) this allows precision storing of decimal values. supports business proccessing etc. on the other hand floating points are only approximations for many real values. has some problems like loss of accuracy throug arithmetic operations

10. Identify a user-defined ordinal type in the Java programming language.

    this is an enumeration.

11. Mathematicians and programmers might have different ideas about the precedence of Boolean operators. Explain.

    in math booleans or/ and must have equal precedence. however with computer science, specifically in c we establish a higher precedence in and than or.

12. Programmers should use `===` rather than `==` to test the equality of the values of two expressions in JavaScript. Why?

    because this === prevents operands from being coerced.in this case === with a string and number would turn out false.

13. Describe a hazard of allowing short-circuited evaluation of expressions and side effects in expressions at the same time.

short circut evaluation can be hazardus when used on and expression and part of an expression that contains a side effect is not evaluated. for example $(a > b)||((b++/3)$ if the programmer assumed b would change every time and a was not greater than b the program would fail.

14. Briefly describe the three steps in the mark-sweep algorithm for garbage collection.

1) all cells in the heap have their indicators set to indicate they are garbage 2) marking phase, every pointer is traced into the heap and all reachable cells are marked as not being garbage 3)sweep phase, all cells in the heap that have not been specifically marked as still being used are returned to the list of available space.

15. What led Yukihiro Matsumoto to create the Ruby programming language?

the motivation for ruby was dissatisfaction of its designer with perl and python. niether was a pure object-oriented language in the sense that each hase primitive types and each supports functions.

16. What did Microsoft aim to achieve with its development of the C# language?

the C# designers obviously disagreed with this wholesale removal of featurs that java did not include. it included all but the multiple inheritance. they wanted to have a language for component based software development (looking alot at .NET)

5.1 More questions for discussion and review.

1. The design of which machine influenced the design of the control statements in FORTRAN?

   IBM 704 system influenced the design of the control statments in FORTRAN

2. How many different kinds of control statements must the designer of a programming language include in a language?

   while it is possible to use only one goto, the minimum number of control statements in a language that doesnt use goto is two. first for choosing between control flow paths, and the other for logically controlled iterations.

3. What is the one question that applies in the design of all statements that allow selection or iteration?

   should the control structure have multiple entries?

4. What is an advantage of requiring that the **then** and **else** clauses of an **if** statement be compound statements?

   It adds readability, as well as ensuring the encapsulation of just what you want in your if statment

5. How does the **switch** statement in C# differ from the **switch** statement in Java?

   in java the switch statement does not allow case expressions anywhere exept the top level in the body of the switch. C# allows the execution of more than one segment. In c# the control exressions as well as case statements can be strings

6. Distinguish between 2 statements in Ruby that correspond to Java's **switch** statement.

   Case expressions are the ruby equivalent of java's **switch** statement. One is semantically similar to nested if statements with case -when- then. the other is with boolean expressions being evaluated one at a time from top to bottom. the value of this case exression ends up as the first value with a true corresponding boolean.

7. Features of a programming language sometimes persist longer than a feature of computing hardware that inspired and supported that part of the language's design. Similarly, features of hardware sometimes persist longer than some parts of a language's design that were created to take advantage of that feature in hardware.

   Give examples.

   the IBM 704 influenced the design of control statements

8. Who most famously warned of the dangers of using the **goto** statement? What did Donald Knuth have to say about the use of the **goto** statement?

   Edsger Dijkstra- "the goto statment as it stads is just too primitivel; it is

   too much an invitation to make a mess of one's program."Donald Knuth argued there were occasions when the efficiency of the goto outweighed its harm to readability

9. Describes Ada's **for** loop. Are there some kinds of iteration that might be easier in Ada than in Java? Easier in Java than in Ada?

   with ada the counter variable cannot be changed in the body. the counter

   is edited in the body. you aslo gain flexability by allowing you to edit any part of the loop.

10. What does it mean to say that the guarded commands of Ada are nondeterministic?

    Guarded commands in Ada are nondeterministically chosen for execution

    when more than one of the statements are evaluated to be true, then each time the program will use one of the two statements. It will not always use the one that appears last, but rather it will choose between them randomly.

11. The header files in a C program contain function prototypes. What is a function prototype?

a function prototype is a function declaration that give the functions name and type signature, but does not specify the function body. It is also referred to as a function interface at times. in other languages these are uncommon becuase subprograms do not need declarations since they do not need to be defined before they are called

12. Every method in a Ruby program belongs to a class. A programmer can place a definition of a method inside the definition of a class or outside of the definition of any class that the programmer writes. To which class does the method belong in the second case?

---

if a method is defined outside of the definition of any class that the programmer writes then the method belongs to the root object, **object**

13. Distinguish between positional and keyword parameters.

Positional parameters are bound based on the order in which they are given to the function. Keyword parameters are when the name of the formal parameter to which an actual parameter is to be bound is specified with the actual parameter in a call. this allows them to be input in any order.

14. Ruby blocks are closures. What does that mean?

---

a closure is an anonymous function which can be passed as a parameter. also, Ruby doesnt nest scope, so variables defined in a method are not accessible outside of the method.

15. What is a pure function?

---

a pure function is a function where the return value is only determined by its input values, without observable side effects.

16. Some languages give programmers means to define both functions and procedures. Java doe not. Is that a serious limitation?

---

it does not seem to be a serious limitation, it just means that the preforming procedures in Java are more verbose becase it requires that the programmer instantiate the class with any needed methods.

17. Declarations of formal parameters in an Ada procedure can include, in addition to the names and types of the parameters, reserved words that do not appear in declarations in Java programs. What is the purpose of those reserved words?

Ada allows the programmer to specify in mode, out mode, and inout mode

for each formal parameter. This means that they can receive data from the actual parameter, they can send data to the actual parameter, or they can do both.

18. The C language imposes a constraint upon programmers who want to pass a multidimensional array to a function. What is the constraint? How did the design of the Java programming language eliminate that constraint for programmers who use that language?

---

you need to tell the function how many columns in the array. java got rid of this allowing the array to tell the function the number of columns

19. An activation record contains a return address, a dynamic link, parameters, and local variables.

    (a) To what does the return address point?

    (b) To what does the dynamic link point?

The return address usually consists of a pointer to the instruction follow-

ing the call in the code segment of the calling program unit. The dynamic link points to the base of the activation record instance of the caller.

20. The stack will contain multiple activation records for a single subprogram under what circumstances?

In a recursive program there can be multiple activation records (although they will be incomplete) for a single subprogram.

21. How (or why?) does the LIFO protocol apply to calls to and returns from

subprograms?

LIFO allows for subprograms to be nested within one another and for subprograms to be used as parameters within other subprograms. (417)

22. Which important development in computer architecture has changed the way that the stack is used in some systems for facilitating calls to and returns from subprograms?

    RISC (reduced instruction set computing) machines have parameters passed in registers in their compilers because RISC machines have more registers than CISC (complex instruction set computing) machines. Chapter 10 assumes parameters are passed in the stack though, as they had been in CISC machines.

23. A dynamic chain contains a history of what?

    A dynamic chain represents the dynamic history of how the execution got to its current position, which is always in the subprogram code whose activation record instance is on top of the stack.

24. Which two numbers are needed to compute the address of a local variable in a subprogram?

    To compute the address of a local variable you need the (chain_offset, local_offset) pair.

25. How does a Ruby module differ from a class?

    Modules are unlike classes in that they cannot be instantiated or subclassed and do not define variables. Methods that are defined in a module include the modules name in their names.

26. Memory for variables can be allocated on the heap and on the stack. In which place or places is memory allocated for objects in C++? in Java?

    In C++, variables can be allocated to the heap either by making them static or by allocating memory with the keyword new. However, variables which are initialized during the execution of a function are allocated to the stack. Java behaves the same way.

27. What problems were solved by the addition of genericity to Java?

Generics allow a type or method to operate on objects of various types while providing compile-time type safety.

28. What is the purpose of the static chain?

The static chain is a path of pointers which go from each function to its parent. They allow child subprograms to use variables which are local to their parent, grandparent, or farther up, without needing to copy those variables to the call stack.

29. What is a singleton?

A singleton is a class which provides a global access point to a single instance. This is useful for tasks which only need one point of access, like a file system.

30. What are the two parts of the definition of an abstract data type?

1. A type definition which allows program units to declare variables of the type but hides the representation of objects of the type. 2. A set of operations for manipulating objects of the type.

Chapter 6

Nicci Geiger

1. Most programming languages require the use of brackets to enclose the index in a reference to an element of an array.

   (a) Identify a language the requires the use of parentheses to enclose the index in a reference to an element of an array.

   (b) Why did the designers of the language choose parentheses rather than brackets?

   (a) Ada uses parentheses to enclose the index refrence to the element of an array.

   (b) This was done for uniformity between array refrences and function call expressions because they are both mappings.

2. What is the relationship between a lexeme and a token?

   A lexemes is the lowest-level syntatic units. Lexemes are patitioned into groups which are represented by tokens.

3. (a) What kind of symbols are found at the internal nodes of a parse tree?

   (b) What kind of symbols are found at the leaves of a parse tree?

(a) Nonterminal symobols are found at internal nodes.

(b) Terminal symbols are found at leaves.

4. One of the most significant contributions from the developers of ALGOL 60 also limited the success of that language. What was that contribution?

The contribution was BNF (Backus-Naur Form) a natural notation for describing syntax.

5. What problem were the creators of Common LISP trying to solve?

They were trying solve the issue of lack of portability in programs written in various dialects.

6. What is an ambiguous context free grammar?

A grammar that generates a sentential for that has two or more parse trees.

7. Contrast the complexity of algorithms that can parse strings that conform to the most general kinds of context free grammars and the complexity of the algorithms that can parse strings that conform to the grammars of programming languages?

The complexity of context free grammars is $O(n^3)$, while the grammars of programing languages have a complexity of $O(n)$ due to the fact that context free grammar parsers must frequently be backed up and reparse part of what is being analyzed.

8. Java represents characters with Unicode. It is the first widely used programming language with this feature. What is the significance of this

feature?

It uses a set of charactes that covers characters for most of the worlds alphabet. natural languages. This caters to the need for global computer communication.

9. How does the binary coded decimal type differ from the floating point type?

Float points type represent real numbers but are only approximate and are stored in binary and represented in fractions or exponents, while decimal type stores a fixed number of decimal points and precicely store them with a restricted range.

10. Identify a user-defined ordinal type in the Java programming language.

Java has enumeration types so long as it is after Java 1.4.

11. Mathematicians and programmers might have different ideas about the precedence of Boolean operators. Explain.

Mathematicians have the OR and AND opperators with equal precedence while programmers have AND with a higher precedence than OR.

12. Programmers should use === rather than == to test the equality of the values of two expressions in JavaScript. Why?

The === over == prevents coersion of one type into another such as a string that is numbers to be coerced into a number rather than being a string containing the number.

13. Describe a hazard of allowing short-circuited evaluation of expressions and side effects in expressions at the same time.

Short-circuit evaluations used in an expression with part of the expression being a side effectthat is not evaluated. The side effect will only occur in complete evaluation of the whole ezxpression. This causes the side effects in expressions to allow for subtle errors.

14. Briefly describe the three steps in the mark-sweep algorithm for garbage collection.

The first are cells in the heap have indicators set to garbage. The second is when every pointer in the program is traced to the heap and reachables are marked to not be garbage. The third is that all cells not marked as being used are returned to being useable space.

15. What led Yukihiro Matsumoto to create the Ruby programming language?

There was a dissatisfaction with Perl and Python and the designers as both supported object oriented programing but neither were purely object oriented programing.

16. What did Microsoft aim to achieve with its development of the C# language?

C# was aimed to be a component-based software development in the .NET framework.

6.1 More questions for discussion and review.

1. The design of which machine influenced the design of the control statements in FORTRAN?

   IBM 704 system

2. How many different kinds of control statements must the designer of a programming language include in a language?

   While it is possible to use only one (GOTO), the minimum number of control statements in a language which does not use GOTO is two. One for choosing between control flow paths, and one for logically controlled iterations. (348)

3. What is the one question that applies in the design of all statements that allow selection or iteration?

   Should the control structure have multiple entries? (349)

4. What is an advantage of requiring that the **then** and **else** clauses of an **if** statement be compound statements?

5. How does the **switch** statement in C# differ from the **switch** statement in Java?

6. Distinguish between 2 statements in Ruby that correspond to Java's **switch** statement.

   Requiring compound statements helps increase the readability and writability for programmers when using nested selector statements, that otherwise can get very messy and complicated.

7. How does the **switch** statement in C# differ from the **switch** statement in Java?

   In Java, the switch statement does not allow case expressions anywhere except the top level in the body of the switch. C# allows the execution of more than one segment. In C#, the control expressions as well as case statements can be strings. (357)

8. Distinguish between 2 statements in Ruby that correspond to Java's **switch** statement.

Case expressions are the Ruby equivalent of Java's **switch** statement. One is semantically similar to nested if statements with case - when - then. The other is with boolean expressions being evaluated one at a time from top to bottom .The value of this case expression ends up as the first value with a true corresponding boolean. (357-358)

9. Features of a programming language sometimes persist longer than a feature of computing hardware that inspired and supported that part of the language's design. Similarly, features of hardware sometimes persist longer than some parts of a language's design that were created to take advantage of that feature in hardware.

   Give examples.

The IBM 704 influenced the design of control statements which are still used today, and prompted the development of Fortran. (348, 42) Most of the popular languages of the past 50 years have been designed around the von Neumann architecture. These are called imperative languages. In a von Neumann computer, both data and programs are stored in the same memory. The CPU, which executes instructions, is separate from the memory. Therefore, instructions and data must be transmitted, or piped, from memory to the CPU. Results of operations in the CPU must be moved back to memory. Nearly all digital computers built since the 1940s have been based on the von Neumann architecture. The Algol standard used several different syntaxes which, among other things, allowed Europeans to use a comma to denote a decimal point, while Americans could continue to use a period. The **register** keyword in C is a hint to the compiler that a variable will be used repeatedly, and so it should be stored in the CPU rather than in memory. However, modern compilers are far better at optimization than programmers, so this keyword is outdated and unnecessary.

10. Who most famously warned of the dangers of using the **goto** statement? What did Donald Knuth have to say about the use of the **goto** statement?

Edsger Dijkstra noted The goto statement as it stands is just too primitive; it is too much an invitation to make a mess of ones program. Donald Knuth argued there were occasions when the efficiency of the goto outweighed its harm to readability. (376)

11. Describes Ada's **for** loop. Are there some kinds of iteration that might be easier in Ada than in Java? Easier in Java than in Ada?

12. What does it mean to say that the guarded commands of Ada are nondeterministic?

13. The header files in a C program contain function prototypes. What is a function prototype?

    The for loop looks like this

    ```
    for variable in [reverse] discrete_range loop     end loop;
    ```

    Adas for loop can use any ordinal type variable for its counter. Arrays with ordinal type subscripts can be conveniently processed. (261)

14. What does it mean to say that the guarded commands of Ada are nondeterministic?

    Guarded commands in Ada are nondeterministically chosen for execution

    when more than one of the statements are evaluated to true. This means that if there are three guarded statements and two of the three evaluate to true, then each time the program will use one of the two statements. It will not always use the one that appears first nor the one that appears last, but rather it will choose between them non-deterministically or randomly.

15. The header files in a C program contain function prototypes. What is a function prototype?

    A function prototype is a function declaration that gives the function's

    name and type signature, but does not specify the function body. It is also referred to as a function interface at times. In other languages theses are uncommon because subprograms do not need declarations since they do no need to be defined before they are called.

16. Every method in a Ruby program belongs to a class. A programmer can place a definition of a method inside the definition of a class or outside of the definition of any class that the programmer writes. To which class does the method belong in the second case?

17. Distinguish between positional and keyword parameters.

18. Ruby blocks are closures. What does that mean?

19. What is a pure function?

20. Some languages give programmers means to define both functions and procedures. Java doe not. Is that a serious limitation?

If a method is defined outside of the definition of any class that the programmer writes then the method belongs to the root object, **Object**.

