CS 2160 Midterm Question Pool

1. A terabyte is 2^\_\_\_ bytes (5)

**A. 40**

B. 20

C. 10

D. 30

2. An embedded computer is distinguished from other types of computer by the fact that it (6)

A. Must survive harsh operating conditions

B. has strict power requirements

**C. Primarily runs a single, predetermined application**

D. is built into an incredibly small space

3. The role of the personal computer is now being replaced by the (7)

A. super computer

B. wearable computer

C. cloud

**D. personal mobile device**

4. The sales of smart phones and tablets have leveled off while the sales of PCs and conventional cell phones have continued their exponential rise (7)

A. T

**B. F**

5. The concept of cloud computing is most closely associated with (7)

A. Personal Mobile Devices

B. Transaction Processing

C. Vague computing

**D. Warehouse Scale Computing**

6. Understanding the hardware layer (8)

A. is the key to an early retirement

B. provides for a steady income

**C. makes you a better programmer**

D. builds character

7. Which area of performance improvement is out of the scope of this course (9)

**A. Algorithms**

B. The I/O system

C. Memory systems

D. Programming languages

8. Which kind of parallelism is not discussed in computer organization and architecture? (10)

A. Data-level parallelism

B. Thread-level parallelism

**C. Circuit-level parallelism**

D. Instruction-level parallelism

9. Which has the least influence on the performance of a computer running a program? (10)

**A. The power supply wattage**

B. The operating system

C. The processor

D. The algorithm chosen

E. The programming language or compiler

F. The I/O system and devices

**G. The size of the screen**

10. Moore's Law (11)

**A. Gives expectations of how chip capacity will change about every two years**

B. states that performance improvements are limited by the kind of problem you are solving

C. Relates power consumed by a device to capacitive load, voltage and switching frequency

D. Underscores the importance of parallelism

E. Shows that latency is more expensive to improve than bandwidth or throughput

F. Says that you should optimize tasks that are frequently performed

**G. Relates the progress on the number of transistors per chip over time**

11. Which is not one of the eight great ideas in Computer Architecture? (11)

A. Reliability via redundancy.

B. Performance via pipelining.

C. Make the common case fast.

**D. Fail early**

12. The notion that integrated circuit resources double every 18-24 months is known as (11)

A. Ohm's Law

B. Boyle's Law

C. Amdahl's Law

**D. Moore's Law**

13. The principle of prediction is based on the idea that (12)

A. You should eat dessert first.

**B. It is better to ask for forgiveness than ask for permission.**

C. If you don't succeed at first, try, try again.

D. When the going gets tough, the tough get going.

14. Which is not a "great idea" in computer architecture? (12)

**A. Flash cache.**

B. Pipelining.

C. Branch Prediction.

D. Memory hierarchy.

15. Which is not in the hierarchical layers of a system? (13)

A. Hardware

B. Applications software

**C. I/O software**

D. Systems Software

16. What do we call the binary representation of machine instructions? (14)

**A. Machine language**

B. source code

C. Assembly language

D. Bytes

17. What is the job of an assembler (15)

A. Run assembly code

**B. Translate assembly code into machine language**

C. Link together code modules into a single executable module

D. Translate source code in to assembly code

18. What is the job of an assembler? (15)

**A. It produces binary machine instructions from the assembly code text**

**B. It translates assembly instructions into machine instructions**

C. It translates high-level source code into assembly language

D. It translates high-level source code into machine instructions

E. It executes machine instructions

F. It optimizes loop structures

G. It translates source code into byte code

19. What is not an advantage of high-level code (16)

A. It is closer to natural language

**B. It produces the most compact run-time code**

C. It improves programmer productivity

D. It allows machine independence

20. The chip fabrication process begins with (26)

**A. An ingot**

B. a die

*C. a wafer*

D. a package

21. The chip manufacturing process begins with (26)

A. Doping the silicon surface

**B. A silicon ingot**

C. Photolithography

D. Slices of silicon about 0.1" thick

E. Slicing and dicing

**F. A large, sausage-shaped piece of silicon**

G. Checking the wafer for defects

22. A major difficulty in determining performance is that it (26)

*A. Can be affected by other components in the computer*

**B. Is application dependent**

**C. Can be measured in different ways**

D. Is largely independent of the processor itself

E. Is heavily influenced by marketing

F. No longer relevant in today's computing market

G. Takes a long time to do accurately

23. Which describe two steps in the chip fabrication process? (26)

**A. Slicing and dicing**

B. Testing and resting

C. Waxing and faxing

D. Slipping and shipping

24. The rectangular area on a wafer containing the circuitry for a single chip is a(n) (27)

A. imprint

*B. module*

**C. die**

D. mask

25. The percentage of good dies on a wafer is called (27)

A. The success rate

B. The beneficial margin

C. Profit

**D. The yield**

26. Yield is primarily affected by (28)

**A. Wafer defects**

B. Transistor size

C. N and P doping

D. Etching chemistry

27. A major problem in measuring performance of a computer is (28)

**A. It can be measured several ways**

B. Variations in the manufacturing process

C. Laughing at the English translation of the instruction manual

D. It is impossible to match conditions from run to run

28. The time it takes a computer to complete a task is an example of (29)

A. Consistency

B. Bandwidth

*C. Clock frequency*

**D. Response time**

29. In addition to cost, what is important in the selection of an off-the-shelf computer for a particular application? (29)

*A. The speed of the network card.*

B. Range of power supply voltages the machine will tolerate

C. The size of the screen.

**D. Limitations in performance.**

30. Quantification of performance can be multi-dimensional. What difficulty does this pose? (29)

A. Understanding the different dimensions of performance at the same time.

B. Knowing whether or not the measurements in different dimensions are equally accurate.

C. Trusting the numbers because statistics can lie.

**D. Transforming performance to a unidimensional scale so that results can be compared.**

31. The total time required for a computer to complete a task, including I/O and operating system overhead is called (29)

**A. Execution time**

B. CPU Time

C. Application time

D. Absolute time

32. Performance is inversely proportional to execution time (30)

**A. T**

B. F

33. Performance and execution time are (30)

**A. Inversely related**

B. proportionally related

C. exponentially related

D. The same thing

34. Sometimes improving bandwidth as opposed to latency is a way of improving performance. Which is an example of improving bandwidth? (30)

*A. More streamlined procedures for cashiers in the food court*

B. Raising the speed limit on I-25 to Denver

C. Making exams shorter so they can be graded faster

**D. Adding more lanes on I-25 to Denver**

E. Faster cash registers in the cashier lanes

F. Moving dorms closer to classroom buildings so students can get to class faster

**G. More open cashier lanes in the food court**

35. Which of the following indicators are inversely proportional to performance? (31)

A. Instructions per second

**B. Execution time**

C. Throughput

D. Bandwidth

E. Memory bits transferred per second

**F. Energy or power consumption**

G. Clock speed

36. If machine A completes a task in 10 seconds and machine B completes the task in 2 seconds, how many times faster is machine B than machine A? (31)

A. 400%

**B. Five**

C. Two and a half

D. Four

37. If machine A completes a task in 10 seconds and machine B completes the task in 2 seconds, how many times slower is machine A than machine B? (31)

A. Four

B. Two and a half

C. Three

**D. Five ggg**

38. The CPU time spent in the operating system performing tasks on behalf of a program is called (32)

**A. System CPU time**

B. execution time ggg

C. user CPU time

D. wasted time

39. Applications running on servers are particularly sensitive to (32)

**A. I/O time ggg**

B. disk performance

C. the amount of main memory

D. cache memory

40. Machine A is 4 times as fast as machine B. B runs an application in 30 seconds. How fast will A run it? (33)

A. 90 seconds

B. 120 seconds

**C. 7.5 seconds ggg**

D. 10 seconds

41. On which of the following does execution time not depend? (34)

A. Clock cycle time

B. Number of instructions executed

**C. Amount of system memory ggg**

D. Average cycles per instruction

42. Which of the following is a typical challenge for computer designers (34)

A. Building machines which appeal to both men and women.

B. Balancing the size of memory and the weight of the machine.

C. Staying awake during the boring parts of class.

**D. The tradeoff between the number of clock cycles needed for a program to execute and the length of each clock cycle. ggg**

43. Which results in better performance? (34)

**A. Decreased CPI ggg**

B. Increased instruction count

C. Increased CPI

D. Decreased clock rate

44. The average number of clock cycles per instruction (CPI) in a program is an integer value (35)

**A. F ggg**

B. T

45. The execution time of a program is related to (35)

**A. The number of instructions times the average CPI for the program. ggg**

B. The number of instructions divided by the clock speed

C. The average CPI times the clock speed. ggg

D. The number of instructions times the clock speed. ggg

46. Cycles Per Instruction (CPI) must be an integer value (35)

**A. F ggg**

B. T

47. Two computers executing the same instructions complete the task in different amounts of time. What must be the case? (36)