21. Distinguish between positional and keyword parameters.

Positional parameters are bound based on the order in which they are given to the function. Keyword parameters are when the name of the formal parameter to which an actual parameter is to be bound is specified with the actual parameter in a call. This allows them to be input in any order

22. Ruby blocks are closures. What does that mean?

A closure is an anonymous function which can be passed as a parameter. Also, Ruby doesn't nest scope, so variables defined in a method are not accessible outside of the method (say, in the class which defines them).

23. What is a pure function?

A pure function is a function where the return value is only determined by its input values, without observable side effects.

24. Some languages give programmers means to define both functions and procedures. Java does not. Is that a serious limitation?

It does not seem to be a serious limitation, it just means that performing procedures in Java is more verbose because it requires that the programmer instantiate the class with any needed methods.

25. Declarations of formal parameters in an Ada procedure can include, in addition to the names and types of the parameters, reserved words that do not appear in declarations in Java programs. What is the purpose of those reserved words?

Ada allows the programmer to specify in mode, out mode, and inout mode for each formal parameter. This means that they can receive data from the actual parameter, they can send data to the actual parameter, or they can do both.

26. The C language imposes a constraint upon programmers who want to pass a multidimensional array to a function. What is the constraint? How did the design of the Java programming language eliminate that constraint for programmers who use that language?

In C, programmers are required to pass the length of an array to a method.

Specifying the length of an array (like in C) is unnecessary because Java arrays store a pointer to an address containing their length.

27. An activation record contains a return address, a dynamic link, parameters, and local variables.

    (a) To what does the return address point?

    (b) To what does the dynamic link point?

    The return address usually consists of a pointer to the instruction following the call in the code segment of the calling program unit.

    (c) To what does the dynamic link point?

    The dynamic link points to the base of the activation record instance of the caller.

28. The stack will contain multiple activation records for a single subprogram under what circumstances?

29. How (or why?) does the LIFO protocol apply to calls to and returns from subprograms?

In a recursive program there can be multiple activation records (although they will be incomplete) for a single subprogram.

30. How (or why?) does the LIFO protocol apply to calls to and returns from subprograms?

LIFO allows for subprograms to be nested within one another and for subprograms to be used as parameters within other subprograms. (417)

31. Which important development in computer architecture has changed the way that the stack is used in some systems for facilitating calls to and returns from subprograms?

32. A dynamic chain contains a history of what?

33. Which two numbers are needed to compute the address of a local variable in a subprogram?

34. How does a Ruby module differ from a class?

RISC (reduced instruction set computing) machines have parameters passed in registers in their compilers because RISC machines have more registers than CISC (complex instruction set computing) machines. Chapter 10 assumes parameters are passed in the stack though, as they had been in CISC machines.

35. A dynamic chain contains a history of what?

Dynamic chain represents the dynamic history of how execution got to its current position, which is always in the subprogram code whose activation record instance is on top of the stack.

36. Which two numbers are needed to compute the address of a local variable in a subprogram?

To compute the address of a local variable you need the (chain_offset, local_offset) pair.

37. How does a Ruby module differ from a class?

Modules are unlike classes in that they cannot be instantiated or subclassed and do not define variables. Methods that are defined in a module include the modules name in their names.

38. Memory for variables can be allocated on the heap and on the stack. In which place or places is memory allocated for objects in C++? in Java?

39. What problems were solved by the addition of genericity to Java?

40. What is the purpose of the static chain?

41. What is a singleton?

42. What are the two parts of the definition of an abstract data type?

In C++, variables can be allocated to the heap either by making them static or by allocating memory with the keyword new. However, variables which are initialized during the execution of a function are allocated to the stack. Java behaves the same way. (216-217)

43. What problems were solved by the addition of genericity to Java?

Generics allow a type or method to operate on objects of various types while providing compile-time type safety.

44. What is the purpose of the static chain?

The static chain is a path of pointers which go from each function to its parent. They allow child subprograms to use variables which are local to their parent, grandparent, or farther up, without needing to copy those variables to the call stack.

45. What is a singleton?

A singleton is a class which provides a global access point to a single instance. This is useful for tasks which only need one point of access, like a file system.

46. What are the two parts of the definition of an abstract data type?

---

1. A type definition which allows program units to declare variables of the type but hides the representation of objects of the type. 2. A set of operations for manipulating objects of the type.

Chapter 7

Brian Hixson-Simeral

1. Most programming languages require the use of brackets to enclose the index in a reference to an element of an array.

   (a) Identify a language the requires the use of parentheses to enclose the index in a reference to an element of an array.

   (b) Why did the designers of the language choose parentheses rather than brackets?

   (a) Ada uses parentheses to enclose the index to an element of an array.

   (b) The reason that brackets are used, rather than parentheses, is that parentheses are also used to denote subprogram calls.

2. What is the relationship between a lexeme and a token?

   Tokens are the catagorey of the lexeme. Eg. 2 and int_literal (lexeme and token).

3. (a) What kind of symbols are found at the internal nodes of a parse tree?

   (b) What kind of symbols are found at the leaves of a parse tree?

   (a) Nonterminal

(b) Terminal

4. One of the most significant contributions from the developers of ALGOL 60 also limited the success of that language. What was that contribution?

   BNF (Backus-Naur Form)

5. What problem were the creators of Common LISP trying to solve?

   They were trying to create one version of LISP, so that there wouldn't be so many dialects being used. With the large amount of dialects came a lack of potability.

6. What is an ambiguous context free grammar?

   A context-free grammar is a generative device for defining languages. Context-free gramars are ambiguous when they generate a sentential form that could have two or more distinct parse trees.

7. Contrast the complexity of algorithms that can parse strings that conform to the most general kinds of context free grammars and the complexity of the algorithms that can parse strings that conform to the grammars of programming languages?

   A parsing algorithm for an unambiguous grammar is ridiculously inefficient ($O(n^3)$). More specifc alogorithms can be made for programming languages that have a complexity of $O(n)$. It is much more efficient to use specific algorithms.

8. Java represents characters with Unicode. It is the first widely used programming language with this feature. What is the significance of this

feature?

It was a 16-bit character set that included characters from most natural languages and ASCII.

9. How does the binary coded decimal type differ from the floating point type?

Floating-point are represented as fractions and exponents while Decimal are stored with a fixed number of decimal digits with the decimal point at a fixed position in the value. The value 0.1 can be represented exactly in decimal, but in floating-point it would come with some uncertainty.

10. Identify a user-defined ordinal type in the Java programming language.

The two user-defined ordinal types in Java are enumeration and subrange.

11. Mathematicians and programmers might have different ideas about the precedence of Boolean operators. Explain.

In math, AND and OR have equal precedence, but in most programming languages AND has a higher precedence than OR.

12. Programmers should use `===` rather than `==` to test the equality of the values of two expressions in JavaScript. Why?

Programmers should use === instead of == because when you use == a string such as ”7” will be coerced to the number 7, but when you use === it remains a string.

13. Describe a hazard of allowing short-circuited evaluation of expressions and side effects in expressions at the same time.

This allows subtle errors to occur. "If the short-circuit evaluation is used on an expression and part od the expression that contains a side effect is not evaluated; the the side effect will occur only in complete evaluations of the wole expression. If program correctness depends on the side effect, short-circuit evaluation can result in a serious error."

14. Briefly describe the three steps in the mark-sweep algorithm for garbage collection.

All cells in the heap have their indicators set to indicate that they are garbage. Every pointer in the program is traced into the heap, and all reachable cells are marked as not being garbage. All cells in the heap that are marked as garbage are returned to the list of available space.

15. What led Yukihiro Matsumoto to create the Ruby programming language?

He was dissatisfied with Perl and Python. He wanted a purely object-oriented language and neither of them lived up to it.

16. What did Microsoft aim to achieve with its development of the C# language?

C# was meant to provide a language for component-based software development. It was geared towards development in the .NET Framework.

7.1 More questions for discussion and review.

1. The design of which machine influenced the design of the control statements in FORTRAN?

IBM 704

2. How many different kinds of control statements must the designer of a programming language include in a language?

Two, one for choosing between two different control flow paths and one for logically controlled iterations, or one if it is something akin to a selectable goto.

3. What is the one question that applies in the design of all statements that allow selection or iteration?

Should the control structure have multiple entries?

4. What is an advantage of requiring that the **then** and **else** clauses of an **if** statement be compound statements?

It helps with readability, and in certain languages there are no other indicators of whether or not something is apart of the if statement.

5. How does the **switch** statement in C# differ from the **switch** statement in Java?

In C# it will only evaluate the first true **switch** statement, whereas in java it will look to evaluate all **switch** statements.

6. Distinguish between 2 statements in Ruby that correspond to Java's **switch** statement.

Case expressions are the Ruby equivalent of Java's **switch** statement. One is semantically similar to nested if statements with case - when - then. The other is with boolean expressions being evaluated one at a time from top to bottom. The value of this case expression is equivalent to the value of the **then** statement corresponding to the first *true* **when** statement.

7. Features of a programming language sometimes persist longer than a feature of computing hardware that inspired and supported that part of the language's design. Similarly, features of hardware sometimes persist longer than some parts of a language's design that were created to take advantage of that feature in hardware.

   Give examples.

The IBM 704 influenced the design of control statements which are still

used today, and prompted the development of Fortran. (348, 42) Most of the popular languages of the past 50 years have been designed around the von Neumann architecture. These are called imperative languages. In a von Neumann computer, both data and programs are stored in the same memory. The CPU, which executes instructions, is separate from the memory. Therefore, instructions and data must be transmitted, or piped, from memory to the CPU. Results of operations in the CPU must be moved back to memory. Nearly all digital computers built since the 1940s have been based on the von Neumann architecture. The Algol standard used several different syntaxes which, among other things, allowed Europeans to use a comma to denote a decimal point, while Americans could continue to use a period. The **register** keyword in C is a hint to the compiler that a variable will be used repeatedly, and so it should be stored in the CPU rather than in memory. However, modern compilers are far better at optimization than programmers, so this keyword is outdated and unnecessary.

8. Who most famously warned of the dangers of using the **goto** statement? What did Donald Knuth have to say about the use of the **goto** statement?

Edsger Dijkstra noted The goto statement as it stands is just too primi-

tive; it is too much an invitation to make a mess of ones program. Donald Knuth argued there were occasions when the efficiency of the goto outweighed its harm to readability.

9. Describes Ada's **for** loop. Are there some kinds of iteration that might be easier in Ada than in Java? Easier in Java than in Ada?

Ada's **for** loop limits the scope of the variables inside it. If there is a variable described outside the loop then that variable will be the same after it ends.

10. What does it mean to say that the guarded commands of Ada are nondeterministic?

Gaurded commands non-deterministically chosen for execution when more than one evaluates to true. Of the true statmentes, it will choose one randomly.

11. The header files in a C program contain function prototypes. What is a function prototype?

A function prototype is a function declaration that gives the function's name and type signature, but does not specify the function body. It is also referred to as a function interface at times. In other languages theses are uncommon because subprograms do not need declarations since they do no need to be defined before they are called.

12. Every method in a Ruby program belongs to a class. A programmer can place a definition of a method inside the definition of a class or outside of the definition of any class that the programmer writes. To which class does the method belong in the second case?

If a method is defined outside of the definition of any class that the programmer writes then the method belongs to the root object, **Object**.

13. Distinguish between positional and keyword parameters.

Posistional parameters are refered to by the position that they are called in. Keyword parameters are refered to by the keyword associated with them.

14. Ruby blocks are closures. What does that mean?

This means that a Ruby block contains a nested subprogram and its refrencing envrionment, which allows the subprogram to be called from anywhere in the program.

15. What is a pure function?

A function that produces no side effects; that is, it modifies neither its parameters nor any variable defined outside the function.

16. Some languages give programmers means to define both functions and procedures. Java does not. Is that a serious limitation?

Java does not allow the definition of procedures. It is not a limitation because functions can return void, which is essentialy the same as returning nothing.

17. Declarations of formal parameters in an Ada procedure can include, in addition to the names and types of the parameters, reserved words that do not appear in declarations in Java programs. What is the purpose of those reserved words?

Ada allows the programmer to specify in mode, out mode, and inout mode for each formal parameter. This means that they can receive data from the actual parameter, they can send data to the actual parameter, or they can do both.

18. The C language imposes a constraint upon programmers who want to pass a multidimensional array to a function. What is the constraint? How did the design of the Java programming language eliminate that constraint for programmers who use that language?

In C and C++ you need to specify the number of columns in the multidimensional array. In Java and C# arrays are objects. They are all single dimensioned, but the elements can be arrays. Each array inherits a named constant (length in Java and Length in C#) that is set to the length of the array when the array object is created.

19. An activation record contains a return address, a dynamic link, parameters, and local variables.

    (a) To what does the return address point?

    (b) To what does the dynamic link point?

The return address usually consists of a pointer to the instruction following the call in the code segment of the calling program unit.

The dynamic link points to the base of the activation record instance of the caller.

20. The stack will contain multiple activation records for a single subprogram under what circumstances?

In a recursive program there can be multiple activation records (although they will be incomplete) for a single subprogram.

21. How (or why?) does the LIFO protocol apply to calls to and returns from subprograms?

Last in first out applies because the last thing on the activation record is the first thing that's returned (Stack). LIFO allows for subprograms to be nested within one another and for subprograms to be used as parameters within other subprograms.

22. Which important development in computer architecture has changed the way that the stack is used in some systems for facilitating calls to and returns from subprograms?

RISC (reduced instruction set computing) machines have parameters passed in registers in their compilers because RISC machines have more registers than CISC (complex instruction set computing) machines. Chapter 10 assumes parameters are passed in the stack though, as they had been in CISC machines.

23. A dynamic chain contains a history of what?

It contains a history of all subprogram activation records, but in the reverse order of when they were activated. (LIFO Stack)

24. Which two numbers are needed to compute the address of a local variable in a subprogram?

To compute the address of a local variable you need the (chain_offset, local_offset) pair.

25. How does a Ruby module differ from a class?

Modules are unlike classes in that they cannot be instantiated or subclassed and do not define variables. Methods that are defined in a module include the modules name in their names.

26. Memory for variables can be allocated on the heap and on the stack. In which place or places is memory allocated for objects in C++? in Java?

n C++, variables can be allocated to the heap either by making them static or by allocating memory with the keyword new. However, variables which are initialized during the execution of a function are allocated to the stack. Java behaves the same way.

27. What problems were solved by the addition of genericity to Java?

Before generics you could get a runtime exception, even though your code compiled, when trying to convert data types. Generics allow a type or method to operate on objects of various types while providing compile-time type safety.

28. What is the purpose of the static chain?

The static chain is a path of pointers which go from each function to its parent. They allow child subprograms to use variables which are local to their parent, grandparent, or farther up, without needing to copy those variables to the call stack.

29. What is a singleton?

A singleton is a class which provides a global access point to a single instance. This is useful for tasks which only need one point of access, like a file system.

30. What are the two parts of the definition of an abstract data type?

---

1. A type definition which allows program units to declare variables of

the type but hides the representation of objects of the type. 2. A set of operations for manipulating objects of the type.

Chapter 8

Tuli Leota

1. Most programming languages require the use of brackets to enclose the index in a reference to an element of an array.

   (a) Identify a language the requires the use of parentheses to enclose the index in a reference to an element of an array.

   (b) Why did the designers of the language choose parentheses rather than brackets?

   (a) Write your answer here.

   (b) Write your answer here.

2. What is the relationship between a lexeme and a token?

   Write your answer here.

3. (a) What kind of symbols are found at the internal nodes of a parse tree?

   (b) What kind of symbols are found at the leaves of a parse tree?

   (a) Write your answer here.

   (b) Write your answer here.

4. One of the most significant contributions from the developers of ALGOL 60 also limited the success of that language. What was that contribution?

Write your answer here.

5. What problem were the creators of Common LISP trying to solve?

Write your answer here.

6. What is an ambiguous context free grammar?

Write your answer here.

7. Contrast the complexity of algorithms that can parse strings that conform to the most general kinds of context free grammars and the complexity of the algorithms that can parse strings that conform to the grammars of programming languages?

Write your answer here.

8. Java represents characters with Unicode. It is the first widely used programming language with this feature. What is the significance of this feature?

Write your answer here.

9. How does the binary coded decimal type differ from the floating point type?

Write your answer here.

10. Identify a user-defined ordinal type in the Java programming language.

Write your answer here.

11. Mathematicians and programmers might have different ideas about the precedence of Boolean operators. Explain.

Write your answer here.

12. Programmers should use `===` rather than `==` to test the equality of the values of two expressions in JavaScript. Why?

Write your answer here.

13. Describe a hazard of allowing short-circuited evaluation of expressions and side effects in expressions at the same time.

Write your answer here.

14. Briefly describe the three steps in the mark-sweep algorithm for garbage collection.

Write your answer here.

15. What led Yukihiro Matsumoto to create the Ruby programming language?

    Write your answer here.

16. What did Microsoft aim to achieve with its development of the C# language?

    Write your answer here.

8.1 More questions for discussion and review.

1. The design of which machine influenced the design of the control statements in FORTRAN?

    The IBM 704 influenced the design of the control statements in FORTRAN.

2. How many different kinds of control statements must the designer of a programming language include in a language?

    While it is possible to use only one goto, the minimum number of conrol statements in a language that doesn't use goto is two. The first for choosing between control flow paths, and the other for logically controlled iterations.

3. What is the one question that applies in the design of all statements that allow selection or iteration?

    Should the control structure have mulitiple entries?

4. What is an advantage of requiring that the **then** and **else** clauses of an **if** statement be compound statements?

Requiring compound statements helps increase the readability and writability for programmers when using nested selector statements, that otherwise can get very messy and complicated.

5. How does the **switch** statement in C# differ from the **switch** statement in Java?

In Java, the switch statment doesn't allow case expressions anywhere except the top level of the body of the switch. C allows the execution of more than one segment. In C the control expressions as well as case statements can be strings.

6. Distinguish between 2 statements in Ruby that correspond to Java's **switch** statement.

Case expressions are the Ruby equivalent of Java's **switch** statement. One is semantically similar to nested if statements with case - when - then. The other is with boolean expressions being evaluated one at a time from top to bottom. The value of this case expression is equivalent to the value of the **then** statement corresponding to the first *true* **when** statement.

7. Features of a programming language sometimes persist longer than a feature of computing hardware that inspired and supported that part of the language's design. Similarly, features of hardware sometimes persist longer than some parts of a language's design that were created to take advantage of that feature in hardware.

   Give examples.

The IBM 704 influenced the design of control statements which are still used today, and prompted the development of Fortran. Von Neumann architecture. The Algol standard used several different syntaxes which, among other things, allowed Europeans to use a comma to denote a decimal point, while Americans could continue to use a period.

8. Who most famously warned of the dangers of using the **goto** statement? What did Donald Knuth have to say about the use of the **goto** statement?

Edsger Dijkstra noted that the goto statement was too primitive and could easily make a mess of one's program. Donald Knuth argued that there were occasions when the efficiency of the goto outweighed its harm to readability.

9. Describes Ada's **for** loop. Are there some kinds of iteration that might be easier in Ada than in Java? Easier in Java than in Ada?

The counter variable can't be changed in the body. You gain flexibility by allowing you to edit any part of the loop.

10. What does it mean to say that the guarded commands of Ada are nondeterministic?

Guarded commands in Ada are nondeterministically chosen for execution when more than one of the statements are evaluated to true. This means that if there are three guarded statements and two of the three evaluate to true, then each time the program will use one of the two statements. It will not always use the one that appears first nor the one that appears last, but rather it will choose between them non-deterministically or randomly.

11. The header files in a C program contain function prototypes. What is a function prototype?

A function prototype is a function declaration that gives the function's name and type signature, but does not specify the function body. It is also referred to as a function interface at times. In other languages these are uncommon because subprograms do not need declarations since they do no need to be defined before they are called.

12. Every method in a Ruby program belongs to a class. A programmer can place a definition of a method inside the definition of a class or outside of the definition of any class that the programmer writes. To which class does the method belong in the second case?

If a method is defined outside of the definition of any class that the programmer writes then the method belongs to the root object, **Object**.

13. Distinguish between positional and keyword parameters.

Positional parameters are bound based on the order in which they are given to the function. Keyword parameters are when the name of the formal parameter to which an actual parameter is to be bound is specified with the actual parameter in a call. This allows them to be input in any order.Positional parameters are bound based on the order in which they are given to the function. Keyword parameters are when the name of the formal parameter to which an actual parameter is to be bound is specified with the actual parameter in a call. This allows them to be input in any order.

14. Ruby blocks are closures. What does that mean?

A closure is an anonymous function which can be passed as a parameter. Also, Ruby doesn't nest scope, so variables defined in a method are not accessible outside of the method (say, in the class which defines them).

15. What is a pure function?

A pure function is a function where the return value is only determined by its input values, without observable side effects.

16. Some languages give programmers means to define both functions and procedures. Java doe not. Is that a serious limitation?

It does not seem to be a serious limitation, it just means that performing procedures in Java is more verbose because it requires that the programmer instantiate the class with any needed methods.

17. Declarations of formal parameters in an Ada procedure can include, in addition to the names and types of the parameters, reserved words that do not appear in declarations in Java programs. What is the purpose of those reserved words?

Ada allows the programmer to specify in mode, out mode, and inout mode for each formal parameter. This means that they can receive data from the actual parameter, they can send data to the actual parameter, or they can do both.

18. The C language imposes a constraint upon programmers who want to pass a multidimensional array to a function. What is the constraint? How did the design of the Java programming language eliminate that constraint for programmers who use that language?

In C, programmers are required to pass the length of an array to a method. Specifying the length of an array (like in C) is unnecessary because Java arrays store a pointer to an address containing their length.

19. An activation record contains a return address, a dynamic link, parameters, and local variables.

    (a) To what does the return address point?

    (b) To what does the dynamic link point?

    The return address consists of a pointer to the instruction following the call in the code segment of the calling program unit. The dynamic link points to the base of the activation record instance of the caller.

20. The stack will contain multiple activation records for a single subprogram under what circumstances?

In a recursive program although they will be incomplete.

21. How (or why?) does the LIFO protocol apply to calls to and returns from subprograms?

LIFO allows for subprograms to be nested within one another and for subprograms to be used as parameters within other subprograms.

22. Which important development in computer architecture has changed the way that the stack is used in some systems for facilitating calls to and returns from subprograms?

RISC (reduced instruction set computing) machines have parameters passed in registers in their compilers because RISC machines have more registers than CISC (complex instruction set computing) machines. Chapter 10 assumes parameters are passed in the stack though, as they had been in CISC machines.

23. A dynamic chain contains a history of what?

A dynamic chain contains history of how execution got to its current position, which is always in the subprogram code whose activation record instance is on top of the stack.

24. Which two numbers are needed to compute the address of a local variable in a subprogram?

You need the (chain_offset, local_offset) pair.

25. How does a Ruby module differ from a class?

Modules differ from classes because they can't be instantiated or subclassed and don't define variables. Methods that are defined in a module include the modules name in their names.

26. Memory for variables can be allocated on the heap and on the stack. In which place or places is memory allocated for objects in C++? in Java?

In C++ variables can be allocated on the heap either by making them static or by allocating memory with the keyword "new". Java is the same.

27. What problems were solved by the addition of genericity to Java?

Genericity allow a type or method to operate on objects of various types while providing compile-time type safety.

28. What is the purpose of the static chain?

The static chain is a path of pointers which go from each function to its parent. They allow child subprograms to use variables which are local to their parent, grandparent, or further up, without needing to copy those variables to the call stack.

29. What is a singleton?

A singleton is a class that provides a global access point to a single instance.

30. What are the two parts of the definition of an abstract data type?

1.) A type definition which allows program units to declare variables of the type, but hides the representation of it. 2.) A set of operations for manipulating objects of the type.

1. Most programming languages require the use of brackets to enclose the index in a reference to an element of an array.

   (a) Identify a language the requires the use of parentheses to enclose the index in a reference to an element of an array.

   (b) Why did the designers of the language choose parentheses rather than brackets?

(a) pre-90 FORTRAN or PL/I

(b) No other suitable characters were available

2. What is the relationship between a lexeme and a token?

A token is a category of a language's lexemes, and a lexeme is the lowest-level syntactic units. Lexemes include identifiers, literals, operators, and special words.

3. (a) What kind of symbols are found at the internal nodes of a parse tree?

   (b) What kind of symbols are found at the leaves of a parse tree?

(a) Nonterminal symbols are found at the internal nodes of the parse tree.

(b) Terminal symbols are found at the leaves of the parse tree

4. One of the most significant contributions from the developers of ALGOL 60 also limited the success of that language. What was that contribution?

   BNF (Backus-Naur form) became the way to describe programming languages

5. What problem were the creators of Common LISP trying to solve?

   They were trying to solve the problem of the lack of portability of programs written in different dialects.

6. What is an ambiguous context free grammar?

   It is a grammmar that generates more than one distinct parse tree, because it specifies less syntactic structure.

7. Contrast the complexity of algorithms that can parse strings that conform to the most general kinds of context free grammars and the complexity of the algorithms that can parse strings that conform to the grammars of programming languages?

   Parsing algorithms that parse strings of the most general grammars have a complexity of $O(n^3)$, while algorithms that parse the grammmars of programming languages have a complexity of $O(n)$. Algorithms for general grammars take much longer than those used for grammars of programming languages. It takes on order of the cube of the length of string to be parsed using a general grammar algorithm, while it only takes a linear relation of the string to be parsed in the algorithms for programming languages.

8. Java represents characters with Unicode. It is the first widely used programming language with this feature. What is the significance of this feature?

The language now had a way to encode all of the world's languages using a global standard.This limited the confusion between computers

9. How does the binary coded decimal type differ from the floating point type?

Decimal types can precisely store decimal values, but floating-points cannot. It also takes more memory to store decimals than it does to store numbers in binary.

10. Identify a user-defined ordinal type in the Java programming language.

Java has enumeration types, where all the possible values are provided in the definition.

11. Mathematicians and programmers might have different ideas about the precedence of Boolean operators. Explain.

Mathematicians have a hierarchy of boolean operations, just like most programming languages. In some programming languages, boolean operators have a higher precedence than relational operators, so programmers will have a different order doing things.

12. Programmers should use `===` rather than `==` to test the equality of the values of two expressions in JavaScript. Why?

If a (===) is used, then no type conversion is done, so the types must be the same to be considered equal. The (==) allows for type conversion. If two values are not the same type, (==) and (===) may return different answers.

13. Describe a hazard of allowing short-circuited evaluation of expressions and side effects in expressions at the same time.

Short-circuted evaluation allows for an expression to be evaluated without evaluating all of the operands and/or operators. If side effects occur, the type of a parameter or global variable may be changed. This could change the value of the answer to a part of the expression, changing the value of the whole expression, creating an error in the program if it short circuits the expression.

14. Briefly describe the three steps in the mark-sweep algorithm for garbage collection.

---

1. Alls the cells in the heap have their indicators set to indicate they are garbage. 2. Every pointer in the program is traced to the heap. All reachable cells are marked as 'not garbage.' 3. The sweep phase is when all the cells in the heap that have not been specifically marked as being used are returned to the list of available space.

15. What led Yukihiro Matsumoto to create the Ruby programming language?

He was dissatisfied with Perl and Python, because neither of them were purely object-oriented languages.

16. What did Microsoft aim to achieve with its development of the C# language?

They wanted to create a language for development in the .NET Framework, and other component-based software.They aimed to improve C++ and Java.

8.2 More questions for discussion and review.

1. The design of which machine influenced the design of the control statements in FORTRAN?

2. How many different kinds of control statements must the designer of a programming language include in a language?

3. What is the one question that applies in the design of all statements that allow selection or iteration?

4. What is an advantage of requiring that the **then** and **else** clauses of an **if** statement be compound statements?

5. How does the **switch** statement in C# differ from the **switch** statement in Java?

6. Distinguish between 2 statements in Ruby that correspond to Java's **switch** statement.

7. Features of a programming language sometimes persist longer than a feature of computing hardware that inspired and supported that part of the language's design. Similarly, features of hardware sometimes persist longer than some parts of a language's design that were created to take advantage of that feature in hardware.

   Give examples.

8. Who most famously warned of the dangers of using the **goto** statement? What did Donald Knuth have to say about the use of the **goto** statement?

9. Describes Ada's **for** loop. Are there some kinds of iteration that might be easier in Ada than in Java? Easier in Java than in Ada?

10. What does it mean to say that the guarded commands of Ada are nondeterministic?

11. The header files in a C program contain function prototypes. What is a function prototype?

12. Every method in a Ruby program belongs to a class. A programmer can place a definition of a method inside the definition of a class or outside of the definition of any class that the programmer writes. To which class does the method belong in the second case?

13. Distinguish between positional and keyword parameters.

14. Ruby blocks are closures. What does that mean?

15. What is a pure function?

16. Some languages give programmers means to define both functions and procedures. Java doe not. Is that a serious limitation?

17. Declarations of formal parameters in an Ada procedure can include, in addition to the names and types of the parameters, reserved words that do not appear in declarations in Java programs. What is the purpose of those reserved words?

18. The C language imposes a constraint upon programmers who want to pass a multidimensional array to a function. What is the constraint? How did the design of the Java programming language eliminate that constraint for programmers who use that language?

19. An activation record contains a return address, a dynamic link, parameters, and local variables.

    (a) To what does the return address point?

    (b) To what does the dynamic link point?

20. The stack will contain multiple activation records for a single subprogram under what circumstances?

21. How (or why?) does the LIFO protocol apply to calls to and returns from subprograms?

22. Which important development in computer architecture has changed the way that the stack is used in some systems for facilitating calls to and returns from subprograms?

23. A dynamic chain contains a history of what?

24. Which two numbers are needed to compute the address of a local variable in a subprogram?

25. How does a Ruby module differ from a class?

26. Memory for variables can be allocated on the heap and on the stack. In which place or places is memory allocated for objects in C++? in Java?

27. What problems were solved by the addition of genericity to Java?

28. What is the purpose of the static chain?

29. What is a singleton?

30. What are the two parts of the definition of an abstract data type?

---

1. IBM 704 system

2. While it is possible to use only one (GOTO), the minimum number of control statements in a language which does not use GOTO is two. One for choosing between control flow paths, and one for logically controlled iterations. (348)

3. Should the control structure have multiple entries? (349)

4. Requiring compound statements helps increase the readability and writability for programmers when using nested selector statements, that otherwise can get very messy and complicated.

5. In Java, the switch statement does not allow case expressions anywhere except the top level in the body of the switch. C# allows the execution of more than one segment. In C#, the control expressions as well as case statements can be strings. (357)

6. Case expressions are the Ruby equivalent of Java's **switch** statement. One is semantically similar to nested if statements with case - when - then. The other is with boolean expressions being evaluated one at a time from top to bottom. The value of this case expression is equivalent to the value of the **then** statement corresponding to the first *true* **when** statement. (357-358)

7. The IBM 704 influenced the design of control statements which are still used today, and prompted the development of Fortran. (348, 42) Most of the popular languages of the past 50 years have been designed around the von Neumann architecture. These are called imperative languages. In a von Neumann computer, both data and programs are stored in the same memory. The CPU, which executes instructions, is separate from the memory. Therefore, instructions and data must be transmitted, or piped, from memory to the CPU. Results of operations in the CPU must be moved back to memory. Nearly all digital computers built since the 1940s have been based on the von Neumann architecture. The Algol standard used several different syntaxes which, among other things, allowed Europeans to use a comma to denote a decimal point, while Americans could continue to use a period. The **register** keyword in C is a hint to the compiler that a variable will be used repeatedly, and so it should be stored in the CPU rather than in memory. However, modern compilers are far better at optimization than programmers, so this keyword is outdated and unnecessary.

8. Edsger Dijkstra noted The goto statement as it stands is just too primitive; it is too much an invitation to make a mess of ones program. Donald Knuth argued there were occasions when the efficiency of the goto outweighed its harm to readability. (376)

9. The for loop looks like this

   ```
   for variable in [reverse] discrete_range loop    end loop;
   ```

   Adas for loop can use any ordinal type variable for its counter. Arrays with ordinal type subscripts can be conveniently processed. (261) For loops in Java are more flexible - can have infinite loops, change loop variable inside body, etc. but is potentially more confusing to read.