A. One machine did more work than the other

B. A mistake was made

C. Cycles per instruction are different ggg

**D. Clock speeds are different ggg**

48. Machine B is 50% faster than Machine A. B runs a program in 12 seconds. How long does it take A to run it (37)

A. 12 seconds

B. Ten seconds

**C. 18 seconds ggg**

D. Eight seconds

**E. Eighteen seconds ggg**

F. 15 seconds

G. Nine seconds

49. CPU A has an architecture with a CPI of 3 and CPU B has a CPI of 2. What is the performance difference? (38)

A. 1.5

**B. It cannot be determined. ggg**

C. 0.33

D. 0.67

50. If two machines run the same program and the machines' instruction sets have the same CPI, on what does execution time depend (38)

**A. Clock cycle time ggg**

B. Number of memory accesses

C. The paging algorithm

D. Cache size

51. The compiler can affect CPI (39)

A. F ggg

**B. T**

52. How can a CPI of less than 1.0 be achieved? (39)

A. Shorter instructions

**B. Issuing multiple instructions per clock cycle ggg**

C. Executing fewer instructions

D. Faster clock cycle time

53. The measure of the frequency of instructions of certain types across a program or multiple programs is the (39)

A. Instruction set architeture.

**B. Instruction mix**.

C. Average CPI.

D. Program count.

54. If Program B on Machine B has fewer instructions than Program A on Machine A, it will run faster unless (40)

A. Machine B doesn't have more cores.

**B. Machine B's CPI isn't too large.**

C. Machine A has more memory.

D. Machine B's clock speed isn't too high. ggg

55. What translates a symbolic representation of computer instructions into binary so that the processor can execute the instructions? (40)

**A. An assembler. ggg**

B. A converter

C. A translator.

D. A compiler.

56. The main reason for the shift from uniprocessors to multi-core processors was (40)

**A. The power wall ggg**

B. iPhones and iPads

C. The internet

D. Personal mobile devices

57. The primary technology used in fabricating computer chips is (41)

**A. Complementary Metal Oxide Semiconductor (CMOS)**

B. Tunneling

C. Three-dimensional printing

D. NAND flash

58. Which is not a factor in CPU power consumption? (41)

A. Switching frequency

B. Capacitive load

**C. Instruction length ggg**

D. Supply voltage

59. An increase in which of the following has the greatest effect on CPU power consumption (41)

A. Capacitive load ggg

B. CPI

C. Switching frequency

**D. Voltage**

60. The big problem with the switch to multi-processing is that (43)

A. More heat-sinking is required

B. Performance measures are very different ggg

**C. Programmers have to think differently**

D. The chips are more difficult to manufacture ggg

61. A key to the future of computation is (43)

A. Making computer science courses harder.

**B. Understanding parallelism ggg**

C. Doubling the size of computer science programs.

D. Creating lower-power processors ggg

62. Which of the following does not result in performance gains? (45)

**A. Lower voltage requirements ggg**

B. Memory Hierarchies

C. Prediction

D. Pipelining

63. One of Hennessey and Patterson's design rules is (46)

A. Get the product to market as soon as possible

**B. Make the common case fast ggg**

C. Memory is cheap

D. Make the instruction set as compact as possible

64. SPEC benchmarks are (46)

A. Unreliable because testers cheat

**B. A long-standing set of standard test programs ggg**

C. Obsolete because of recent innovations in chip design

D. A brand new way to measure performance

65. To get a better picture of CPU performance (47)

A. Tests are run multiple times

B. Customers are allowed to assess CPU performance on their own

C. The industry has standards for honesty and integrity in testing

**D. A suite of benchmark programs is used ggg**

66. Multiple SPEC ratios are combined using (47)

A. The arithmetic mean

B. Covariance

**C. The geometric mean**

D. Logarithms

67. Adding more cores to a processor will not improve performance if (49)

A. Several parts of the problem can be worked on at once

**B. The program only contains steps which must be executed in sequence ggg**

C. The program can be broken into multiple parallel paths

D. Program execution speed depends on the clock frequency

**E. No part of the program can be done in parallel with another part ggg**

F. The cores are all the same size

G. More cores means more power consumption

68. Idling computers consume a tiny fraction of the power they use at full load (50)

A. T ggg

**B. F**

69. Energy efficiency can be an important performance goal (50)

A. F

**B. T ggg**

70. When comparing the performance of different operating systems on different processors, instruction count can be ignored (50)

A. T

**B. F ggg**

71. What is the vocabulary of commands understood by the processor? (62)

A. The binary executable

**B. The instructions**

C. The byte code

**D. The instruction set**

E. The instruction set architecture ggg

F. The assembly language

G. The intermediate code

72. The vocabulary of the CPU is the (62)

A. Organization

**B. Instructions**

**C. Instruction set ggg**

D. Datapath

E. Description

F. Programs

G. Architecture

73. The fastest access to data is via (64)

A. Disk

**B. Registers**

C. Main memory

D. Cache ggg

74. The contents of a memory location whose address is contained in register $s1 may be expressed as: (64)

A. Mem[$s1]

B. 4($s1)

**C. Mem[Regs[$s1]]**

D. Regs[$s1]

E. Regs[Mem[$s1]]

F. $s1 ggg

75. The C code "d = a - e" could be represented in MIPS assembly as (65)

**A. SUB d, a, e**

**B. sub $t0, $t1, $t2 where d=$t0, a=$t1 and e=$t2**

C. SUB d, e, a

D. sub $t0, $t2, $t1 where d=$t0, a=$t1 and e=$t2

E. SUB a, e, d

F. d SUB a, e

G. sub $t0, $t1, $t2 where d=$t0, a=$t2 and e=$t1

76. Which language will have the most lines of code to implement a sort algorithm? (66)

A. Python

**B. Assembly code ggg**

**C. MIPS Assembly Language**

D. Pascal

E. C#

F. Java

G. C++

77. In the instruction "add $s0, $s1, $s2", $s1 and $s2 are (67)

A. Outputs

**B. Registers**

C. Memory locations

D. Bytes

E. Immediate values

**F. Operands ggg**

G. Arguments

78. In MIPS, data is moved to and from memory using (68)

A. Value instructions

B. Argument instructions

**C. Load and Store instructions ggg**

D. I-type add and subtract instructions

**E. Data transfer instructions**

F. Branch instructions

G. R-type add and subtract instructions

79. What is the generic name of instructions which move data between the processor and memory? (68)

A. Branch instructions

B. Arithmetic instructions

**C. Data transfer instructions ggg**

**D. Data transfer**

E. Cache control

F. Memory Control

G. Register-register

80. What is the reason that addresses of integer data stored in main memory must have the last two bits set to 0? (69)

A. Memory is scarce and must be used efficiently

B. The cache can operate more efficiently

C. Data can be located anywhere in memory.

**D. Word data must be four-byte aligned ggg**

**E. There is an alignment restriction**

F. The data bus is only 30 bits wide

G. Words must be eight-byte aligned

81. Memory alignment means (69)

A. More than one word can be read from memory at once

B. Dynamic RAM is faster than Static RAM

C. Similar data items are placed next to each other in memory

D. All eight bits of a byte must reside on the same chip

**E. The bottom two bits of a MIPS instruction address must be zero**

F. Memory chips must be placed next to each other

**G. 4-byte words must begin on 4-byte boundaries ggg**

82. If four-byte alignment is required, at which address could an integer not be stored? (70)

A. 0x00123454

**B. 0x00410003**

C. 0x00418210

D. 0x00ABFEDC ggg

E. 0x0041791C

**F. 0x0041000E**

G. 0x00410AA4

83. The typical MIPS register-to-register arithmetic instruction (e.g., ADD, SUB) contains references to how many registers? (72)

**A. Three**

B. None

C. Two

**D. 3 ggg**

E. Five

F. One

G. Four

84. The typical MIPS immediate instruction (e.g., ADDI, SUBI) contains references to how many registers? (72)

A. Three

**B. Two**

C. One

D. None

E. Four

**F. 2 ggg**

G. Five

85. Immediate operands are (72)

A. Obtained by adding an offset to a register ggg

**B. Data located in the instruction itself**

**C. Data encoded in the instruction**

D. Data located in registers

E. Data located in cache

F. Data located in main memory

G. Calculated from register values

86. Which 16-bit hex value would represent a negative number? (74)

A. 0x100f

B. 0x7777

**C. 0x8001**

D. 0x0001

E. 0X7fff

F. 0x0fff

**G. 0xffff ggg**

87. In a 32-bit MIPS instruction, the location of bit 0 (the least significant bit) (74)

A. T=Is on the left and middle

B. Is in the middle and the right

**C. is on the right**

**D. The farthest right bit' ggg**

E. Depends on the instruction

F. Is on the left

G. Varies by memory manufacturer

88. Negative numbers can be represented in binary (75)

A. F

**B. T ggg**

89. In binary, subtracting a large positive number from a small positive number produces (75)

A. a zero in the leftmost bit of the result

**B. A one in the left bit of the result ggg**

C. an overflow condition

D. an underflow condition

90. Which set of binary numbers are the two's complement of each other? (76)

A. 00000001 00000000

B. 00000000 11111111

C. 1010 0101

**D. 00000001 11111111**

**E. 0101 1011**

F. 1111 0000

G. 00000011 11111100 ggg

91. Which signed number is the same as 0x80? (78)

A. 0x7f80

**B. 0xFFFFFF80**

C. 0x807FFFFF

D. 0x7FFFFF80

**E. 0xff80**

F. 0xff08

G. 0x80FFFFFF

92. The interface between low-level hardware and software is called (78)

A. Instruction-level parallelism

B. The operating system ggg

C. The interpreter.

D. Interrupt subsystem.

**E. The instruction set architecture**

F. Cache.

93. The problem with one's-complement representation of negative numbers was (79)

**A. There were two zeroes**

B. There were more negative numbers than positive

C. It was hard to recognize the difference between positive and negative numbers

**D. It was more difficult to implement than twos-complement**

E. There is no such thing as one's complement.

F. It confused the programmers

G. There were more positive numbers than negative

94. In biased notation, the most negative value is (79)

A. 0x10000000

B. 0xFFFFFFFF ggg

**C. 0x0000**

D. 0xfffe

E. ox80000000

F. 0x0001

**G. 0x00000000**

95. What is the native, binary form of instructions used by the processor? (81)

A. Byte Code

B. Intermediate Language

**C. Machine instructions**

D. Source code.

E. Assembly language

**F. Machine language ggg**

G. Assembly code.

96. All MIPS instructions have (81)

**A. An Operations Code field ggg**

**B. Length of 32 bits ggg**

C. Immediate data

D. A byte which indicates how long it is

E. A field for the Rd register

F. Embedded memory addresses

G. A function code

97. Which instruction field is 6-bits long? (82)

**A. Op Code ggg**

**B. Fct Code ggg**

C. Rt

D. Rs

E. Shamt

F. Rd

G. Imm

98. Which is not a MIPS type of instruction? (82)

A. I

B. R

**C. B ggg**

D. J

99. The easiest conversion is from (82)

A. Decimal to Binary

**B. Hexadecimal to binary. ggg**

C. binary to decimal

**D. Binary to hexadecimal ggg**

E. positive to negative

F. hexadecimal to decimal

G. Decimal to Hexadecimal.

100. To refer to each of the 32 MIPS processor register, an instruction needs (83)

**A. 5 bits ggg**

**B. Five bits ggg**

C. 16 bits

D. 32 bits

E. Three bits

F. Six bits

G. Four bits

101. How many bits in a MIPS instruction are allocated for immediate data (84)

A. 12

**B. 16 ggg**

**C. Sixteen ggg**

D. 15

E. Ten

F. Eight

G. Twenty-four

102. Which instruction does MIPS not have? (87)

A. Shift left

B. ADD

**C. NOT**

D. Shift right

E. XOR

F. ADDI

**G. SUBI ggg**

103. The shamt field in the MIPS instruction is use for which instruction? (88)

A. NOR

**B. SRL**

C. XOR

D. Branch

E. SLLI

F. SLT ggg

**G. SLL**

104. Which is not a field in the MIPS instruction format? (88)

A. shamt

B. op

C. rt

D. rs

**E. instruction length00 ggg**

**F. precision ggg**

G. funct

105. What operation wound combine 0x80000000 with 0xC0000000 to produce 0x40000000? (89)

A. NOT

**B. XOR ggg**

C. Inclusive OR

D. NAND

E. XAND

**F. Exclusive OR ggg**

G. NOR

106. How many Branch instructions does the MIPS processor recognize? (90)

A. 3

**B. 2**

**C. Two (BNE, BEQ)**

D. 4

E. One (BRN)

F. Four (BNE, BEQ, BEZ, BNZ)

G. Three (BNE, BEQ, BEZ)

107. The BEQ and BNE instructions are (90)

**A. Conditional branches**

B. Unconditional branches ggg

**C. Not jumps**

D. GOTOs

E. Jumps

F. Labels

G. Offsets

108. MIPS conditional branch instructions require the comparison of (90)

A. One register to zero

B. rs and rd

**C. Two registers**

D. rt and rd

**E. rs and rt**

F. A combination of three registers

G. A register and an immediate value

109. The Program Counter contains the address of the next instruction to be executed. Which MIPS instruction would affect the value of the PC? (90)

A. SW

**B. BEQ or BNE**

C. IF

D. LW

E. GO

F. ADDI

**G. J**

110. A conditional branch is appropriate for (92)

A. A subroutine call

B. Shifting a register right or left

**C. A loop**

D. Jumping to a new location.

**E. Checking for a zero denominator.**

F. Transferring control to a different part of the program.

G. Termination of a program

111. A sequence of instructions without branches in the middle is call a(n) (93)

A. Text

B. Subroutine

C. Code segment

D. Module

**E. Basic block**

F. Data

**G. Basic code block**

112. To create all conditions for branches, MIPS uses which set of instructions? (93)

A. beqz, addi, add, sub

**B. The $0 register, slt/slti and bne/beq**

C. blt, bgt, lgbt

D. j, jne, jeq, jal and the $zero register

**E. slt, slti, beq, bne, and the $zero register**

F. blt, bgt, blte, bgte

G. sll, srl, beq, bne, and the $zero register

113. Register $t0 contains 0xFFFFFFFF and register $t1 contains 0x00000000. Which instruction will see $t0 as greater than $t1? (94)