10. Guarded commands in Ada are nondeterministically chosen for execution when more than one of the statements are evaluated to true. This means that if there are three guarded statements and two of the three evaluate to true, then each time the program will use one of the two statements. It will not always use the one that appears first nor the one that appears last, but rather it will choose between them non-deterministically or randomly.

11. A function prototype is a function declaration that gives the function's name and type signature, but does not specify the function body. It is also referred to as a function interface at times. In other languages these are uncommon because subprograms do not need declarations since they do no need to be defined before they are called.

12. If a method is defined outside of the definition of any class that the programmer writes then the method belongs to the root object, **Object**. ¡–This is latex code to make it bold

13. Positional parameters are bound based on the order in which they are given to the function. Keyword parameters are when the name of the formal parameter to which an actual parameter is to be bound is specified with the actual parameter in a call. This allows them to be input in any order.

14. A closure is an anonymous function which can be passed as a parameter. Also, Ruby doesn't nest scope, so variables defined in a method are not accessible outside of the method (say, in the class which defines them). A **pure function** is a **function** where the return value is only determined by its input values, without observable side effects.

15. It does not seem to be a serious limitation, it just means that performing procedures in Java is more verbose because it requires that the programmer instantiate the class with any needed methods.

16. Ada allows the programmer to specify *in* mode, *out* mode, and *inout* mode for each formal parameter. This means that they can receive data from the actual parameter, they can send data to the actual parameter, or they can do both.

17. In C, programmers are required to pass the length of an array to a method. Specifying the length of an array (like in C) is unnecessary because Java arrays store a pointer to an address containing their length.

18. The return address usually consists of a pointer to the instruction following the call in the code segment of the calling program unit. The dynamic link points to the base of the activation record instance of the caller.

19. In a recursive program there can be multiple activation records (although they will be incomplete) for a single subprogram.

20. LIFO allows for subprograms to be nested within one another and for subprograms to be used as parameters within other subprograms. (417)

21. RISC (reduced instruction set computing) machines have parameters passed in registers in their compilers because RISC machines have more registers than CISC (complex instruction set computing) machines. Chapter 10 assumes parameters are passed in the stack though, as they had been in CISC machines.

22. A dynamic chain represents the dynamic history of how the execution got to its current position, which is always in the subprogram code whose activation record instance is on top of the stack.

23. To compute the address of a local variable you need the (chain_offset, local_offset) pair.

24. Modules are unlike classes in that they cannot be instantiated or subclassed and do not define variables. Methods that are defined in a module include the modules name in their names.

25. In C++, variables can be allocated to the heap either by making them static or by allocating memory with the keyword new. However, variables which are initialized during the execution of a function are allocated to the stack. Java behaves the same way. (216-217)

26. Generics allow a type or method to operate on objects of various types while providing compile-time type safety.(From wikipedia)

27. The static chain is a path of pointers which go from each function to its parent. They allow child subprograms to use variables which are local to their parent, grandparent, or farther up, without needing to copy those variables to the call stack.

28. A singleton is a class which provides a global access point to a single instance. This is useful for tasks which only need one point of access, like a file system.

29. 1. A type definition which allows program units to declare variables of the type but hides the representation of objects of the type. 2. A set of operations for manipulating objects of the type.

Chapter 9

Aeint Thet Ngon

1. Most programming languages require the use of brackets to enclose the index in a reference to an element of an array.

   (a) Identify a language the requires the use of parentheses to enclose the index in a reference to an element of an array.

   (b) Why did the designers of the language choose parentheses rather than brackets?

   (a) Ada

   (b) Because the designers wanted uniformity between array references and function calls in expressions in spite of potential readability problems.

2. What is the relationship between a lexeme and a token?

   A lexeme is a formal descriptions of a syntax of progamming languages and most of the times do not include descriptions of the lowest-level syntactic units. Each lexeme group is represented by a name or a token. So a token of a language is a category of its lexemes.

3. (a) What kind of symbols are found at the internal nodes of a parse tree?

   (b) What kind of symbols are found at the leaves of a parse tree?

(a) Non-terminal cateogires of the grammar are found at the internal nodes of a parse tree.

(b) Leaf nodes are labelled by terminal categories.

4. One of the most significant contributions from the developers of ALGOL 60 also limited the success of that language. What was that contribution?

BNF, one of the most important contributions to computer science, is considered a simple and elegant means of syntax description but in 1960 it seemed strange and complicated and was a factor in its lack of acceptance.

5. What problem were the creators of Common LISP trying to solve?

During the 1970s and early 198s, due to the usage of diverse dialects of LISP, there was a problem of lack of portability among the programs writting using different dialects. To solve the problem, Common Lisp was created by combining the features of different dialects of LISP.

6. What is an ambiguous context free grammar?

A grammar that genereates a sentential form for which there are two or more distinct parse trees is said to be ambiguous.

7. Contrast the complexity of algorithms that can parse strings that conform to the most general kinds of context free grammars and the complexity of the algorithms that can parse strings that conform to the grammars of programming languages?

Parsing algorithms that work for any unambiguous grammar are complicated and inefficient, the amount of time they take is on the order of the cube of the lenght of the string to be parsed. So generality is traded for efficiency. Faster alogrithms have been found that work only for a subset of the set of all possible grammars and the time they take is linearly related to the length of the string to be parsed.

8. Java represents characters with Unicode. It is the first widely used programming language with this feature. What is the significance of this feature?

   This feature includes the characters from most of the world's natural languages. It was developed because of globalization of business and the need for computers to communicate with other computers around the world.

9. How does the binary coded decimal type differ from the floating point type?

   Floating point data types model real numbers but the representation are only apprximations for many real values. Decimal types have the advantage of being able to precisely store decimal values, which cannot be done with floating point.

10. Identify a user-defined ordinal type in the Java programming language.

    Enumeration Types

11. Mathematicians and programmers might have different ideas about the precedence of Boolean operators. Explain.

    In mathematics, Boolean algebras have equal precedence, however, C-based languages assign higher precedence to AND than OR. This might have resulted from the baseless correlation of mulitplcation with AND and of additon with OR, which would then naturally assign higher precedence to AND.

12. Programmers should use `===` rather than `==` to test the equality of the values of two expressions in JavaScript. Why?

    Because whena string and a number are the operands of a relational operator, the string is coerced to a number but when === is used no coercion is done on the operands of this operator.

13. Describe a hazard of allowing short-circuited evaluation of expressions and side effects in expressions at the same time.

A language that provides short-circuit evaluations of Boolean expressions and also has side effects in expressions allows subltel errors to occr. Suppose that short-circuit evaluation is used on an expression and part of the expression that contains a side effect is not evaluated; then the side effect will occur only in complete evaluations of the whole expression. If a program correctness depends on the side effect, short-circuit evaluation can result in a serious error.

14. Briefly describe the three steps in the mark-sweep algorithm for garbage collection.

- All cells in the heap have their indicators set to indicate they are garbage
- Marking phase-Every pointer in the program is traced into the heap and all reachable cells are marked as not being garbage
- Sweep phase-all cells in the heap that have not been specifially marked as still being used are returned to the list of available space

15. What led Yukihiro Matsumoto to create the Ruby programming language?

Yukihiro Matsumoto's motivation was due to his dissatisfaction with Perl and Python, which support object-oriented programming but neither is puer object oriented langauge.

16. What did Microsoft aim to achieve with its development of the C# language?

The purose of C# is to provide a language for componenet based software development. In this environment, components from a variety of languages can be easily combined to form systems.

9.1 More questions for discussion and review.

1. The design of which machine influenced the design of the control statements in FORTRAN?

   IBM 704 influenced the design of the control statements in FORTRAN.

2. How many different kinds of control statements must the designer of a programming language include in a language?

   One of the primary conclusions of these efforts was that, although a single control statement (a selectable goto) is minimally sufficient, a language that is designed not to include a goto needs only a small number of different control statements.

3. What is the one question that applies in the design of all statements that allow selection or iteration?

   Should multiple entries be permitted?

4. What is an advantage of requiring that the **then** and **else** clauses of an **if** statement be compound statements?

   Requiring compound statements helps increase the readability and writability for programmers when using nested selector statements, that otherwise can get very messy and complicated.

5. How does the **switch** statement in C# differ from the **switch** statement in Java?

   The C# switch statement differs from that of its C-based predecessors in two ways. First, C# has a static semantics rule that disallows the implicit execution of more than one segment. The rule is that every selectable segment must end with an explicit unconditional branch statement: either a break, which transfers control out of the switch statement, or a goto, which can transfer control to one of the selectable segments (or virtually anywhere else). The other way C#s switch differs from that of its predecessors is that the control expression and the case statements can be strings in C#.

6. Distinguish between 2 statements in Ruby that correspond to Java's **switch** statement.

Ruby's case expression is similar to the switch statement. And **when** in Ruby's case expression is used to evaluate as **if** does in Java's switch statement and **then** statement is executed.

7. Features of a programming language sometimes persist longer than a feature of computing hardware that inspired and supported that part of the language's design. Similarly, features of hardware sometimes persist longer than some parts of a language's design that were created to take advantage of that feature in hardware.

   Give examples.

The IBM 704 influenced the design of control statements

8. Who most famously warned of the dangers of using the **goto** statement? What did Donald Knuth have to say about the use of the **goto** statement?

Donald Knuth argues that well thought out, rational, controlled use of goto is not entirely harmful but uncontrolled and thoughtless use of GoTo is probably a bad thing.

9. Describes Ada's **for** loop. Are there some kinds of iteration that might be easier in Ada than in Java? Easier in Java than in Ada?

Adas for loop can use any ordinal type variable for its counter. Arrays with ordinal type subscripts can be conveniently processed

10. What does it mean to say that the guarded commands of Ada are nondeterministic?

A guarded command is a statement, or list of statements, that is "guarded" by a boolean expression.The semantics is that all the expressions are evaluated simultaneously, if one of the expressions is true, then the statement associated with it is performed. If more that one is true, then one statement is picked nondeterministically. Finally, if all of the expressions evaluate to false, the program exits the loop.

11. The header files in a C program contain function prototypes. What is a function prototype?

A function prototype or function interface is a declaration of a function that specifies the function's name and type signature (arity, parameter types, and return type), but omits the function body.

12. Every method in a Ruby program belongs to a class. A programmer can place a definition of a method inside the definition of a class or outside of the definition of any class that the programmer writes. To which class does the method belong in the second case?

If a method is defined outside of the definition of any class that the programmer writes then the method belongs to the root object, **Object**

13. Distinguish between positional and keyword parameters.

Positional parameters are bound based on the order in which they are given

to the function. Keyword parameters are when the name of the formal parameter to which an actual parameter is to be bound is specified with the actual parameter in a call. This allows them to be input in any order

14. Ruby blocks are closures. What does that mean?

A closure is an anonymous function which can be passed as a parameter. Also, Ruby doesn't nest scope, so variables defined in a class are not accessible in that class's methods.

15. What is a pure function?

A pure function is a function where the return value is only determined by its input values, without observable side effects. This is how functions in math work: Math.cos(x) will, for the same value of x , always return the same result.

16. Some languages give programmers means to define both functions and procedures. Java doe not. Is that a serious limitation?

It does not seem to be a serious limitation, it just means that performing procedures in Java is more verbose because it requires that the programmer instantiate the class with any needed methods.

17. Declarations of formal parameters in an Ada procedure can include, in addition to the names and types of the parameters, reserved words that do not appear in declarations in Java programs. What is the purpose of those reserved words?

Ada and Fortran 95+ allow the programmer to specify in mode, out mode, or inout mode on each formal parameter.A reserved word is a special word of a programming language that cannot be used as a name. As a language design choice, reserved words are better than keywords because the ability to redefine keywords can be confusing.

18. The C language imposes a constraint upon programmers who want to pass a multidimensional array to a function. What is the constraint? How did the design of the Java programming language eliminate that constraint for programmers who use that language?

In C, programmers are required to pass the length of an array to a method. Specifying the length of an array (like in C) is unnecessary because Java arrays store a pointer to an address containing their length.

19. An activation record contains a return address, a dynamic link, parameters, and local variables.

    (a) To what does the return address point?

    (b) To what does the dynamic link point?

The return address usually consists of a pointer to the instruction following the call in the code segment of the calling program unit. The dynamic link points to the base of the activation record instance of the caller.

20. The stack will contain multiple activation records for a single subprogram under what circumstances?

In a recursive program there can be multiple activation records (although they will be incomplete) for a single subprogram.

21. How (or why?) does the LIFO protocol apply to calls to and returns from subprograms?

LIFO allows for subprograms to be nested within one another and for subprograms to be used as parameters within other subprograms.

22. Which important development in computer architecture has changed the way that the stack is used in some systems for facilitating calls to and returns from subprograms?

RISC (reduced instruction set computing) machines have parameters passed

in registers in their compilers because RISC machines have more registers than CISC (complex instruction set computing) machines. Chapter 10 assumes parameters are passed in the stack though, as they had been in CISC machines.

23. A dynamic chain contains a history of what?

Dynamic chain represents the dynamic history of how execution got to its current position, which is always in the subprogram code whose activation record instance is on top of the stack.

24. Which two numbers are needed to compute the address of a local variable in a subprogram?

To compute the address of a local variable you need the (chain_offset, local_offset) pair.

25. How does a Ruby module differ from a class?

Modules are unlike classes in that they cannot be instantiated or subclassed and do not define variables. Methods that are defined in a module include the modules name in their names

26. Memory for variables can be allocated on the heap and on the stack. In which place or places is memory allocated for objects in C++? in Java?

In C++, variables can be explicitly fixed in the heap by using the keyword static. They can also be also be dynamically allocated to the heap with the new keyword.

27. What problems were solved by the addition of genericity to Java?

Generics allow a type or method to operate on objects of various types while providing compile-time type safety.

28. What is the purpose of the static chain?

Static chain points to the bottom of the activation record instance of an activation of the static parent. It is used for accesses to nonlocal variables.

29. What is a singleton?

A singleton is a class which provides a global access point to a single instance. This is useful for tasks which only need one point of access, like a file system.

30. What are the two parts of the definition of an abstract data type?

(a) A type definition which allows program units to declare variables of the type but hides the representation of objects of the type

(b) A set of operations for manipulating objects of the type.

Chapter 10

Huong Nguyen

1. Most programming languages require the use of brackets to enclose the index in a reference to an element of an array.
   (a) Identify a language the requires the use of parentheses to enclose the index in a reference to an element of an array.
   (b) Why did the designers of the language choose parentheses rather than brackets?

   (a) The languages are pre-90 Fortrans and PL/I
   (b) Because there were no other suitable characters available at the time. Card punches did not include bracket characters.

2. What is the relationship between a lexeme and a token?

   Lexemes are the lowest-level syntactic units. A token of a language is a category of its lexemes.

3. (a) What kind of symbols are found at the internal nodes of a parse tree?
   (b) What kind of symbols are found at the leaves of a parse tree?

   (a) Nonterminal symbols are found at the internal nodes of a parse trees.

(b) Terminal symbols are found at the leaves of a parse trees.

4. One of the most significant contributions from the developers of ALGOL 60 also limited the success of that language. What was that contribution?

It was the use of BNF as the formal means of describing syntax. In 1960 it seemed strange and complicated.

5. What problem were the creators of Common LISP trying to solve?

During the 1970s and early 1980s, a large number of different dialects of LISP were developed and used. This led to the familiar problem of lack of portability among programs written in the various dialects. Common LIPS was created to rectify the situation by combining the features of several dialects of LISP, including Scheme, into a single language.

6. What is an ambiguous context free grammar?

An ambiguous grammar is a grammar that generates a sentential form for which there are two or more distinct parse trees. Context-free grammars were developed by Noam Chomsky as a class of natural languages, but became the primary method through which the syntax of programming languages are described.

7. Contrast the complexity of algorithms that can parse strings that conform to the most general kinds of context free grammars and the complexity of the algorithms that can parse strings that conform to the grammars of programming languages?

Algorithms that can parse general kinds of context-free grammars have a cubic time complexity. But the grammars of programming languages are far less general and only a subset of general context-free grammars. Algorithms to parse them have O(n), or linear time complexity, which is more efficient.

8. Java represents characters with Unicode. It is the first widely used programming language with this feature. What is the significance of this feature?

   Unicode includes the characters from most of the world's natural languages. Before Unicode was created, the 8-bit code ASCII was used to code character data and became inadequate for communication with computers around the world.

9. How does the binary coded decimal type differ from the floating point type?

   Floating-points store only approximations for many real values, while decimals store precise decimal values up to a specific range. Floating-point values are represented as fractions and exponents, typically taking 32 bits. Binary-coded decimals store one to two digits per byte, since a digit takes at least 4 bits, so they take up a lot of storage.

10. Identify a user-defined ordinal type in the Java programming language.

    The java user-defined ordinal type is Enum. All enumeration types in Java are implicitly subclasses of the predefined class Enum.

11. Mathematicians and programmers might have different ideas about the precedence of Boolean operators. Explain.

    In the mathematics of Boolean algebras, the OR and AND operators must have equal precedence. However, the C-base languages assign a higher precedence to AND than OR.

12. Programmers should use `===` rather than `==` to test the equality of the values of two expressions in JavaScript. Why?

=== prevents Javascript operands from being coerced, and in effect checks for type equivalence. For example, "7" == 7 evaluates to true in Javascript, while "7" === 7 evaluates to false.

13. Describe a hazard of allowing short-circuited evaluation of expressions and side effects in expressions at the same time.

Allowing both at the same time may lead to errors where the side effect is part of the second boolean expression, and is not evaluated due to the expression having been short-circuited after the first.

14. Briefly describe the three steps in the mark-sweep algorithm for garbage collection.

First, all cells in the heap are marked as garbage. Second, every pointer is traced into the heap, and mark all cells they reach as not-garbage. Finally, all cells still marked as garbage are returned to the list of available space.

15. What led Yukihiro Matsumoto to create the Ruby programming language?

Matsumoto was dissatisfied with Perl and Python, which were not pure object-oriented languages.

16. What did Microsoft aim to achieve with its development of the C# language?

As the flagship language of .NET and Microsoft, C# was meant to allow .NET to combine components written in all other .NET languages, such as Visual Basic and C++.

10.1 More questions for discussion and review.

1. The design of which machine influenced the design of the control statements in FORTRAN?

   (a) The IBM 704 system.

2. How many different kinds of control statements must the designer of a programming language include in a language?

   (a) Only GOTO is necessary but if you don't use GOTO 2 are necessary, 1 for branching and 1 for iteration.

3. What is the one question that applies in the design of all statements that allow selection or iteration?

   (a) Should the control structure have multiple entries?

4. What is an advantage of requiring that the **then** and **else** clauses of an **if** statement be compound statements?

   (a) The advantage is for writability and readability of nested statements.

5. How does the **switch** statement in C# differ from the **switch** statement in Java?

   (a) C# enforces executing only the first statement.

6. Distinguish between 2 statements in Ruby that correspond to Java's **switch** statement.

   (a) One is semantically similar to nested if statements with case - when - then. The other is with boolean expressions being evaluated one at a time from top to bottom.

7. Features of a programming language sometimes persist longer than a feature of computing hardware that inspired and supported that part of the language's design. Similarly, features of hardware sometimes persist longer than some parts of a language's design that were created to take advantage of that feature in hardware.

   Give examples.

   (a) The IBM 704 influenced the design of control statements which are still used today.

8. Who most famously warned of the dangers of using the **goto** statement? What did Donald Knuth have to say about the use of the **goto** statement?

   (a) Dijkstra

9. Describes Ada's **for** loop. Are there some kinds of iteration that might be easier in Ada than in Java? Easier in Java than in Ada?

   (a) Ada's loop is:

```
for variable in [reverse] discrete_range loop        end loop;
```

   (b) Arrays with ordinal type subscripts are easier to process in Ada. Loops with changes inside are easier in Java.

10. What does it mean to say that the guarded commands of Ada are nondeterministic?

    (a) Guarded commands in Ada are nondeterministically chosen for execution when more than one of the statements are evaluated to true. This means that if there are three guarded statements and two of the three evaluate to true, then each time the program will use one of the two statements. It will not always use the one that appears first nor the one that appears last, but rather it will choose between them non-deterministically or randomly.

11. The header files in a C program contain function prototypes. What is a function prototype?

    (a) It's a function declaration that gives the function's name and type signature, but does not specify the function body.

12. Every method in a Ruby program belongs to a class. A programmer can place a definition of a method inside the definition of a class or outside of the definition of any class that the programmer writes. To which class does the method belong in the second case?

    (a) If a method is defined outside of the definition of any class that the programmer writes then the method belongs to the root object

13. Distinguish between positional and keyword parameters.

    (a) Positional parameters are bound based on the order in which they are given to the function. Keyword parameters are when the name of the formal parameter to which an actual parameter is to be bound is specified with the actual parameter in a call. This allows them to be input in any order.

14. Ruby blocks are closures. What does that mean?

    (a) A closure is an anonymous function which can be passed as a parameter.

15. What is a pure function?

    (a) A function where the return value is only determined by its input values, without observable side effects.

16. Some languages give programmers means to define both functions and procedures. Java doe not. Is that a serious limitation?

    (a) No, it just means that performing procedures in Java is more verbose because it requires that the programmer instantiate the class with any needed methods.

17. Declarations of formal parameters in an Ada procedure can include, in addition to the names and types of the parameters, reserved words that do not appear in declarations in Java programs. What is the purpose of those reserved words?

    (a) Ada allows the programmer to specify *in* mode, *out* mode, and *inout* mode for each formal parameter. This means that they can receive data from the actual parameter, they can send data to the actual parameter, or they can do both.

18. The C language imposes a constraint upon programmers who want to pass a multidimensional array to a function. What is the constraint? How did

the design of the Java programming language eliminate that constraint for programmers who use that language?

(a) In C, programmers are required to pass the length of an array to a method. Specifying the length of an array (like in C) is unnecessary because Java arrays store a pointer to an address containing their length.

19. An activation record contains a return address, a dynamic link, parameters, and local variables.

    (a) To what does the return address point?

    (b) To what does the dynamic link point?

(a) The return address usually consists of a pointer to the instruction following the call in the code segment of the calling program unit.

(b) The dynamic link points to the base of the activation record instance of the caller.

20. The stack will contain multiple activation records for a single subprogram under what circumstances?

(a) If the program is recursive.

21. How (or why?) does the LIFO protocol apply to calls to and returns from subprograms?

(a) LIFO allows for subprograms to be nested within one another and for subprograms to be used as parameters within other subprograms.

22. Which important development in computer architecture has changed the way that the stack is used in some systems for facilitating calls to and returns from subprograms?

   (a) RISC (reduced instruction set computing) machines have parameters passed in registers in their compilers because RISC machines have more registers than CISC (complex instruction set computing) machines.

23. A dynamic chain contains a history of what?

   (a) It's the the dynamic history of how the execution got to its current position.

24. Which two numbers are needed to compute the address of a local variable in a subprogram?

   (a) chain offset and local offset.

25. How does a Ruby module differ from a class?

   (a) Ruby models cannot be instantiated or subclassed and do not define variables. Also, methods that are defined in a module include the modules name in their names.

26. Memory for variables can be allocated on the heap and on the stack. In which place or places is memory allocated for objects in C++? in Java?

(a) In both C++ and Java, variables can be allocated to the heap either by making them static or by allocating memory with the keyword new. However, variables which are initialized during the execution of a function are allocated to the stack.

27. What problems were solved by the addition of genericity to Java?

(a) A type or method can operate on objects of various types with compile-time type safety.

28. What is the purpose of the static chain?

(a) The static chain is a path of pointers which go from each function to its parent. They allow child subprograms to use variables which are local to their parent, grandparent, or farther up, without needing to copy those variables to the call stack.

29. What is a singleton?

(a) A class which provides a global access point to a single instance.

30. What are the two parts of the definition of an abstract data type?

(a) A type definition which allows program units to declare variables of the type but hides the representation of objects of the type, and a set of operations for manipulating objects of the type.

10.2 More questions for discussion and review.

1. The design of which machine influenced the design of the control statements in FORTRAN?

2. How many different kinds of control statements must the designer of a programming language include in a language?

3. What is the one question that applies in the design of all statements that allow selection or iteration?

4. What is an advantage of requiring that the **then** and **else** clauses of an **if** statement be compound statements?

5. How does the **switch** statement in C# differ from the **switch** statement in Java?

6. Distinguish between 2 statements in Ruby that correspond to Java's **switch** statement.

7. Features of a programming language sometimes persist longer than a feature of computing hardware that inspired and supported that part of the language's design. Similarly, features of hardware sometimes persist longer than some parts of a language's design that were created to take advantage of that feature in hardware.

   Give examples.

8. Who most famously warned of the dangers of using the **goto** statement? What did Donald Knuth have to say about the use of the **goto** statement?

9. Describes Ada's **for** loop. Are there some kinds of iteration that might be easier in Ada than in Java? Easier in Java than in Ada?

10. What does it mean to say that the guarded commands of Ada are nondeterministic?

11. The header files in a C program contain function prototypes. What is a function prototype?

12. Every method in a Ruby program belongs to a class. A programmer can place a definition of a method inside the definition of a class or outside of the definition of any class that the programmer writes. To which class does the method belong in the second case?

13. Distinguish between positional and keyword parameters.

14. Ruby blocks are closures. What does that mean?

15. What is a pure function?

16. Some languages give programmers means to define both functions and procedures. Java doe not. Is that a serious limitation?

17. Declarations of formal parameters in an Ada procedure can include, in addition to the names and types of the parameters, reserved words that do not appear in declarations in Java programs. What is the purpose of those reserved words?

18. The C language imposes a constraint upon programmers who want to pass a multidimensional array to a function. What is the constraint? How did the design of the Java programming language eliminate that constraint for programmers who use that language?

19. An activation record contains a return address, a dynamic link, parameters, and local variables.

    (a) To what does the return address point?

    (b) To what does the dynamic link point?

20. The stack will contain multiple activation records for a single subprogram under what circumstances?

21. How (or why?) does the LIFO protocol apply to calls to and returns from subprograms?

22. Which important development in computer architecture has changed the way that the stack is used in some systems for facilitating calls to and returns from subprograms?

23. A dynamic chain contains a history of what?

24. Which two numbers are needed to compute the address of a local variable in a subprogram?

25. How does a Ruby module differ from a class?

26. Memory for variables can be allocated on the heap and on the stack. In which place or places is memory allocated for objects in C++? in Java?

27. What problems were solved by the addition of genericity to Java?

28. What is the purpose of the static chain?

29. What is a singleton?

30. What are the two parts of the definition of an abstract data type?

Chapter 11

Spencer Rudnick

1. Most programming languages require the use of brackets to enclose the index in a reference to an element of an array.

    (a) Identify a language the requires the use of parentheses to enclose the index in a reference to an element of an array.

    (b) Why did the designers of the language choose parentheses rather than brackets?

(a) Ada is an example of a programming language that uses parentheses to enclose indices to reference elements of arrays.

(b) The designers of Ada chose to use parentheses for enclosing indices, in spite of the fact that it makes code more difficult to read, because both array references and function calls map to an address in memory. An understanding of how a program runs on the hardware of computer reveals that this seemingly strange choice actually makes a lot of sense.

2. What is the relationship between a lexeme and a token?

A token describes a category of lexemes. For example, all variable names, *i, index, currentNode, etc.*, are *lexemes* within the token *Identifiers.* =̈ḧas its own token, *equal_sign.* And so on, all down the line of reserved words, characters, and user-defined variables.

3. (a) What kind of symbols are found at the internal nodes of a parse tree?
   (b) What kind of symbols are found at the leaves of a parse tree?

   (a) Nonterminal symbols are found at the internal nodes of a parse tree. Nonterminals include *if()* statements and other function calls.
   (b) Terminal symbols are found at the leaves of a parse tree. Terminals include lexemes and tokens, such as integer literals and variable names.

4. One of the most significant contributions from the developers of ALGOL 60 also limited the success of that language. What was that contribution?

ALGOL 60 was the first language to be defined using what is now known as Backus-Naur Form, or BNF for short. BNF was a very significant contribution to the field of computer science. However, it did not catch on very quickly at the time of ALGOL 60's release, so few people adopted the language.

5. What problem were the creators of Common LISP trying to solve?

The creators of Common LISP were trying to solve the problem of fragmentation in the LISP family, which made code portability very difficult. Common LISP combined elements of many other versions of LISP, creating a language which was portable, yet very complex.

6. What is an ambiguous context free grammar?

A grammar for which sentences can have more than one valid parse tree.

7. Contrast the complexity of algorithms that can parse strings that conform to the most general kinds of context free grammars and the complexity

of the algorithms that can parse strings that conform to the grammars of programming languages?

Algorithms which parse strings conforming to general kinds of context free grammars function on $O(n)$.

Algorithms which parse strings conforming to the grammars of a modern programming language usually function on the order of $O(n^3)$.

8. Java represents characters with Unicode. It is the first widely used programming language with this feature. What is the significance of this feature?

Unicode represents characters from almost every natural language and number system. This allows information encoded in Unicode to be read on almost any machine (especially those running Java).

9. How does the binary coded decimal type differ from the floating point type?

Decimals are precise definitions of a value, though they are usually only allowed to be in a range specified by the language and hardware. They contain a fixed decimal point with exact numbers defined on both sides.

Floating points are approximations of a value. They are not precise, but they are able to represent a vastly greater span of values. Floating points are made up of a sign bit, an exponent byte (or two), and between three and eight bytes defining a fraction.

10. Identify a user-defined ordinal type in the Java programming language.

---

integer is a user-defined ordinal type in Java.

11. Mathematicians and programmers might have different ideas about the precedence of Boolean operators. Explain.

In boolean algebra, the AND and OR operators have the same precedence. However, C-based languages assign a higher precedence to AND than OR, likely because of an erroneous associations between arithmetic multiplication with the AND function, and arithmetic addition with the OR function.

12. Programmers should use `===` rather than `==` to test the equality of the values of two expressions in JavaScript. Why?

In JavaScript, `"=="` allows the operands to be coerced (one is converted into the same type as the other for easy comparison). `"7" == 7` would return true. `"==="` prevents coercion by the interpreter, so `"7" === 7` would return false.

13. Describe a hazard of allowing short-circuited evaluation of expressions and side effects in expressions at the same time.

There is a possibility that the whole expression may not be parsed, and so the side effect which was intended to happen is skipped. This could result in small errors in which an integer which was supposed to be incremented is not! (gasp!)

14. Briefly describe the three steps in the mark-sweep algorithm for garbage collection.

Step 1. All cells in the heap are marked as garbage.

Step 2. Each cell is checked to see if it is reachable. If it is, it is marked as still in use (not garbage).

Step 3. All cells still marked as garbage are destroyed (removed from the heap).

15. What led Yukihiro Matsumoto to create the Ruby programming language?

Yukihiro Matsumoto created Ruby out of dissatisfaction with Perl and Python, specifically that they were not pure object-oriented languages.

16. What did Microsoft aim to achieve with its development of the C# language?

The purpose of C# was to create a language which can combine components in any other language within the .NET framework (C#, Visual Basic, .Net, Managed C++, F#, and JScript .Net). Seems like they were trying to incorporate some of the Unix philosophy we learned about on the first day!

11.1 More questions for discussion and review.

1. The design of which machine influenced the design of the control statements in FORTRAN?

The IBM 704 system.

2. How many different kinds of control statements must the designer of a programming language include in a language?

While it is possible to use only one (GOTO), the minimum number of control statements in a language which does not use GOTO is two. One for choosing between control flow paths, and one for logically controlled iterations.

3. What is the one question that applies in the design of all statements that allow selection or iteration?

Should the control structure have multiple entries?

4. What is an advantage of requiring that the **then** and **else** clauses of an **if** statement be compound statements?