A. salt

B. slti

C. slt $s0, $t1, $t0

**D. sltu $s0, $t0, $t1**

E. slt $s0, $t0, $t1

F. beq $t0, $t1, label

**G. sltu**

114. A good argument for MIPS not having a hardware instruction for "branch on less than" is: (94)

A. The MIPS assembler can do it.

**B. The complexity of the instruction would stretch the clock cycle for many more, less complicated instructions.**

C. It gives students something to think about.

D. It is a pseudo instruction.

E. The space on the silicon to implement the instruction would reduce the size of cache.

**F. It can be produced from two simple instructions**

G. Because of its complexity the instruction would sink more power causing the chip to heat up too much.

115. Which signed byte has the smallest value? (95)

A. 0x7f

B. 0x0f

C. 0x00

D. 0x01

**E. 0x80 ggg**

F. 0x87

**G. 0x81**

116. A jump table in combination with the \_\_\_\_ instruction can be used to implement a case or switch statement. (95)

A. j

**B. jr**

**C. jump register**

D. jeqz

E. beq

F. bne

G. jal

117. Which instruction is commonly used for a subroutine call? (96)

A. bne

B. j

C. beq

D. call

**E. jal**

F. jr $ra

**G. jump and link**

118. The jal instruction is unique because it also (97)

A. Resets the $pc register

B. Operates on four registers

**C. Copies $pc to $ra**

D. Checks the $ra before it branches

**E. Writes to the $ra register**

F. Does not update the program counter

G. Always executes (i.e., it is not conditional)

119. To adjust the stack pointer to make room for three 32-bit values, the correct instruction is (99)

A. addi $sp, $sp, 12

B. addi $sp, $sp, 0x0012

**C. addi $sp, $sp, 0xfff4**

D. addi $sp, $sp, 0xc

E. addi $sp, $sp, 3

F. addi $sp, $sp, -3

**G. addi $sp, $sp, -12**

120. If subroutine A calls subroutine B (100)

**A. Subroutine A must preserve $ra.**

B. Subroutine A is a leaf routine.

C. Subroutine B must preserve $ra.

D. No stack space is needed.

E. There is no convention as long as the code works.

**F. Subroutine A is a non-leaf subroutine.**

G. Neither subroutine needs to preserve $ra; it is automatically saved.

121. In a recursive procedure (101)

A. JAL is never used.

**B. Local storage must be allocated on the stack.**

C. $ra does not have to be preserved.

D. $t register values must be preserved

122. What MIPS registers bound the local variable storage area for a subroutine? (103)

**A. $fp and $sp**

B. $gp and $fp

C. $ra and $sp

**D. The stack pointer and the frame pointer.**

E. $sp and $gp

F. The frame pointer and the global pointer.

G. The stack pointer and the return address.

123. Which register is used in allocation of memory on the heap? (104)

A. $ra

B. $sp

C. The OS pointer.

D. The heap pointer.

**E. $gp**

F. $fp

**G. The global pointer**

124. A disadvantage of recursion is that it can consume a lot of (104)

A. Registers

**B. Stack space**

C. Heap memory

D. Main memory. ggg

**E. Memory on the stack. ggg**

F. Execution cycles

G. Cache memory.

125. Which MIPS registers should not be used by the programmer? (105)

A. $sp

B. $ra

**C. $at**

D. $v0, $v1 ggg

126. The MIPS instruction(s) used to load a 32-bit register with data which is hard-coded into the program is/are (105)

**A. lui, ori**

B. mfhi, mflo

C. addui

D. ld, la

**E. ori, lui**

F. ld32

G. mov

127. Storage and use of decimal numbers for computation in computers is (107)

A. A good idea years ago, but no longer necessary because of cheap memory.

B. Inefficient because numbers are more accurately represented in binary

C. Never used anymore.

**D. Sometimes useful in financial applications.**

**E. More difficult to implement and consumes more memory than needed.**

F. Unpopular because the standards are complicated.

G. Becoming more likely with the advent of multicore processors

128. The difference between ASCII and Unicode is (109)

A. ASCII is used in Java and unicode is used by Microsoft

B. Apple only uses ASCII

C. Unicode represents up to 256 characters, and ASCII represents up to 65536 characters.

D. ASCII is popular in foreign languages and Unicode is not

**E. Unicode can represent more characters than ASCII. ggg**

F. Unicode is used by Java and ASCII is used by Microsoft.

**G. ASCII represents up to 256 characters, and Unicode represents up to 65536 characters. ggg**

129. To move a unicode character from memory to a register, which MIPS instruction should be used? (110)

**A. lhu**

B. lw ggg

C. lh

D. lb

E. lwu

F. lbu

130. Some character sets use 16-bit characters. The would be loaded by which MIPS command? (110)

A. LB

B. LW ggg

**C. LHU**

D. LBU

131. Loading a 32-bit value from immediate data into a MIPS register can be done with which instructions (112)

A. addi, slr

**B. lui, ori**

C. mfhi, mflo

D. l32

**E. lui, ori**

F. lui, addi

G. lui, slt

132. A branch instruction is a(n) \_\_\_\_ instruction (113)

A. P-Type

B. B-Type

C. R-type

**D. I-type**

E. K-Type

F. J-type

133. Which is not a type of MIPS instruction? (113)

**A. B-Type**

B. J-type

**C. S-type**

D. R-type

E. I-type

134. The J-type instruction has an immediate field which is \_\_\_\_ bits long (113)

A. 16

B. 6

C. 25 plus a sign

D. 15 plus a sign

**E. 26 unsigned**

**F. 26**

G. 5

135. The J instruction immediate field is \_\_\_\_\_ bits. (113)

A. 16

B. 28

**C. 26**

D. 15

136. How is immediate data handled in branch instructions (114)

**A. It is shifted left 2 after being sign extended and added to the program counter**

B. It is used to decide whether or not to branch

C. It is sign-extended and subtracted from the program counter

D. It is added to the program counter

**E. It is sign extended, shifted left 2 and added to the program counter**

F. It is sign extended and added to the program counter

G. It replaces the program counter

137. An addressing scheme where the target address is the sum of the program counter and a number specified in an instruction is (114)

A. Immediate addressing.

**B. Program Counter relative addressing**

C. Memory-relative addressing

D. Absolute addressing

E. Register addressing

**F. PC-relative addressing**

G. Memory Indirect addressing.

138. The 16-bit immediate field of a conditional branch instruction can jump through what address range? (115)

A. +- 128K instructions

B. 0-64K instructions.

**C. +-128K bytes**

**D. +-32K instructions**

E. 0 to 256K bytes

F. 0 to 64K bytes

G. +- 32K bytes

139. The branch offset in a MIPS instruction (115)

A. is a 32-bit value.

**B. is the number of instrutions forward or back that hte branch takes.**

C. is 26 bits long.

D. is the number of bytes forward or back that the branch can take.

140. The branch instructions can jump plus or minus 32k instructions. What if we need to branch farther than that? (116)

**A. Branch logic can be combine with the jump instruction to do a longer branch.**

B. It cannot be done if there is a condition attached to the jump.

C. Multiple branch instructions can be used to jump farther.

**D. A jr instruction can be used if the target is fixed.**

E. Write directly to the $pc register.

F. Put the target address into $ra and execute jr $ra.

G. The immediate field can be extended.

141. The MIPS J instruction uses \_\_\_\_\_\_ addressing (116)

A. Immediate

B. Shift right two bits

C. PC-relative

**D. Replace bottom 28 bits of PC**

**E. Pseudodirect**

F. Add bottom 26 bits to PC

G. Register

142. The MIPS instruction set does not include a "branch on less than". Why not? (120)

A. It was originally designed into the circuitry, but a wiring error forced it to be disabled.

**B. It would take too long as a single instruction and stretch the clock cycle for all other instructions**

C. There was not enough room on the chip to implement the circuits needed for the logic.

D. BLT is a pseudo instruction.

**E. It can be done by a combination of simple instructions.**

F. The need for the branch on less than instruction wasn't discovered until after foundational decisions had been made.

G. The Assembler can do that.

143. What best describes what a linker does? (124)

**A. Combines separately compiled code and data modules into a single executable file.**

B. Checks the assembly code against the manufacturer's specification.

**C. Resolves references among separate code modules.**

D. Notifies the programmer of errors in the source code.

E. Compresses executable code size.

F. Calculates load parameters for the program.

G. Ensures instructions are on 4-byte boundaries.

144. In the traditional scheme of compiling and running a program, what is the sequence of actions? (124)

A. Link, assemble, load.

B. Link, assemble.

**C. Compile, assemble, link, load.**

D. Load, Link, Compile.

E. Load, go.

**F. Compile, assemble, link.**

G. Compile, assemble, load, link.

145. An assembly instruction which is not executed in hardware but rather decomposed into a more verbose assembly-language construct by the assembler is called a (125)

**A. Pseudoinstruction ggg**

B. Pseudocode

**C. pseudo instruction**

D. Hardware instruction

E. Macro

F. Artificial instruction

G. Simulated instruction

146. Which describes a symbol table? (125)

A. A table containing the operations codes of all the assembly language instructions.

**B. A table that matches names of labels to the addresses of the memory words that instructions occupy.**

C. A table of all the special characters in the ASCII character set used in the source code.

D. A list of all globally declared data in the program.

E. The list of ASCII characters used in the program code.

**F. A list of needed and supplied program locations.**

G. A table that contains no information.

147. What is the output of a linker? (126)

A. Assembly language instructions

B. A list of all the variables used in a program and their relative addresses

C. Microsoft Intermediate Language.

**D. An executable file.**

**E. The binary executable.**

F. Java bytecode.

G. Modules of code to be strung together and executed

148. Which is not an advantage to using compiled libraries of commonly used code (e.g., a math library)? (126)

A. The library code does not have to be compiled each time it's referenced in a program.

B. It keeps the user from having to write complicated routines (e.g., matrix inversion)

C. It makes it easier to update the program when libraries have been updated.

**D. It makes the executable run faster. ggg**

149. Dynamically Linked Libraries (DLLs) (129)

A. Are assembled by the loader and loaded as a complete unit when the program runs

B. Were invented by IBM in the 1960s.

**C. Are not linked until runtime.**

D. Tend to make programs take more space in memory

**E. Use "Lazy Linking"**

F. Contain bare-bones machine instruction information in executable form

G. Are rarely used today because main memory is cheap.

150. What is the output of a Java compiler (131)

A. machine instructions

**B. A .class file.**

C. A .jar file.0 ggg

**D. Java bytecode**

E. Dynamically linked libraries

F. A .exe file.

G. Assembly instructions

151. What is the input to the Java Virtual Machine (131)

A. Virtual code.

B. Assembly instructions

C. MSIL

**D. Java bytecode**

**E. .class files.0**

F. Java source code

G. Dynamically linked libraries

152. What does the following code do?

lw $t0, 0($s0)

lw $t1, 4($s0)

sw $t0, 4($s0)

sw $t1, 0($s0) (134)