Requiring compound statements helps increase the readability and writability for programmers when using nested selector statements, that otherwise can get very messy and complicated.

5. How does the **switch** statement in C# differ from the **switch** statement in Java?

In Java, the switch statement does not allow case expressions anywhere except the top level in the body of the switch. C# allows the execution of more than one segment. In C#, the control expressions as well as case statements can be strings.

6. Distinguish between 2 statements in Ruby that correspond to Java's **switch** statement.

Case expressions are the Ruby equivalent of Java's **switch** statement. One is semantically similar to nested if statements with case - when - then. The other is with boolean expressions being evaluated one at a time from top to bottom. The value of this case expression is equivalent to the value of the **then** statement corresponding to the first *true* **when** statement.

7. Features of a programming language sometimes persist longer than a feature of computing hardware that inspired and supported that part of the language's design. Similarly, features of hardware sometimes persist longer than some parts of a language's design that were created to take advantage of that feature in hardware.

   Give examples.

The IBM 704 influenced the design of control statements which are still used today, and prompted the development of Fortran. The **register** keyword in C is a hint to the compiler that a variable will be used repeatedly, and so it should be stored in the CPU rather than in memory. However, modern compilers are far better at optimization than programmers, so this keyword is outdated and unnecessary. The Algol standard used several different syntaxes which, among other things, allowed Europeans to use a comma to denote a decimal point, while Americans could continue to use a period.

8. Who most famously warned of the dangers of using the **goto** statement? What did Donald Knuth have to say about the use of the **goto** statement?

Edsger Dijkstra noted The goto statement as it stands is just too primitive; it is too much an invitation to make a mess of ones program. Donald Knuth argued there were occasions when the efficiency of the goto outweighed its harm to readability.

9. Describes Ada's **for** loop. Are there some kinds of iteration that might be easier in Ada than in Java? Easier in Java than in Ada?

The for loop looks like this for variable in [reverse] discrete_range loop end loop; Adas for loop can use any ordinal type variable for its counter. Arrays with ordinal type subscripts can be conveniently processed.

10. What does it mean to say that the guarded commands of Ada are nondeterministic?

Guarded commands in Ada are nondeterministically chosen for execution when more than one of the statements are evaluated to true. This means that if there are three guarded statements and two of the three evaluate to true, then each time the program will use one of the two statements. It will not always use the one that appears first nor the one that appears last, but rather it will choose between them non-deterministically or randomly.

11. The header files in a C program contain function prototypes. What is a function prototype?

A function prototype is a function declaration that gives the function's name and type signature, but does not specify the function body. It is also referred to as a function interface at times. In other languages these are uncommon because subprograms do not need declarations since they do no need to be defined before they are called.

12. Every method in a Ruby program belongs to a class. A programmer can place a definition of a method inside the definition of a class or outside of the definition of any class that the programmer writes. To which class does the method belong in the second case?

If a method is defined outside of the definition of any class that the programmer writes then the method belongs to the root object, **Object**.

13. Distinguish between positional and keyword parameters.

Positional parameters are bound based on the order in which they are given to the function. Keyword parameters are when the name of the formal parameter to which an actual parameter is to be bound is specified with the actual parameter in a call. This allows them to be input in any order.

14. Ruby blocks are closures. What does that mean?

A closure is an anonymous function which can be passed as a parameter. Also, Ruby doesn't nest scope, so variables defined in a method are not accessible outside of the method (say, in the class which defines them).

15. What is a pure function?

A **pure function** is a **function** where the return value is only determined by its input values, without observable side effects.

16. Some languages give programmers means to define both functions and procedures. Java does not. Is that a serious limitation?

It does not seem to be a serious limitation, it just means that performing procedures in Java is more verbose because it requires that the programmer instantiate the class with any needed methods.

17. Declarations of formal parameters in an Ada procedure can include, in addition to the names and types of the parameters, reserved words that do not appear in declarations in Java programs. What is the purpose of those reserved words?

Ada allows the programmer to specify *in* mode, *out* mode, and *inout* mode for each formal parameter. This means that they can receive data from the actual parameter, they can send data to the actual parameter, or they can do both.

18. The C language imposes a constraint upon programmers who want to pass a multidimensional array to a function. What is the constraint? How did the design of the Java programming language eliminate that constraint for programmers who use that language?

In C, programmers are required to pass the length of an array to a method. Specifying the length of an array (like in C) is unnecessary because Java arrays store a pointer to an address containing their length.

19. An activation record contains a return address, a dynamic link, parameters, and local variables.

    (a) To what does the return address point?

    (b) To what does the dynamic link point?

The return address usually consists of a pointer to the instruction following the call in the code segment of the calling program unit. The dynamic link points to the base of the activation record instance of the caller.

20. The stack will contain multiple activation records for a single subprogram under what circumstances?

In a recursive program there can be multiple activation records (although they will be incomplete) for a single subprogram.

21. How (or why?) does the LIFO protocol apply to calls to and returns from subprograms?

LIFO allows for subprograms to be nested within one another and for subprograms to be used as parameters within other subprograms.

22. Which important development in computer architecture has changed the way that the stack is used in some systems for facilitating calls to and returns from subprograms?

RISC (reduced instruction set computing) machines have parameters passed in registers in their compilers because RISC machines have more registers than CISC (complex instruction set computing) machines. Chapter 10 assumes parameters are passed in the stack though, as they had been in CISC machines.

23. A dynamic chain contains a history of what?

A dynamic chain represents the dynamic history of how execution got to its current position, which is always in the subprogram code whose activation record instance is on top of the stack.

24. Which two numbers are needed to compute the address of a local variable in a subprogram?

To compute the address of a local variable you need the (chain_offset, local_offset) pair.

25. How does a Ruby module differ from a class?

Modules are unlike classes in that they cannot be instantiated or subclassed and do not define variables. Methods that are defined in a module include the modules name in their names.

26. Memory for variables can be allocated on the heap and on the stack. In which place or places is memory allocated for objects in C++? in Java?

In C++, variables can be allocated to the heap either by making them static or by allocating memory with the keyword new. However, variables which are initialized during the execution of a function are allocated to the stack. Java behaves the same way.

27. What problems were solved by the addition of genericity to Java?

Generics allow a type or method to operate on objects of various types while providing compile-time type safety.

28. What is the purpose of the static chain?

The static chain is a path of pointers which go from each function to its parent. They allow child subprograms to use variables which are local to their parent, grandparent, or farther up, without needing to copy those variables to the call stack.

29. What is a singleton?

A singleton is a class which provides a global access point to a single instance. This is useful for tasks which only need one point of access, like a file system.

30. What are the two parts of the definition of an abstract data type?

(a) A type definition which allows program units to declare variables of the type but hides the representation of objects of the type.

(b) A set of operations for manipulating objects of the type.

Chapter 12

Kat Sayrs

1. Most programming languages require the use of brackets to enclose the index in a reference to an element of an array.

   (a) Identify a language the requires the use of parentheses to enclose the index in a reference to an element of an array.

   (b) Why did the designers of the language choose parentheses rather than brackets?

   (a) Ada

   (b) They chose parenthasees so that there would be uniformity between array refences and function calls in expressions, regardless of potential readability issues.

2. What is the relationship between a lexeme and a token?

   A lexeme is the lowest-level syntactc unit. They are partitioned into groups. Each lexeme group is represented by a token. So, a token is a category of its lexemes.

3. (a) What kind of symbols are found at the internal nodes of a parse tree?

   (b) What kind of symbols are found at the leaves of a parse tree?

(a) nonterminal symbols

(b) terminal symbols

4. One of the most significant contributions from the developers of ALGOL 60 also limited the success of that language. What was that contribution?

Backus-Naur Form, or BNF, a way of describing sntax.

5. What problem were the creators of Common LISP trying to solve?

The lack of portability among programs written in various dialects.

6. What is an ambiguous context free grammar?

A grammar that generates a sentential form for which there are two or more distinct parse trees.

7. Contrast the complexity of algorithms that can parse strings that conform to the most general kinds of context free grammars and the complexity of the algorithms that can parse strings that conform to the grammars of programming languages?

Parsing algorithms that work for any unambiguous grammar are complicated and inefficient. The complexity of such algorithms is O(n3). This relatively large amount of time is required because these algorithms frequently must back up and reparse part of the sentence being analyzed.Backing up the parser also requires that part of the parsemtree being constructed must be dismantled and rebuilt. In terms of parsing, faster algorithms have been found that work for only a subset of the set of all possible grammars. These algorithms are acceptable as long as the subset includes grammars that describe programming languages.

8. Java represents characters with Unicode. It is the first widely used programming language with this feature. What is the significance of this feature?

Unicode includes characters from most of the world's natural languages, allowing for communication among computers worldwide.

9. How does the binary coded decimal type differ from the floating point type?

Decimal types are able to precisely store decimal values within a restricted range, which cannot be don with a floating-point. However, their storage in memory is somewhat wasteful. Floating-points are approximations, but take up less storage space in memory.

10. Identify a user-defined ordinal type in the Java programming language.

An enumeration type.

11. Mathematicians and programmers might have different ideas about the precedence of Boolean operators. Explain.

In mathematics, OR and AND have equal precedence, and this is the case with Ada. However, C-based languages assign AND a higher prescedence than OR, possibly resulting from a correlation between multiplication and the AND operator, and addition and OR.

12. Programmers should use `===` rather than `==` to test the equality of the values of two expressions in JavaScript. Why?

With ==, the expressions are coerced into the same type before evaluation, but with ===, no coercion is done.

13. Describe a hazard of allowing short-circuited evaluation of expressions and side effects in expressions at the same time.

The side effect will only occur in complete evaluations of the whole expression, so if the program depends upon that side effect for correctness and the part of the expression with the side effect is not evaluated, the program will result in a serious error and fail.

14. Briefly describe the three steps in the mark-sweep algorithm for garbage collection.

1. All cells in the heap are indeicated as garbage. 2. Every pointer in the program is traced to the heap, and all reachable cells are marked as not garbage anymore. 3. sweep phase: all cells not marked as still being used are returned to the list of available space.

15. What led Yukihiro Matsumoto to create the Ruby programming language?

He was dissatisfied with Perl and Python, since neither of them are pure object-oriented languages in that they have primitive types and support functions.

16. What did Microsoft aim to achieve with its development of the C# language?

The designers wanted to provide a language for component-based software development, specifically for such development in the .NET framework. In this environment, components from a variety of languages can be easily combined to form systems.

12.1 More questions for discussion and review.

1. The design of which machine influenced the design of the control statements in FORTRAN?

   IBM 704

2. How many different kinds of control statements must the designer of a programming language include in a language?

   While it is possible to use only one (GOTO), the minimum number of control statements in a language which does not use GOTO is two. One for choosing between control flow paths, and one for logically controlled iterations.

3. What is the one question that applies in the design of all statements that allow selection or iteration?

   Should the control structure have multiple entries?

4. What is an advantage of requiring that the **then** and **else** clauses of an **if** statement be compound statements?

   Requiring compound statements helps increase the readability and writability for programmers when using nested selector statements, that otherwise can get very messy and complicated.

5. How does the **switch** statement in C# differ from the **switch** statement in Java?

   In Java, the switch statement does not allow case expressions anywhere except the top level in the body of the switch. C# allows the execution of more than one segment. In C#, the control expressions as well as case statements can be strings. (357)

6. Distinguish between 2 statements in Ruby that correspond to Java's **switch** statement.

Case expressions are the Ruby equivalent of Java's **switch** statement. One is semantically similar to nested if statements with case - when - then. The other is with boolean expressions being evaluated one at a time from top to bottom. The value of this case expression is equivalent to the value of the **then** statement corresponding to the first *true* **when** statement.

7. Features of a programming language sometimes persist longer than a feature of computing hardware that inspired and supported that part of the language's design. Similarly, features of hardware sometimes persist longer than some parts of a language's design that were created to take advantage of that feature in hardware.

   Give examples.

   The IBM 704 influenced the design of control statements, and prompted the development of Fortran. The Algol standard used several different syntaxes which, among other things, allowed Europeans to use a comma to denote a decimal point, while Americans could continue to use a period. Most of the popular languages of the past 50 years have been designed around the von Neumann architecture. These are called imperative languages. In a von Neumann computer, both data and programs are stored in the same memory. The CPU, which executes instructions, is separate from the memory. Therefore, instructions and data must be transmitted, or piped, from memory to the CPU. Results of operations in the CPU must be moved back to memory. Nearly all digital computers built since the 1940s have been based on the von Neumann architecture. The **register** keyword in C is a hint to the compiler that a variable will be used repeatedly, and so it should be stored in the CPU rather than in memory. However, modern compilers are far better at optimization than programmers, so this keyword is outdated and unnecessary.

8. Who most famously warned of the dangers of using the **goto** statement? What did Donald Knuth have to say about the use of the **goto** statement?

   Edsger Dijkstra warned that it would damage readability. Donald Knuth thought that they might sometimes be more efficient than other statements, so it was worth having them despite the poor readability.

9. Describes Ada's **for** loop. Are there some kinds of iteration that might be easier in Ada than in Java? Easier in Java than in Ada?

   Adas for loop can use any ordinal type variable for its counter. Arrays with ordinal type subscripts can be conveniently processed. For loops in

Java are more flexible - can have infinite loops, change loop variable inside body, etc. but is potentially more confusing to read. Since Ada's for loops are stricter, they are easier to read.

10. What does it mean to say that the guarded commands of Ada are nondeterministic?

    Guarded commands in Ada are nondeterministically chosen for execution when more than one of the statements are evaluated to true. This means that if there are three guarded statements and two of the three evaluate to true, then each time the program will use one of the two statements. It will not always use the one that appears first nor the one that appears last, but rather it will choose between them non-deterministically or randomly.

11. The header files in a C program contain function prototypes. What is a function prototype?

    A function prototype is a function declaration that gives the function's name and type signature, but does not specify the function body.

12. Every method in a Ruby program belongs to a class. A programmer can place a definition of a method inside the definition of a class or outside of the definition of any class that the programmer writes. To which class does the method belong in the second case?

    If a method is defined outside of the definition of any class that the programmer writes then the method belongs to the root object, **Object**

13. Distinguish between positional and keyword parameters.

    Positional parameters are bound based on the order in which they are given to the function. Keyword parameters are when the name of the formal parameter to which an actual parameter is to be bound is specified with the actual parameter in a call. This allows them to be input in any order.

14. Ruby blocks are closures. What does that mean?

A closure is an anonymous function which can be passed as a parameter.

Also, Ruby doesn't nest scope, so variables defined in a method are not accessible outside of the method (say, in the class which defines them).

15. What is a pure function?

    A pure function is a function where the return value is only determined

    by its input values, without observable side effects.

16. Some languages give programmers means to define both functions and procedures. Java doe not. Is that a serious limitation?

    It does not seem to be a serious limitation, it just means that performing

    procedures in Java is more verbose because it requires that the programmer instantiate the class with any needed methods.

17. Declarations of formal parameters in an Ada procedure can include, in addition to the names and types of the parameters, reserved words that do not appear in declarations in Java programs. What is the purpose of those reserved words?

    Ada allows the programmer to specify in mode, out mode, and inout mode

    for each formal parameter. This means that they can receive data from the actual parameter, they can send data to the actual parameter, or they can do both.

18. The C language imposes a constraint upon programmers who want to pass a multidimensional array to a function. What is the constraint? How did the design of the Java programming language eliminate that constraint for programmers who use that language?

    In C, programmers are required to pass the length of an array to a method.

    Specifying the length of an array (like in C) is unnecessary because Java arrays store a pointer to an address containing their length.

19. An activation record contains a return address, a dynamic link, parameters, and local variables.

    (a) To what does the return address point?

        To the instruction following the call in the code segment of the calling

        program unit

(b) To what does the dynamic link point?

To the base of the activation record instance of the caller

20. The stack will contain multiple activation records for a single subprogram under what circumstances?

In a recursive program there can be multiple activation records (although they will be incomplete) for a single subprogram.

21. How (or why?) does the LIFO protocol apply to calls to and returns from subprograms?

LIFO allows for subprograms to be nested within one another and for subprograms to be used as parameters within other subprograms.

22. Which important development in computer architecture has changed the way that the stack is used in some systems for facilitating calls to and returns from subprograms?

RISC (reduced instruction set computing) machines have parameters passed in registers in their compilers because RISC machines have more registers than CISC (complex instruction set computing) machines. Chapter 10 assumes parameters are passed in the stack though, as they had been in CISC machines.

23. A dynamic chain contains a history of what?

Dynamic chain represents the dynamic history of how execution got to its current position, which is always in the subprogram code whose activation record instance is on top of the stack.

24. Which two numbers are needed to compute the address of a local variable in a subprogram?

(chain_offset, local_offset) pair

25. How does a Ruby module differ from a class?

Modules are unlike classes in that they cannot be instantiated or subclassed and do not define variables. Methods that are defined in a module include the modules name in their names. They're just a collection of constants and methods.

26. Memory for variables can be allocated on the heap and on the stack. In which place or places is memory allocated for objects in C++? in Java?

In C++, variables can be allocated to the heap either by making them static or by allocating memory with the keyword new. However, variables which are initialized during the execution of a function are allocated to the stack. Java behaves the same way.

27. What problems were solved by the addition of genericity to Java?

Generics allow a type or method to operate on objects of various types while providing compile-time type safety.

28. What is the purpose of the static chain?

The static chain is a path of pointers which go from each function to its parent. They allow child subprograms to use variables which are local to their parent, grandparent, or farther up, without needing to copy those variables to the call stack.

29. What is a singleton?

A singleton is a class which provides a global access point to a single instance. This is useful for tasks which only need one point of access, like a file system.

30. What are the two parts of the definition of an abstract data type?

1. A type definition which allows program units to declare variables of the type but hides the representation of objects of the type. 2. A set of operations for manipulating objects of the type.

Chapter 13

Cameron Seebach

1. Most programming languages require the use of brackets to enclose the index in a reference to an element of an array.

   (a) Identify a language the requires the use of parentheses to enclose the index in a reference to an element of an array.

   (b) Why did the designers of the language choose parentheses rather than brackets?

   (a) The original FORTRAN used parentheses to to enclose an array reference.

   (b) At the time FORTRAN was first introduced, keypunch machines did not include the square bracket keys.

2. What is the relationship between a lexeme and a token?

   Lexemes and tokens are both produced by the lexer. A lexeme is a string that makes up some symbol in the language, while a token describes a whole category of lexemes.

   For example, an identifier lexeme is commonly a string of characters a-z and underscore. Regardless of what characters are chosen, the token is still identifier.

3. (a) What kind of symbols are found at the internal nodes of a parse tree?

(b) What kind of symbols are found at the leaves of a parse tree?

(a) Non-terminal symbols are found at the internal nodes of a parse tree.

(b) Terminal symbols are found at the leaves of a parse tree. All the non-terminals have been replaced at the leaves of the tree.

4. One of the most significant contributions from the developers of ALGOL 60 also limited the success of that language. What was that contribution?

Backus-Naur form described the syntax of ALGOL 60 in a way that was new and unfamiliar at the time.

5. What problem were the creators of Common LISP trying to solve?

The creators of Common LISP sought to combine several different dialects of Lisp, including Scheme, into one language.

6. What is an ambiguous context free grammar?

It is a context free grammar that can describe more than one distinct parse tree.

7. Contrast the complexity of algorithms that can parse strings that conform to the most general kinds of context free grammars and the complexity of the algorithms that can parse strings that conform to the grammars of programming languages?

Algorithms that parse the most general CFGs have a time complexity of O(n**3). For a programming language grammar, algorithms exist that run in O(n) time.

8. Java represents characters with Unicode. It is the first widely used programming language with this feature. What is the significance of this feature?

   It is easier to write programs intended for an international audience in Java. Less special work is required to support character sets outside of ASCII.

9. How does the binary coded decimal type differ from the floating point type?

   Binary coded decimal typically allows for a fixed number of digits before the decimal point, and a fixed number of digits afterwards. By contrast, floating point types have a number of digits of precision which is not related to the location of the decimal point.

   Also, binary coded decimal is typically made up of groups of 4 bits (0-F), each four bits representing a digit (0-9). As a result, a binary coded decimal number takes more bits than a floating point number to store the same number.

10. Identify a user-defined ordinal type in the Java programming language.

    The user-defined ordinal type in Java is the Enum.

11. Mathematicians and programmers might have different ideas about the precedence of Boolean operators. Explain.

    In mathematics, AND and OR have the same precendence, but in the C language AND has a higher precedence.

12. Programmers should use `===` rather than `==` to test the equality of the values of two expressions in JavaScript. Why?

Equality of strings and other objects is the best reason to use one over the other. === compares the content of two objects or strings, while == compares only the identity.

13. Describe a hazard of allowing short-circuited evaluation of expressions and side effects in expressions at the same time.

If the expression with a side effect is placed second in a short-circuit operation, it may not always be applied if the operation short-circuits.

This can lead to a situation where a programmer thinks that a mutating change has occurred, when in fact it has not.

14. Briefly describe the three steps in the mark-sweep algorithm for garbage collection.

(a) Traverse the tree of objects reachable from the 'root set', marking each as in-use.

(b) Scan all memory, freeing objects which are not in-use.

(c) Reset the in-use flag on each object still extant.

15. What led Yukihiro Matsumoto to create the Ruby programming language?

Matsumoto wanted an easy, truly object-oriented scripting language, but didn't like the options available at that time in 1993.

16. What did Microsoft aim to achieve with its development of the C# language?

Microsoft aimed to have a flagship language similar to C and C++ for use with its new Common Language Runtime. Portability and general-purposeness were other important design goals.

13.1 More questions for discussion and review.

1. The design of which machine influenced the design of the control statements in FORTRAN?

   The IBM 704 mainframe computer's assembly language influenced the control statements in the first version of FORTRAN.

2. How many different kinds of control statements must the designer of a programming language include in a language?

   Technically, just one, a "selectable goto". However, designers usually include at least one selection statement and one iteration statement.

3. What is the one question that applies in the design of all statements that allow selection or iteration?

   Should the control structure have multiple statements? In other words, should execution always begin with the first statement in the structure?

4. What is an advantage of requiring that the **then** and **else** clauses of an **if** statement be compound statements?

   Readability is increased. It is easier to distinguish which statements are in an if statement body when only compound statements are allowed.

5. How does the **switch** statement in C# differ from the **switch** statement in Java?

   C# allows only one of the paths to be executed, the first matching. Java requires a **break** statement to be added to each path for this behaviour.

6. Distinguish between 2 statements in Ruby that correspond to Java's **switch** statement.

   One of these forms is similar to a series of nested if statements, and evaluates to a value. The other is similar to the switch statement in Java

7. Features of a programming language sometimes persist longer than a feature of computing hardware that inspired and supported that part of the language's design. Similarly, features of hardware sometimes persist longer than some parts of a language's design that were created to take advantage of that feature in hardware.

   Give examples.

   The **register** keyword in C was intended as a hint to the compiler that a variable should be kept on a CPU register instead of in memory. Compiler technology has improved so much that the compiler and optimizer make better decisions about what to put in registers than humans can, and this keyword has thus fallen into disuse and deprecation.

   In the x86 assembly language, there is limited support for Binary Coded Decimal arithmetic. Modern programming languages do not have built-in Binary Coded Decimal types, and yet the BCD opcodes in the x86 architecture have been retained for backwards compatibility.

8. Who most famously warned of the dangers of using the **goto** statement? What did Donald Knuth have to say about the use of the **goto** statement?

   Edsger Dijkstra's famous letter *Go To Statement Considered Harmful* warned against the use of **goto**. Donald Knuth said it was usually a bad idea to use **goto**, but that there were a few exceptional cases where it improved the readability of a program.

9. Describes Ada's **for** loop. Are there some kinds of iteration that might be easier in Ada than in Java? Easier in Java than in Ada?

   Ada's for loop is less flexible than the for loop in Java, but as a result is more constrained and easier to read. A reverse iteration over a discrete range is mildly easier in Ada than Java, as Ada has the built-in reverse keyword for these situations. Java has easier iteration over Iterable objects, because Ada requires that the iterable be over a discrete range and not from an object.

10. What does it mean to say that the guarded commands of Ada are nondeterministic?

   Any of the guards that evaluate to true may have their expression chosen for execution, at random. This is part of Ada's concurrency support.

11. The header files in a C program contain function prototypes. What is a function prototype?

A function prototype contains the name, return type, and arguments of a function.

12. Every method in a Ruby program belongs to a class. A programmer can place a definition of a method inside the definition of a class or outside of the definition of any class that the programmer writes. To which class does the method belong in the second case?

It belongs to the Object class.

13. Distinguish between positional and keyword parameters.

Positional parameters must be passed in a distinct order, while keyword parameters can be passed in any order and may sometimes even not be passed, using a default value.

14. Ruby blocks are closures. What does that mean?

A Ruby block is an anonymous subprogram passed to a method. It can reference its enclosing environment.

15. What is a pure function?

A pure function is a function without side effects, meaning it does not modify the global state of the computer.

16. Some languages give programmers means to define both functions and procedures. Java does not. Is that a serious limitation?

This is not a serious limitation, because one can declare a function with return type void in Java, mimicking a procedure.

17. Declarations of formal parameters in an Ada procedure can include, in addition to the names and types of the parameters, reserved words that

do not appear in declarations in Java programs. What is the purpose of those reserved words?

Those words are **in**, **out**, and **inout**. They specify the modifiability of the parameters they describe. In parameters cannot be modified by the function. Out parameters cannot be read. Inout parameters can be both modified and read.

18. The C language imposes a constraint upon programmers who want to pass a multidimensional array to a function. What is the constraint? How did the design of the Java programming language eliminate that constraint for programmers who use that language?

A programmer must pass the number of columns in a two-dimensional array as part of the type signature, so that the compiler can build a mapping function. Java arrays carry with them a length attribute, eliminating this requirement.

19. An activation record contains a return address, a dynamic link, parameters, and local variables.

    (a) To what does the return address point?

    (b) To what does the dynamic link point?

The return address points to the next instruction to be executed in the calling code. The dynamic link points to the base of the activation record instance of the caller.

20. The stack will contain multiple activation records for a single subprogram under what circumstances?

The stack behaves like this when recursive subprograms are allowed.

21. How (or why?) does the LIFO protocol apply to calls to and returns from subprograms?

Subprogram activation records are added to the stack, and then the top is popped off as subprograms finish. The last run subprogram is always at the top.

22. Which important development in computer architecture has changed the way that the stack is used in some systems for facilitating calls to and returns from subprograms?

    The introduction of the call stack allowed for recursive subprograms.

23. A dynamic chain contains a history of what?

    Which subprograms have been called and in what order to get to the current subprogram.

24. Which two numbers are needed to compute the address of a local variable in a subprogram?

    You need the EP - the Environment Pointer - plus an offset.

25. How does a Ruby module differ from a class?

    A module cannot be instantiated - it is just a collection of methods and objects.

26. Memory for variables can be allocated on the heap and on the stack. In which place or places is memory allocated for objects in C++? in Java?

    In C++, local variables are on the stack, and variables that outlive a single function live on the heap and are created by **new**. In Java all objects live on the heap and are garbage collected.

27. What problems were solved by the addition of genericity to Java?

    Casts had to be done back and forth into and out of a data structure prior to generics.

28. What is the purpose of the static chain?

    The static chain allows the access of non-local variables.

29. What is a singleton?

A singleton is a class that is instantiated only once.

30. What are the two parts of the definition of an abstract data type?

A type definition and a set of operations defined on that type.

Chapter 14

Joe Sterchele

1. Most programming languages require the use of brackets to enclose the index in a reference to an element of an array.

   (a) Identify a language the requires the use of parentheses to enclose the index in a reference to an element of an array.

   (b) Why did the designers of the language choose parentheses rather than brackets?

   (a) Ada uses parentheses to enclose the index to an element of an array.

   (b) The reason that brackets are used, rather than parentheses, is that parentheses are also used to denote subprogram calls.

2. What is the relationship between a lexeme and a token?

   Tokens are the catagorey of the lexeme. Eg. 2 and int_literal (lexeme and token).

3. (a) What kind of symbols are found at the internal nodes of a parse tree?

   (b) What kind of symbols are found at the leaves of a parse tree?

   (a) Nonterminal

   (b) Terminal

4. One of the most significant contributions from the developers of ALGOL 60 also limited the success of that language. What was that contribution?

   BNF (Backus-Naur Form)

5. What problem were the creators of Common LISP trying to solve?

   They were trying to create one version of LISP, so that there wouldn't be so many dialects being used. With the large amount of dialects came a lack of potability.

6. What is an ambiguous context free grammar?

   A context-free grammar is a generative device for defining languages. Context-free gramars are ambiguous when they generate a sentential form that could have two or more distinct parse trees.

7. Contrast the complexity of algorithms that can parse strings that conform to the most general kinds of context free grammars and the complexity of the algorithms that can parse strings that conform to the grammars of programming languages?

   A parsing algorithm for an unambiguous grammar is ridiculously inefficient ($(O(n^3))$). More specifc alogorithms can be made for programming languages that have a complexity of $O(n)$. It is much more efficient to use specific algorithms.

8. Java represents characters with Unicode. It is the first widely used programming language with this feature. What is the significance of this feature?

   It was a 16-bit character set that included characters from most natural languages and ASCII.

9. How does the binary coded decimal type differ from the floating point type?

   Floating-point are represented as fractions and exponents while Decimal are stored with a fixed number of decimal digits with the decimal point at a fixed position in the value. The value 0.1 can be represented exactly in decimal, but in floating-point it would come with some uncertainty.

10. Identify a user-defined ordinal type in the Java programming language.

The two user-defined ordinal types in Java are enumeration and subrange.

11. Mathematicians and programmers might have different ideas about the precedence of Boolean operators. Explain.

    In math, AND and OR have equal precedence, but in most programming languages AND has a higher precedence than OR.

12. Programmers should use `===` rather than `==` to test the equality of the values of two expressions in JavaScript. Why?

    Programmers should use === instead of == because when you use == a string such as ”7” will be coerced to the number 7, but when you use === it remains a string.

13. Describe a hazard of allowing short-circuited evaluation of expressions and side effects in expressions at the same time.

    This allows subtle errors to occur. ”If the short-circuit evaluation is used on an expression and part od the expression that contains a side effect is not evaluated; the the side effect will occur only in complete evaluations of the wole expression. If program correctness depends on the side effect, short-circuit evaluation can result in a serious error.”

14. Briefly describe the three steps in the mark-sweep algorithm for garbage collection.

    All cells in the heap have their indicators set to indicate that they are garbage. Every pointer in the program is traced into the heap, and all reachable cells are marked as not being garbage. All cells in the heap that are marked as garbage are returned to the list of available space.

15. What led Yukihiro Matsumoto to create the Ruby programming language?

    He was dissatisfied with Perl and Python. He wanted a purely object-oriented language and neither of them lived up to it.

16. What did Microsoft aim to achieve with its development of the C# language?

    C# was meant to provide a language for component-based software development. It was geared towards development in the .NET Framework.

14.1 More questions for discussion and review.

1. The design of which machine influenced the design of the control statements in FORTRAN?

   IBM 704

2. How many different kinds of control statements must the designer of a programming language include in a language?

   Two, one for choosing between two different control flow paths and one for logically controlled iterations, or one if it is something akin to a selectable goto.

3. What is the one question that applies in the design of all statements that allow selection or iteration?

   Should the control structure have multiple entries?

4. What is an advantage of requiring that the **then** and **else** clauses of an **if** statement be compound statements?

   It helps with readability, and in certain languages there are no other indicators of whether or not something is apart of the if statement.

5. How does the **switch** statement in C# differ from the **switch** statement in Java?

   In C# it will only evaluate the first true **switch** statement, whereas in java it will look to evaluate all **switch** statements.

6. Distinguish between 2 statements in Ruby that correspond to Java's **switch** statement.

   Case statements, in Ruby, behave like nested if statements, while cond statements go through a list of statements that work as else if.