A. Clears memory

B. Ensures proper order.

C. Sorts a list of numbers

D. Nothing

E. Performs an atomic exchange.

**F. Swaps values. ggg**

**G. Exchanges two values ggg**

153. What kind of sort involves successive exchanges of values in an array until they are in order? (135)

A. Card sort

B. Short sort

C. Bucket sort

D. Quick sort

**E. Bubble sort ggg**

F. Speed sort.

154. Which of the following is a pseudoinstruction? (137)

A. add

B. addi

**C. move ggg**

D. lgb

E. j

**F. blt**

G. add

155. If one subroutine calls another, we must at least preserve (138)

**A. The return address register**

**B. The $ra register**

C. The $s registers

D. The $a registers

E. The $sp register

F. The $t registers.

G. The frame pointer.

156. What is the fastest way of saving a variable in the $a0 register (138)

A. Use the $at register.

B. Push it onto the stack

C. Place it into cache.

D. Write it to disk

**E. Copy it to another register**

**F. Add it to $0 and store it in another register.**

G. Save it in memory

157. Inlining saves (138)

A. Space on the heap.

B. variables onto the stack.

**C. Execution time at the expense of code size.**

**D. The overhead of a procedure call**

E. Code size at the expense of execution time.

F. calling parameters in $a registers.

G. Programmers from themselves

158. Machine Code optimized for speed has the fewest instructions (140)

A. T

B. F ggg

159. Two methods of indexing lists of numbers are (141)

A. Tables and arrays ggg

B. serial and parallel

C. in-order and out-of-order

**D. Arrays and pointers**

E. Offsets and shifts. ggg

F. Increments and decrements.

**G. Pointers and arrays.**

160. Which is the fastest way to convert an array index to a memory address location (144)

**A. shift left 2 in hardware ggg**

B. shift left 2 in software

C. multiply by four

D. Shift right in hardware.

**E. Reroute address lines, each to the second one to its left.**

F. Divide by four.

G. successive adds of four

161. While the C language is compiled and linked, the Java language is \_\_\_\_\_\_\_\_\_\_ (145)

A. Run

B. Executed

C. Loaded

**D. Run with the JVM.**

E. Parsed

**F. Compiled and Interpreted ggg**

G. Linked

162. The most popular instruction set for embedded devices is the \_\_\_\_\_ ISA (146)

**A. ARM**

B. Verilog ggg

C. IBM JCL

D. MIPS

E. Motorola

F. Intel 32-bit

**G. Acorn RISC Machine**

163. Which is not a difference between the MIPS and ARM processor (146)

A. Number of register designator bits in instructions.

**B. Use of RISC instructions.**

**C. The year of first product announcement**

D. The number of data addressing modes

E. The number of registers

F. Use of condition codes for branching.

G. The use of the zero register

164. Which operation is unsupported in the ARM-7 instruction set? (146)

A. load halfword signed

**B. Divide**

C. Branch

D. atomic swap

E. compare

**F. div**

G. Shift right and left.

165. The MIPS instruction set has many more data addressing modes than the ARM set. (147)

A. T ggg

**B. F**

166. The MIPS and ARM instruction sets differ widely on how they perform (147)

**A. Compares and conditional branches**

B. Addition.

C. Register addressing.

**D. Memory addressing.**

E. Signed addition.

F. Register-to-register transfers.

G. Multiplication.

167. How many extra instructions does the ARM instruction set require to conditionally skip a single instruction? (147)

A. Three

B. It cannot skip instructions.

**C. None**

**D. Zero**

E. Two

F. One

G. Four.

168. The ARM processor uses a status word of four bits: negative, zero, carry and overflow. What is this status word used for? (147)

A. Performing the divide function.

B. Passing arguments to subroutines.

C. Terminating the program if it crashes.

D. Returning values from subroutines.

E. Arguments for conditional branch instructions.

**F. Deciding whether or not to execute the instruction.**

**G. Conditionally executing instructions.**

169. Each ARM instruction begins with \_\_\_\_\_ (147)

A. An indication of how long it is.

B. The designation for the destination register.

C. A six-bit operation code which can be zero.

D. Bits that tell the length of the instruction.

**E. A four-bit code which determines whether the instruction will execute.**

F. A valid bit.

**G. Condition codes**

170. In the ARM 12-bit immediate field, the eight least significant bits are zero-extended to a 32-bit value, then rotated right the number of bits specified in the first four bits of the field multiplied by two. This allows (148)

A. All 32-bit values to be represented.

B. Faster addition..

**C. The most frequently used 32-bit values to be represented.**

D. All even numbers in a 32-bt word to be represented.

E. Positive and negative numbers.

F. All 24-bit values to be represented.

**G. All powers of 2 in a 32-bit value to be represented.**

171. ARMS processors can load blocks of memory into multiple registers with one instruction (148)

**A. T**

B. F

172. In general, a more complex instruction set means (148)

A. Lower cycles per instruction

B. Easier pipelining.

C. Shorter clock cycles

**D. Lower instruction count**

E. Faster execution time

F. More compact code

**G. Faster compilers.**

173. Intel's instruction set is characterized by (149)

A. Regularity.

B. Reverse engineering.

C. Simplicity.

**D. Backward compatibility.**

174. Integer addition and subtraction use the same add-and-carry logic in the CPU. (178)

A. F

**B. T**

175. Computers write numbers to memory differently than characters (178)

**A. F ggg**

B. T

176. Twos complement is (179)

A. A way to speed up multiplication.

B. A way of adding positive numbers

C. How to figure out the sign of the result.

D. A way of preventing overflow

E. A simplification for long division

**F. A way of representing negative numbers**

**G. A way of changing the sign of a number. ggg**

177. Because binary representation involves only 0's and 1's, carrying in addition is unnecessary. (179)

A. T

**B. F ggg**

178. In binary, 1011 + 1 = (179)

A. 1111

B. 1010

C. 11

**D. 0xC ggg**

**E. 1100 ggg**

F. 0x1010

G. 0x1100

179. In binary, 1011 - 100 = (179)

**A. 111 ggg**

B. 1010

C. 0xC

**D. 0x7 ggg**

E. 11

F. 0xE

G. 1100

180. In a 32-bit register, overflow can occur when adding a positive and negative number (179)

**A. F ggg**

B. T

181. Which does not indicate an overflow? (179)

**A. Subtracting two negative numbers and getting a positive result.**

B. Subtracting a positive number from a negative number and getting a positive result.

C. Subtracting a negative number from a positive number and getting a negative result.

D. Borrowing from the sign bit when subtracting

E. adding two negative numbers and getting a positive result

F. Adding two positive numbers and getting a negative result

**G. Subtracting one positive number from another and getting a negative result. ggg**

182. In binary, 1111...10001 - 1111...11000 = (180)

A. 1111...11000

**B. 1111...11001**

C. 0xffffffff

D. 1000...01001

E. 0000...01000

F. 0xfffffff7

**G. 0xfffffff9**

183. When MIPS detects an overflow condition a(n) \_\_\_\_\_ occurs (180)

A. reboot

**B. exception ggg**

C. branch

D. dump

E. crash

F. halt

**G. interrupt**

184. In which situation does an overflow occur? (180)

A. Adding a positive and negative number and getting a positive result.

B. Adding a positive and negative number and getting a negative result.

C. Adding two negative numbers and getting a negative result.

**D. Adding two positive numbers and getting a negative result. ggg**

185. Processing data for multimedia, especially graphics, has led to (181)

A. Smaller instruction sets.

**B. register partitioning for vector math**

**C. Long registers with carry partitions.**

D. quad-precision floating-point instructions.

E. Processor heat load

F. three-color processors

G. wider cache buses

186. In multimedia arithmetic, an overflow often results in (181)

A. an exception

B. Exceptions.

C. a black screen

**D. saturation**

**E. Achieving a maximum value, but no greater.**

F. Numbers less than zero.

G. a blue screen

187. Twos-complement is a method of (182)

A. A method of preventing overflow.

**B. Converting a number such that its sum with the original number is zero. ggg**

C. A method of detecting overflow.

D. Multiplying numbers.

**E. Expressing negative numbers**

F. Performing addition.

G. Adding logarithms.

188. The EPC register is used for (182)

**A. Storing the PC when an exception occurs.**

**B. Storing the address of the instruction causing overflow**

C. handling environmental protection faults

D. Figuring out if the program will be terminated or not.

E. performing a crash dump

F. Determining the cause of the exception.

G. resuming a halted program

189. After an exception, the EPC register contains (182)

A. The reason why the exception occurred.

B. The location of the crash dump

C. the result of the overflow calculation

**D. the PC when the exception occurred.**

E. the address of the exception handling routine

**F. The address of the instruction where processing may resume.**

G. The system exit instruction.

190. Addition can be speeded up by (182)

**A. Performing add operations in parallel with more hardware.**

B. Lengthening the clock cycle.

**C. anticipating the high-order carry bit**

D. increasing the processor voltage

E. widening the address bus

F. Using floating point numbers.

G. adding smaller numbers ggg

191. Ignoring the sign bit, the length of the product of an m-bit multiplicand and an n-bit multiplier is (183)

**A. up to m + n bits**

**B. The sum of the lengths of the shortest expression of the operands.**

C. 2m bits

D. 32 bits.

E. 2n bits

F. m + n + 1 bits

G. 64 bits.

192. A simple implementation of a multiplier (183)

**A. shifts the multiplicand left for each bit of the multiplier, and adds it to the product if the multiplier bit is 1**

B. Holds the product in a 48-bit register (not including the sign bit)

C. Anticipates the carry at the mid-point

**D. Performs a binary version of elementary school-style multiplication.**

E. Can be pipelined.

F. Calculates the logarithm of the multiplier and multiplicand, adds them and then takes the antilog to form the product

G. Is also the quickest.

193. The most common use of processor arithmetic is for (184)

**A. Determining locations in memory.**

**B. Address calculations**

C. calculating loop variables.

D. floating point multiplies.

E. Calculating colors.

F. graphics calculations. ggg

G. floating-point adds.

194. Multiplying two 32 bit numbers (184)

A. Can overflow a 64-bit register.

B. Cannot overflow a 32-bit register

**C. Can result in a 64-bit product ggg**

D. Must be 64 bits long.

195. Signed multiplication (187)

A. Performs successive subtractions to determine the result

**B. XORs the sign bits to determine the sign of the result.**

C. Is no different fron unsigned multiplication.

**D. forms the product from unsigned numbers and two-complements the product if the multiplicand and multiplier signs are different**

E. Performs right-shifts if either multiplier or multiplicand but not both is negative

F. Cannot cause overflow conditions.

G. Throws an exception if the sign of the multiplier and multiplicand is not the same as the sign of the product

196. Moore's law allows multiplications to be faster because (187)

A. Lower voltages operate with less resistance.

B. Loop unrolling provides adequate parallelism.

**C. More transistors means more adders, up to one per multiplier bit ggg**

D. High clock frequency means shorter clock period.

**E. More hardware means operations can be done in parallel. ggg**

F. Longer instructions can do more.

G. Faster circuits will consume less power.

197. How is multiply overflow handled in MIPS? (188)

A. It triggers an exception.

**B. Software must check the hi half of the product**

C. The EPC register contains the result of the overflow.

**D. It will not occur if a 64-bit result is used.**

E. There is no overflow if the multiplier and multiplicand are both less than 32 bits

F. the sign-bit logic prevents overflow

G. The 32-bit product register is long enough to prevent overflow

198. Faster multipliers (188)

A. Perform shifts and adds sequentially.

B. Can be created with less hardware than a standard multiploer.

**C. Perform more than one operation at once**

D. Work better with smaller numbers.

E. Use much higher clock speeds.

**F. Are easier to implement than faster dividers.**

G. Cannot be pipelined.

199. An invalid mathematical operation which can happen when computers do calculations is (189)

**A. Division by zero ggg**

**B. Arithmetic overflow.**

C. Subtracting minus zero

D. Dividing by 1.

E. Adding zero

F. Multiplying by zero

G. Adding zero to a negative number.

200. The two parts of the integer division result are (189)

A. The dividend and the remainder

B. The dividend and the divisor

C. The quotient and the dividend

**D. Stored in the HI and LO registers.**

**E. The quotient and the remainder**

F. Stored in consecutive floating point registers, e.g., $f0 and $f1.

G. Stored in consecutive integer registers, e.g., $s0 amd $s1.

201. The two operands of the division process are (189)

A. Are stored in the HI and LO registers.

B. The dividend and the remainder

C. Must have the same sign.

D. The quotient and the dividend

**E. Can be stored in conventional registers.**

F. The quotient and the remainder

**G. The dividend and the divisor**

202. A simple and efficient way to implement division is to (189)

A. Look up the answers in a division table.

**B. use hardware to perform "grammar school" division**

**C. Implement long division in hardware.**

D. perform successive subtractions at clock speed.

E. Use transistors to add bits.

F. Run multiply hardware with reversed voltages.

G. Group bits into fours and perform hexadecimal division.

203. Whereas a simple multiplication algorithm is based on read a bit, conditionally add and shift, a simple division is based on (190)

A. Take the log2 of the divisor and shift left that many bits.

**B. Checking to see if the result of a subtraction is positive, and setting a bit if so.**

**C. Subtract and shift and conditionally write a bit**

D. Judicious placements of ones and zeros in the quotient.

E. Successive subtractions without shifting.

F. Add and shift the opposite direction.

G. Take the log2 of the divisor and shift right that many bits.

204. Signed division can be done by (192)

A. Doing unsigned division and setting the sign of the remainder based on sign of the dividend

**B. Doing unsigned division and setting the signs of the quotient and remainder based on signs of the operands**

C. Checking for overflow.

D. Doing unsigned division and setting the sign of the quotient based on sign of the divisor

**E. XORing the sign bits of the operands and calculating the signs of the remainders.**

F. Doing unsigned division, totally disregarding signs

G. Transistors which have three states.

205. The sign of an integer divide is (192)

**A. The XOR of the dividend and divisor signs.**

B. The AND of the dividend and divisor signs.

C. The OR of the dividend and divisor signs.

D. The NOR of the dividend and divisor signs.

206. In determining the sign of the quotient, what rule must always hold? (193)

**A. Remainder = Dividend - Quotient \* Divisor**

B. sign of Remainder = sign of divisor

C. The sign bit of the remainder is the XOR of the operands' sign bits.

D. Remainder = Quotient \* Divisor - Dividend

**E. Dividend = Quotient \* Divisor + Remainder**

F. The remainder is always positive.

G. sign of Remainder = sign of quotient

207. Using many adders to speed up division (194)

**A. Is limited in its effectiveness because there are branches in the logic.**

**B. Won't work because decisions have to be made after the subtractions.**

C. works because of Moore's law.

D. Is difficult because the result is unknown.

E. won't work because you also have to calculate the remainder.

F. Works because fast multiplies can check the answer.

G. Is difficult because division involves subtraction.

208. If by mistake you divide a positive number by a negative number using unsigned divide (divu) (194)

**A. The remainder is the dividend.**

B. The result will overflow

C. An exception occurs

D. The quotient is zero and the remainder is the divisor

**E. The quotient is zero.**

F. The remainder will be zero.

G. The remainder and the quotient are interchanged.

209. One way of making the division process faster is to (194)

A. Increase the power supply voltage.

B. Produce multiple results and pick the one that is closest to the real answer.

**C. Looking up partial results in a table rather than use combinational logic to obtain them.**

D. Use combinational logic to obtain results rather than looking them up n a table.

**E. Guess intermediate results and make changes if the guesses are wrong.**

F. Guess the result if being wrong doesn't change the answer.

G. Add cache memory.

210. Binary representations of floating point numbers have greater range but less precision than binary representations of integers (196)

A. F

**B. T**

211. Normalized scientific notation (196)

**A. In binary means that there is only a 1 to the left of the binary point.**

B. Applies to binary but not decimal numbers.

C. Represents any real number with infinite precision.

**D. Means the number left of the point is not zero.**

E. Makes it difficult to multiply numbers.

F. Makes sense only in the decimal system.

G. Is widely used in integer calculations.

212. A floating point number is composed of (197)

A. Exponent, Fraction, Logarithm

**B. One bit for the sign, 8 for the exponent and 23 for the fraction.**

C. Sign, Extent

**D. Sign, Exponent, Fraction**

E. 16 bits for the signed numerator, 16 bits for the denominator.

F. Sign, Characteristic, Mantissa

G. The logarithm of the number.

213. In single-precision binary floating point notation, the exponent is \_\_\_\_\_\_ bits long (198)

**A. Eight**

**B. 8**

C. Five

D. 11

E. Ten

F. 23

G. 52

214. In single-precision binary floating point notation, the fraction is \_\_\_\_\_\_ bits long (198)

**A. 23**

B. Sixteen

**C. Twenty-three**

D. 52

E. 11

F. Nine

G. 8

215. The standard for floating point notation is (198)

**A. IEEE-754**

B. ISO-9000

C. NFPA-1002

D. ASA-400

216. the NaN notion can result from (199)

A. Adding a large positive and negative numbers

B. Incorrect use of associativity.

C. Multiplication by zero.0

D. Subtracting two numbers which are nearly the same

**E. Performing an operation with a NaN value.**

F. A sign miscompare

**G. Divide by zero**

217. Negative exponents in floating point binary are represented as twos-complement numbers (200)

A. T

**B. F**

218. The length of a single-precision floating point binary number is \_\_\_\_ bits (200)

A. 16

B. 128

**C. 32**

D. 64

219. The length of a double-precision floating point binary number is \_\_\_\_ bits (200)

**A. 64**

B. 128

C. 16

D. 32

220. which number below is the hex representation of a negative floating point number? (202)