7. Features of a programming language sometimes persist longer than a feature of computing hardware that inspired and supported that part of the language's design. Similarly, features of hardware sometimes persist longer than some parts of a language's design that were created to take advantage

of that feature in hardware. Give examples.

The IBM 704 influenced the design of control statements.

8. Who most famously warned of the dangers of using the **goto** statement? What did Donald Knuth have to say about the use of the **goto** statement?

   Edsger Dijkstra noted The goto statement as it stands is just too primitive; it is too much an invitation to make a mess of ones program. Donald Knuth argued there were occasions when the efficiency of the goto outweighed its harm to readability.

9. Describes Ada's **for** loop. Are there some kinds of iteration that might be easier in Ada than in Java? Easier in Java than in Ada?

   Ada's **for** loop limits the scope of the variables inside it. If there is a variable described outside the loop then that variable will be the same after it ends.

10. What does it mean to say that the guarded commands of Ada are non-deterministic?

    Gaurded commands non-deterministically chosen for execution when more than one evaluates to true. Of the true statmentes, it will choose one randomly.

11. The header files in a C program contain function prototypes. What is a function prototype?

    A function prototype is a function declaration that gives the function's name and type signature, but does not specify the function body. It is also referred to as a function interface at times. In other languages theses are uncommon because subprograms do not need declarations since they do no need to be defined before they are called.

12. Every method in a Ruby program belongs to a class. A programmer can place a definition of a method inside the definition of a class or outside of the definition of any class that the programmer writes. To which class does the method belong in the second case?

    If a method is defined outside of the definition of any class that the programmer writes then the method belongs to the root object, **Object**.

13. Distinguish between positional and keyword parameters.

    Posistional parameters are refered to by the position that they are called in. Keyword parameters are refered to by the keyword associated with them.

14. Ruby blocks are closures. What does that mean?

    This means that a Ruby block contains a nested subprogram and its refrencing envrionment, which allows the subprogram to be called from anywhere in the program.

15. What is a pure function?

    A function that produces no side effects; that is, it modifies neither its parameters nor any variable defined outside the function.

16. Some languages give programmers means to define both functions and procedures. Java does not. Is that a serious limitation?

    Java does not allow the definition of procedures. It is not a limitation because functions can return void, which is essentialy the same as returning nothing.

17. Declarations of formal parameters in an Ada procedure can include, in addition to the names and types of the parameters, reserved words that do not appear in declarations in Java programs. What is the purpose of those reserved words?

    Ada allows the programmer to specify in mode, out mode, and inout mode for each formal parameter. This means that they can receive data from the actual parameter, they can send data to the actual parameter, or they can do both.

18. The C language imposes a constraint upon programmers who want to pass a multidimensional array to a function. What is the constraint? How did the design of the Java programming language eliminate that constraint for programmers who use that language?

    In C and C++ you need to specify the number of columns in the multidimensional array. In Java and C# arrays are objects. They are all single dimensioned, but the elements can be arrays. Each array inherits a named constant (length in Java and Length in C#) that is set to the length of the array when the array object is created.

19. An activation record contains a return address, a dynamic link, parameters, and local variables.
    (a) To what does the return address point?
    (b) To what does the dynamic link point?

    The return address usually consists of a pointer to the instruction following the call in the code segment of the calling program unit. The dynamic link points to the base of the activation record instance of the caller.

20. The stack will contain multiple activation records for a single subprogram under what circumstances?

    In a recursive program there can be multiple activation records (although they will be incomplete) for a single subprogram.

21. How (or why?) does the LIFO protocol apply to calls to and returns from subprograms?

    Last in first out applies because the last thing on the activation record is the first thing that's returned (Stack). LIFO allows for subprograms to be nested within one another and for subprograms to be used as parameters within other subprograms.

22. Which important development in computer architecture has changed the way that the stack is used in some systems for facilitating calls to and returns from subprograms?

    RISC (reduced instruction set computing) machines have parameters passed in registers in their compilers because RISC machines have more registers than CISC (complex instruction set computing) machines. Chapter 10 assumes parameters are passed in the stack though, as they had been in CISC machines.

23. A dynamic chain contains a history of what?

    It contains a history of all subprogram activation records, but in the reverse order of when they were activated. (LIFO Stack)

24. Which two numbers are needed to compute the address of a local variable in a subprogram?

    To compute the address of a local variable you need the (chain_offset, local_offset) pair.

25. How does a Ruby module differ from a class?

Modules are unlike classes in that they cannot be instantiated or subclassed and do not define variables. Methods that are defined in a module include the modules name in their names.

26. Memory for variables can be allocated on the heap and on the stack. In which place or places is memory allocated for objects in C++? in Java?

n C++, variables can be allocated to the heap either by making them static or by allocating memory with the keyword new. However, variables which are initialized during the execution of a function are allocated to the stack. Java behaves the same way.

27. What problems were solved by the addition of genericity to Java?

Before generics you could get a runtime exception, even though your code compiled, when trying to convert data types. Generics allow a type or method to operate on objects of various types while providing compile-time type safety.

28. What is the purpose of the static chain?

The static chain is a path of pointers which go from each function to its parent. They allow child subprograms to use variables which are local to their parent, grandparent, or farther up, without needing to copy those variables to the call stack.

29. What is a singleton?

A singleton is a class which provides a global access point to a single instance. This is useful for tasks which only need one point of access, like a file system.

30. What are the two parts of the definition of an abstract data type?

1. A type definition which allows program units to declare variables of the type but hides the representation of objects of the type. 2. A set of operations for manipulating objects of the type.

Chapter 15

Htut Khine Win

1. Most programming languages require the use of brackets to enclose the index in a reference to an element of an array.

   (a) Identify a language the requires the use of parentheses to enclose the index in a reference to an element of an array.

   (b) Why did the designers of the language choose parentheses rather than brackets?

   (a) Ada

   (b) The architects of Ada chose parentheses to enclose the index in a reference to an element of an array so that it is consistent between array references and function calls in expressions. Moreover, the choice is in part influenced by the notion that both array element references and function calls are mappings.

2. What is the relationship between a lexeme and a token?

   A lexeme is the lowest-level syntatic unit in the programming languages. Lexemes are grouped and each lexeme group is represented by a token. Therefore, a token is a category of its lexemes. For example, an identifier is a token that can have lexemes.

3. (a) What kind of symbols are found at the internal nodes of a parse tree?

(b) What kind of symbols are found at the leaves of a parse tree?

(a) Nonterminal symbols are found at the internal nodes of a parse tree.

(b) Terminal symbols are found at the leaves of a parse tree.

4. One of the most significant contributions from the developers of ALGOL 60 also limited the success of that language. What was that contribution?

The revised method of syntax description called Backus-Naur Form, or BNF.

5. What problem were the creators of Common LISP trying to solve?

LISP creators were trying to solve the lack of portability among programs written in various dialets.

6. What is an ambiguous context free grammar?

A grammar that generates a sentential form for which there are two or more distinct parse trees is called ambiguous context free grammar.

7. Contrast the complexity of algorithms that can parse strings that conform to the most general kinds of context free grammars and the complexity of the algorithms that can parse strings that conform to the grammars of programming languages?

Parsing algorithms that work for unambiguous grammar are complicated and inefficient.The complexitiy of such algorithms is O n to power 3.All algorithms used for the syntax analyzers that conform to the grammars of programming languages have complexity O(n).

8. Java represents characters with Unicode. It is the first widely used programming language with this feature. What is the significance of this feature?

Before Unicode, most commonly used coding was the 8-bit ASCII code. Unicode is a 16-bit character set. Unicode made huge impacts on the globalization of business and the need for computers to communicate with other computers around the world.

9. How does the binary coded decimal type differ from the floating point type?

The float type is stored in four bytes of memory. Binary coded decimals are stored one digit per byte, in some cases, they are packed two digits per byte and it takes at least four bits to code a decimal digit.

10. Identify a user-defined ordinal type in the Java programming language.

Enumeration types.

11. Mathematicians and programmers might have different ideas about the precedence of Boolean operators. Explain.

In Mathematics, OR and AND operators have equal precedence. However, C-based languages assign higher precedence to AND than OR because programmers tend to associate multipliacation with AND and addition with OR, which would naturally assign higher precedence to AND.

12. Programmers should use === rather than == to test the equality of the values of two expressions in JavaScript. Why?

Because == forces the operand to match the type. Yet, === does not coerce the operands of this operator.

13. Describe a hazard of allowing short-circuited evaluation of expressions and side effects in expressions at the same time.

Suppose that short-circuit evaluation is used on an expression and part of the expression that contains a side effect is not evaluated; then the side effect will occur only in complete evaluations of the whole expression. If the program correctness depends on the side effect, short-circuit evaluation can cause serious problems.

14. Briefly describe the three steps in the mark-sweep algorithm for garbage collection.

First, all cells in the heap have their indicators set to indicate they are garbage.In the second part, every pointer in the program is traced into the heap and all reachable cells are marked as not being garbage. In the third part, all cells in the heap that have not been specifically marked as still being used are returned to the list of available spaces.

15. What led Yukihiro Matsumoto to create the Ruby programming language?

Because he was dissatisfied with the design of Perl and Python, although both of them support object-oriented language, neither is a pure object-oriented language, at least in the sense that each has primitive(nonobject) types and each supports functions.

16. What did Microsoft aim to achieve with its development of the C# language?

The purpose of C# is to provide a language for component-based software development, specifically for .NET framework.

15.1 More questions for discussion and review.

1. The design of which machine influenced the design of the control statements in FORTRAN?

   IBM 704 system

2. How many different kinds of control statements must the designer of a programming language include in a language?

   Even though it is possible to use only one GOTO statement, the minimum number of control statements in a language that does not use GOTO is two. One for choosing between control flow paths, and one for logically controlled iterations.

3. What is the one question that applies in the design of all statements that allow selection or iteration?

   Should the control structures have multiple entries?

4. What is an advantage of requiring that the **then** and **else** clauses of an **if** statement be compound statements?

   Requiring compound statements helps increase the readability and writability for programmers when using nested selector statements, that otherwise can get very messy and complicated.

5. How does the **switch** statement in C# differ from the **switch** statement in Java?

   In Java, the switch statement does not allow case expressions anywhere except the top level in the body of the switch. C# allows the execution of more than one segment. In C#, the control expressions as well as case statements can be strings.

6. Distinguish between 2 statements in Ruby that correspond to Java's **switch** statement.

Case expressions are the Ruby equivalent of Java's **switch** statement. One is semantically similar to nested if statements with case - when - then. The other is with boolean expressions being evaluated one at a time from top to bottom. The value of this case expression is equivalent to the value of the **then** statement corresponding to the first *true* **when** statement.

7. Features of a programming language sometimes persist longer than a feature of computing hardware that inspired and supported that part of the language's design. Similarly, features of hardware sometimes persist longer than some parts of a language's design that were created to take advantage of that feature in hardware.

   Give examples.

The IBM 704 influenced the design of control statements. Von Neumann architecture and imperative languages.

8. Who most famously warned of the dangers of using the **goto** statement? What did Donald Knuth have to say about the use of the **goto** statement?

Edsger Dijkstra noted The goto statement as it stands is just too primitive; it is too much an invitation to make a mess of ones program. Donald Knuth argued there were occasions when the efficiency of the goto outweighed its harm to readability.

9. Describes Ada's **for** loop. Are there some kinds of iteration that might be easier in Ada than in Java? Easier in Java than in Ada?

The for loop looks like this for variable in [reverse] discreterange loop end loop;

Adas for loop can use any ordinal type variable for its counter. Arrays with ordinal type subscripts can be conveniently processed.

10. What does it mean to say that the guarded commands of Ada are nondeterministic?

Guarded commands in Ada are nondeterministically chosen for execution when more than one of the statements are evaluated to true. This means that if there are three guarded statements and two of the three evaluate to true, then each time the program will use one of the two statements. It will not always use the one that appears first nor the one that appears last, but rather it will choose between them non-deterministically or randomly.

11. The header files in a C program contain function prototypes. What is a function prototype?

A function prototype is a function declaration that gives the function's name and type signature, but does not specify the function body. It is also referred to as a function interface at times. In other languages these are uncommon because subprograms do not need declarations since they do no need to be defined before they are called.

12. Every method in a Ruby program belongs to a class. A programmer can place a definition of a method inside the definition of a class or outside of the definition of any class that the programmer writes. To which class does the method belong in the second case?

If a method is defined outside of the definition of any class that the programmer writes then the method belongs to the root object, **Object**.

13. Distinguish between positional and keyword parameters.

Positional parameters are bound based on the order in which they are given to the function. Keyword parameters are when the name of the formal parameter to which an actual parameter is to be bound is specified with the actual parameter in a call. This allows them to be input in any order.

14. Ruby blocks are closures. What does that mean?

A closure is an anonymous function which can be passed as a parameter. Also, Ruby doesn't nest scope, so variables defined in a method are not accessible outside of the method (say, in the class which defines them).

15. What is a pure function?

A pure function is a function where the return value is only determined by its input values, without observable side effects.

16. Some languages give programmers means to define both functions and procedures. Java doe not. Is that a serious limitation?

It does not seem to be a serious limitation, it just means that performing

procedures in Java is more verbose because it requires that the programmer instantiate the class with any needed methods.

17. Declarations of formal parameters in an Ada procedure can include, in addition to the names and types of the parameters, reserved words that do not appear in declarations in Java programs. What is the purpose of those reserved words?

Ada allows the programmer to specify in mode, out mode, and inout mode

for each formal parameter. This means that they can receive data from the actual parameter, they can send data to the actual parameter, or they can do both.

18. The C language imposes a constraint upon programmers who want to pass a multidimensional array to a function. What is the constraint? How did the design of the Java programming language eliminate that constraint for programmers who use that language?

In C, programmers are required to pass the length of an array to a method.

Specifying the length of an array (like in C) is unnecessary because Java arrays store a pointer to an address containing their length.

19. An activation record contains a return address, a dynamic link, parameters, and local variables.

    (a) To what does the return address point?

    (b) To what does the dynamic link point?

The return address usually consists of a pointer to the instruction follow-

ing the call in the code segment of the calling program unit. The dynamic link points to the base of the activation record instance of the caller.

20. The stack will contain multiple activation records for a single subprogram under what circumstances?

In a recursive program there can be multiple activation records (although they will be incomplete) for a single subprogram.

21. How (or why?) does the LIFO protocol apply to calls to and returns from subprograms?

LIFO allows for subprograms to be nested within one another and for subprograms to be used as parameters within other subprograms.

22. Which important development in computer architecture has changed the way that the stack is used in some systems for facilitating calls to and returns from subprograms?

RISC (reduced instruction set computing) machines have parameters passed in registers in their compilers because RISC machines have more registers than CISC (complex instruction set computing) machines. Chapter 10 assumes parameters are passed in the stack though, as they had been in CISC machines

23. A dynamic chain contains a history of what?

Dynamic chain represents the dynamic history of how execution got to its current position, which is always in the subprogram code whose activation record instance is on top of the stack.

24. Which two numbers are needed to compute the address of a local variable in a subprogram?

To compute the address of a local variable you need the (chain offset, local offset) pair.

25. How does a Ruby module differ from a class?

Modules are unlike classes in that they cannot be instantiated or subclassed and do not define variables. Methods that are defined in a module include the modules name in their names.

26. Memory for variables can be allocated on the heap and on the stack. In which place or places is memory allocated for objects in C++? in Java?

In C++, variables can be allocated to the heap either by making them static or by allocating memory with the keyword new. However, variables which are initialized during the execution of a function are allocated to the stack. Java behaves the same way.

27. What problems were solved by the addition of genericity to Java?

Generics allow a type or method to operate on objects of various types while providing compile-time type safety

28. What is the purpose of the static chain?

The static chain is a path of pointers which go from each function to its parent. They allow child subprograms to use variables which are local to their parent, grandparent, or farther up, without needing to copy those variables to the call stack.

29. What is a singleton?

A singleton is a class which provides a global access point to a single instance. This is useful for tasks which only need one point of access, like a file system.

30. What are the two parts of the definition of an abstract data type?

---

1. A type definition which allows program units to declare variables of the type but hides the representation of objects of the type. 2. A set of operations for manipulating objects of the type.