**A. 0x80100000**

**B. 0xC0100000**

C. 0x0C010000

D. 0x40200000

E. 0x10800000

F. 0x08010000

G. 0x20400000

221. Adjusting exponents and adding fractional parts describes (202)

**A. Floating point addition.**

B. Integer addition.

C. Floating point multiplication.

D. Integer multiplication.

222. Which is (are) the most complex operation(s) for floating point calculations in the list below? (203)

**A. Addition**

B. Rounding

**C. Subtraction**

D. Multiplication.

E. Division

F. Comparison with zero

G. magnitude sorting

223. Another name for the fractional part of a floating point number is the (204)

A. Denominator.

B. Numerator.

C. Exponent.

**D. Significand.**

E. Reduction.

F. Precision.

224. The first step in adding two floating point numbers is (205)

**A. Make the exponents equal**

**B. Adjust the numbers so the exponents are the same.**

C. Express the numbers in binary fractions.

D. Make the fractional parts equal.

E. Convert negative numbers to twos-complement notation.

F. Compare the signs.

G. Make the numbers the same length.

225. Normalization is (205)

**A. Adjusting the exponent so that the number is >=1 and < 2.**

B. The same as rounding.

**C. Ensuring the binary scientific notation has only a 1 left of the binary point.**

D. Converting the number from binary to hexadecimal.

E. Setting the exponents so they are equal.

F. Making the fractional part equal to zero.

G. Making the exponent equal to zero.

226. When binary floating point numbers are multiplied, what in the following list does not happen? (206)

A. Exponents are added.

B. Signs are XORed.

C. Fractional parts are multiplied.

**D. Twos-complementation is performed.**

E. The result is normalized.

**F. Exponents are made equal.**

G. The result is rounded.

227. Which is not a step in multiplying two floating point numbers? (209)

A. Multiplying the fractional parts

B. Rounding

C. Normalizing

**D. Aligning the exponents**

228. If a double-precision floating point number is loaded into register $f2, what other register is used (211)

A. $f1

B. $0

C. None

**D. $f3**

E. $f0

F. $s3

229. MIPS supports single-precision floating point, but not double-precision floating point (211)

A. T

**B. F**

230. In MIPS Assembly Language, which is the instruction for a single-precision operation? (212)

A. adds

B. mul

**C. mul.s**

D. smul

E. lw

**F. div.s**

G. muls

231. Double precision floating point numbers may not accurately represent decimal numbers (218)

**A. T**

B. F

232. The big difference between the IEEE-754 representation of 32-bit and 64-bit floating point numbers is (218)

**A. The precision of the numbers**

B. The size of the exponent.

**C. The number of bits allocated to the fractional part.**

D. The speed at which computations are performed.

E. The magnitude of the numbers

F. The length of the numbers

G. The range of the numbers

233. Which is not a length of floating point data? (221)

A. 64 bits

B. 16 bits

C. 32 bits

**D. 8 bits**

234. Subword parallelism is another name for what happens in (222)

A. Unsigned addition.

B. Executing half-word load/store instructions.

**C. SIMD**

D. Floating point addition

E. Simultaneous Multi-Threading (SMT)

F. Floating point multiply

**G. Single Instruction Multiple data**

235. Enabling a 128-bit register to perform four addition operations on four 32-bit values at one is done by (222)

A. Increasing cache size.

B. Lengthening the word length.

**C. Stopping the carry bit on 32-bit boundaries.**

D. Widening the memory bus.

E. Decreasing the cycle time.

**F. Carry partitions.**

G. Putting more transistors on a chip.

236. ARM's NEON extensions (223)

A. Can perform 256-bit adds

B. Allow registers to be viewed as 32 32-byte registers or 64 16-byte registers

C. Support 64-bit floating point numbers.

**D. Provide multimedia support.**

E. Support quad-precision calculations.

F. Provide simultaneous multithreading support

**G. Allow registers to be viewed as both 32 8-byte registers and 16 16-byte registers.**

237. Which is not a feature of the MIPS processor? (244)

**A. Complex instructions are longer (more bytes) than simpler instructions**

B. The set of arithmetic instructions are simple building-blocks

C. The only memory accesses are the load and store instructions.

D. There are only two conditional branch instructions

238. The LW instruction requires that \_\_\_ register(s) be read. (245)

A. Two

B. Three

**C. One**

D. No

239. Which is not an element of the MIPS CPU (246)

**A. Flash memory**

B. Program Counter

C. Data Memory

D. Instruction Memory

E. Arithmetic Logic Unit

**F. Interrupt handler**

G. Register File

240. The Arithmetic Logic Unit is not used for which set of instructions (246)

A. Load (LW, LH, LB)

**B. J-types**

C. Branch (BEQ, BNE)

**D. Jump (J)**

E. R-types

F. Store (SW, SH, SB)

G. I-types

241. MUX is the abbreviation for which device? (247)

A. Memory access

**B. Multiplexer**

C. Multicrossfeed

D. Multiplier

E. Memory storage

F. Cache blocks

242. What happens in edge-triggered clocking? (249)

A. Instruction execution determines when the clock pulse starts

B. Reads and writes occur during the clock pulse

C. The clock pulse starts on the falling edge

**D. Writes occur on the rising edge of the clock pulse**

**E. State is recorded on the rising edge.**

F. COmbination logic starts before the pulse.

G. Combinational logic continues after the pulse.

243. When does combinational logic happen? (249)

A. After Instruction Fetch and before Instruction Decode

B. Between the falling edges of two adjacent clock pulses

C. During Data Write

D. Between adjacent cycles

**E. After the state is sampled and before the state is stored**

F. During instruction Fetch

**G. Between the rising edges of two adjacent clock pulses**

244. If a state element is not read on every clock edge, (250)

A. It cannot be read.

B. A multiplexor must be used.

C. It must be written to main memory.

**D. It requires a control signal to be read.**

245. Adding sign-extended, shifted-left-2 immediate data to the program counter is done in the \_\_\_\_\_\_\_ instruction. (252)

A. J

B. ADDI

C. JAL

**D. BEQ**

246. To add two numbers, you would use what device? (252)

**A. Arithmetic Logic Unit**

**B. ALU**

C. MUX

D. Branch control

E. Control Unit

F. Sign-extender

G. Cache

247. Which device determines which instruction to read next? (253)

**A. The program counter**

B. The Write-back logic

**C. The PC**

D. The control unit.

E. Memory Access

F. The ALU

G. The instruction Multiplexer

248. How many five-bit inputs does the Register File have (253)

A. Five

B. One

**C. Three**

D. Two

E. Non

F. Four

249. Which is not used in the execution of a BEW instruction? (254)

A. Register file.

B. ALU.

**C. Data memory.**

D. Sign extender.

250. Which action does a MIPS branch instruction not do? (254)

**A. Write to the register file.**

B. Subtract the contents of one register from another.

C. Shift the immediate data left two bits.

D. Sign extend the immediate data

251. Which device accepts 16-bit immediate data and produces a 32-bit output? (255)

**A. Sign Extender**

B. The control unit0

C. The hazard detector

D. Program Counter

E. Arithmetic Logic Unit

F. Register file

252. What happens to the 16-bit sign-extended offset of a branch instruction before it is added to the program counter contents? (256)

**A. It is shifted left two bits**

B. it is added to the program counter.

C. It depends on the result of the compare

D. It is shifted right two bits

E. it is set to zero

F. It replaces the program counter.

253. The ALU zero output is used in handling which instruction? (256)

A. JAL

B. ADD

C. SLT

**D. BNE**

254. How many registers does a branch instruction read from the register file? (258)

A. Three

B. One

C. None

D. Five

**E. Two**

F. Four

255. How many registers does an R-type instruction read from the register file? (258)

A. One

B. None

C. Four

D. Three

E. Five

**F. Two**

256. In the branch instructions, the number of instructions to skip is given in the immediate field. How long is this field? (262)

A. 15 bits

B. 18 bits

C. 26 bits

D. 5 bits

**E. 16 bits**

F. 17 bits

257. Which instruction calculates an address with the ALU and then uses that address in main memory? (265)

A. BEQ

B. SLT

**C. LW**

D. J

E. ADDI

**F. SW**

258. Which instruction has a 26-bit immediate field? (270)

**A. J**

B. BEQ

C. SW

D. ADDI

E. ADD

F. LW

259. The settings of the multiplexors is how each instruction is processed individually. What unit controls most of the mux settings? (271)

A. The instruction fetch unit

**B. The control unit.**

C. The program counter

D. The register file

E. The ALU

F. Instruction memory

260. Instructions which execute in a single clock cycle execute faster than instructions requiring multiple clock cycles (272)

A. T

**B. F**

261. In pipelining, which statement is not true? (273)

**A. You need duplicate hardware to execute multiple steps at the same time**

B. Pipelining is not without hazards.

C. the overall process must be broken into steps

D. The speedup of the process (vs non pipelined) is roughly equal to the number of steps

E. The steps must be independent so that they can be executed simultaneously

F. Excessive speculation leads to wasted energy consumption.

**G. The simplest CPU designs have the greatest throughput.**

262. Doing laundry in four equally long stages allows the process to be sped up by a factor of \_\_\_\_ over repeatedly doing the process end-to-end. (273)

A. 1.75

B. Eight

C. Two

D. Seven

**E. Four**

263. The fact that all instructions must be fetched, many instructions require register access and ALU operations led to the idea of (274)

A. Data Level Parallelism

**B. Instruction Level Parallelism**

C. Thread Level Parallelism

D. Request Level Parallelism

264. A single-cycle instruction spends 100, 200, 150, 250 and 200 pico seconds in the IF, ID, EX, MEM and WB stages of the processor. If the processor is pipelined, how long should the clock cycle be? (275)

A. 100 ps

B. 200 ps

**C. 250 ps**

D. 900 ps

E. 180 ps

F. 300 ps

265. What determines the clock cycle time of a pipelined, multi-stage instruction? (275)

A. The number of instructions executing at once.

**B. The stage which takes the longest.**

C. Market requirements.

D. The duration of the shortest stage

E. The duration of the first stage

F. Total execution time divided by number of stages

**G. The duration of the longest stage**

266. Which is not a MIPS pipeline stage? (275)

A. Instruction Decode

B. Memory Access

C. Write-back

**D. Data flush.**

**E. Instruction write**

F. Instruction decode

G. Execute

267. A major effect of pipelining is (276)

A. Shorter execution time for single instructions.

B. Smaller CPI

C. Larger CPI

D. Decreased latency.

**E. a shorter clock cycle.**

**F. Greater instruction throughput.**

G. reduced instruction count

268. Which kind of hazard occurs when two pipeline stages require the same hardware, and the hardware is in use? (278)

**A. Structural**

B. Intentional

C. Exception

D. Control

E. Accidental

F. Data

269. If, in two adjacent instructions, one produces data needed by the following instruction, what type of hazard can occur? (279)

A. Structural

**B. Data**

C. Accidental

D. Control

E. Intentional

F. Exception

270. If a data hazard occurs and there is no way to supply the data an instruction needs to execute, what happens (280)

A. A structural hazard occurs.

**B. Execution is delayed with stalls.**

C. An exception occurs

D. A control hazard occurs

**E. a pipeline stall occurs.**

F. An interrupt occurs.

G. Data hazard occurs

271. What is the MIPS solution to a data hazard? (280)

A. Executing a page fault

B. Rewriting the program to eliminate the hazard.

C. Throwing a tantrum.

D. Inserting a delay slot instruction

**E. Setting ALU input muxes to obtain data from earlier instructions.**

F. Throwing an exception

**G. Forwarding**

272. The Rd of an R-type instruction can be forwarded to an instruction (281)

A. One cycle previous

B. Two cycles previous

**C. One cycle later**

D. In the same cycle

273. The extra time needed to calculate the result of a branch condition creates a(n) (282)

A. In-line exception

B. Instruction NOPs

**C. Control hazard**

D. Data hazard

E. structural hazard

F. Pipeline flush.

274. Guessing which branch to take is an acceptable way of resolving control hazards (284)

**A. T**

B. F

275. What is the correct order of MIPS pipeline stages? (287)

A. IF-ID-MEM-WB-EX

B. MEM-ID-IF-EX-WB

C. ID-EX-MEM-WB-IF

D. EX-MEM-WB-IF-ID

E. ID-IF-EX-WB-MEM

**F. IF-ID-EX-MEM-WB**

276. The writeback mux appears in stage 5 of the MIPS processor pipeline. What device is the destination of the writeback mux output? (287)

A. The branch adder

**B. The PC**

C. Instruction memory

D. The branch target mux

E. Data Memory

**F. The register file**

G. The ALU

277. Pipelined circuitry requires (289)

A. More ALUs

B. A writeback mux with more than 2 inputs.

C. Sign extenders.

**D. Pipeline registers.**

278. In which pipeline stage is the ALU found? (296)

A. IF

**B. EX**

C. ID

D. Hazard Detection

E. Control

F. WB

**G. Execute**

279. The primary decoding of what an instruction does is done by (302)

**A. The control unit.**

B. The ALU.

C. The register file.

D. Instruction memory.

280. What is a method of resolving data hazards? (303)

**A. Forwarding**

**B. Stalling**

C. Caching

D. Paging

E. Reading and Writing

F. Throwing an exception.

G. Starting and Stopping

281. When during a clock cycle can data be written to registers? (305)

A. During the second half of the clock cycle.

B. Anywhere, depending on the timing of the strobe.

C. The falling edge of the clock cycle.

**D. Right after the rising edge of the clock pulse.**

E. Both on the rising edge and the falling edge of the clock pulse.

**F. During the first half of the clock cycle.**

G. Anywhere in the middle.

282. When during a clock cycle can data be read from registers? (305)

**A. Right after it is written.**

B. Both on the rising edge and the falling edge of the clock pulse.

C. The falling edge of the clock cycle.

**D. During the second half of the clock cycle.**

E. During the first half of the clock cycle.

F. Anywhere, depending on the timing of the strobe.

G. Anywhere in the middle.

283. Which pipeline register usually provides data for the ALU? (306)

A. IF/ID

B. EX/WB

C. EM/MEM

D. IF/EX

E. MEM/WB

**F. ID/EX**

284. How are data hazards detected? (306)

A. Comparing the destination register of the current instruction with the source registers of previous instructions.

**B. Comparing the source registers for the current instruction with destination registers of previous instructions.**

**C. Using instruction-specific information from the pipeline registers.**

D. Determining whether or not there has been a cache miss or page fault.

E. By looking for page faults.

F. By counting the number of cycles since execution began.

G. Determining whether or not preceding instructions have executed correctly.

285. It is possible that the ID/EX register will contain values needed by the ALU in the current cycle, but which are scheduled to be written back to the registers several cycles from now. (309)

A. F

**B. T**

286. What do the Forward A and Forward B control signals do? (309)

**A. Forwarding in the case of a data hazard.**

B. Control the ALU

C. Tell when the next instruction should be read.

D. Stalls instructions if there is a data hazard.

E. Determine whether or not a conditional branch instruction takes the branch.

F. Determine if it is a load or store instruction.

**G. Tell the muxes on the inputs to the ALU which inputs should be selected.**

287. Once a data hazard is detected, the MIPS processor typically resolves it by (310)

A. Loading the program counter.

B. Stalling the instruction

**C. Setting the input muxes to the ALU**

D. Causing the instruction to be No-oped

E. Deasserting all control lines in that stage.

F. Storing variables on the stack.

**G. Asserting the Forward A or Forward B lines.**

288. The forwarding unit moves data backwards in the pipeline (312)

A. T

**B. F**

289. If a register is written during a cycle, its contents are not available to be read until the following cycle (313)

A. T

**B. F**

290. An instruction is converted to a NOP by (314)

A. Incrementing the program counter

**B. Setting all control lines to zero.**

C. Jumping forward one instruction

D. Replacing the instruction op code with zeroes.

291. An instruction is stalled by (315)

A. Zeroing the $pc for one cycle

**B. Making the instruction a NOP**

C. Inserting a NOP after the instruction

D. Suppressing the clock pulse

292. What method is not used by the MIPS processor to speed branch handling? (318)

A. Using an extra adder to compute branch targets quicker.

**B. Avoiding the use of branch instructions**

C. Using a comparator to determine branch outcomes more quickly.

D. Predicting branch outcomes.

293. In a pipelined processor (320)

A. Data hazards are automatically taken care of.

B. The speedup is indicated by the number of instructions which can be issued in the same cycle.

**C. Pipeline registers isolate the stages.**

D. More than one instruction can be in a stage at one time.

294. Branch predictors should predict that the branch at the end of a loop is rarely taken (321)

A. T

**B. F**

295. A two-bit predictor has \_\_\_\_ states. (322)

A. One

B. Two

**C. Four**

D. Eight

296. What is true about a 2-bit branch predictor? (322)

**A. It has four states.**

B. It is always more accurate than a one-bit predictor.

C. It works best in integer applications.

D. It can change its prediction if it is wrong one time.

297. As pipelines become deeper (323)

A. Exception handling becomes simpler

**B. Branch instructions have more branch delay slots**

C. Branch prediction becomes more accurate

D. Branch instructions can branch farther

298. What kind of instruction is ideally put in the delay slot (323)

**A. An instruction whose execution is independent of the branch being taken or not.**

B. R-type instructions.

C. An instruction which would be executed only if the branch is taken.

D. An instruction which would be executed only if the branch is not taken.

E. It doesn't matter.

**F. In instruction which is independent of the branch.**

G. Any instruction will do, as long as it can be no-oped.

299. It is acceptable to predict not only whether or not the branch is taken, but also the branch target if the branch is taken. (324)

**A. T**

B. F

300. The origin of an interrupt is (326)

A. Spontaneous.

B. Unanticipatable.

C. Called an exception

**D. usually a normal system event**

E. In memory

F. Internal to the processor

**G. Something whose occurrance can be anticipated.**

301. Which register is used in conjunction with the Exception Program Counter (EPC) to handle MIPS exceptions? (327)

A. $zero

B. The $v registers

C. the return address register

D. The $a registers

**E. The cause register**

F. The program counter

302. If an exception occurs, the program must be terminated (327)

A. T

**B. F**

303. What prevents an instruction that causes an exception from erroneously writing back to the register file? (328)

**A. It is No-oped before it gets to the WB stage.**

B. All register writes are inhibited while the exception is being handled.

C. Hazard logic.

D. Instructions which write back to the registers cannot cause exceptions.

E. A rollback command is issued to the register file.

F. Page faults.

**G. Setting WB control lines to zero.**

304. What characterizes imprecise exception handling? (331)

A. Truncation error.

B. The exception handling routine does not know what to do.

C. Knowing which instruction must be restarted.

D. The use of non-pipelined architectures with pipelined instructions

E. Roundoff error.

**F. Allowing some instructions begun after the one causing the exception to complete.**

**G. The inability to associate the cause of the exception with a specific instruction**

305. Which activity would cause an exception in the earliest pipeline stage? (332)

**A. Undefined op code**

B. page fault

C. Data hazard.

D. Writing to read-ony memory.

**E. Unaligned instruction fetch.**

F. Divide by zero

G. Arithmetic overflow

306. What kind of parallelism does pipelining support? (332)

A. Control level parallelism.

B. Data Level Parallelism

C. Request Level Parallelism

**D. ILP**

**E. Instruction Level Parallelism**

F. Flow parallelism.

G. Thread Level Parallelism

307. Speculative Execution is (333)

A. Energy efficient.

B. Not needed in superscalar processors.

C. A method of achieving high performance in pipelined architectures.

**D. Irreversible.**

E. A way to use CPU hardware effectively.

**F. One response to control hazards.**

G. A system of justice in third-world countries.

308. Static multiple issue is done by the (334)

A. Instruction cache

B. Data cache

C. Loader

D. Programmer

**E. Compiler**

F. Processor

309. To be effective, multiple issue typically requires (336)

A. Programmer awareness.

B. Larger main memory

**C. More hardware.**

D. Larger heat sinks.

310. If, in a multiple-issue processor, we wish to issue an ALU and a memory instruction in the same cycle, (336)

A. memory must be dual-ported.

**B. the register file must allow up to four registers to be read in the ID stage.**

C. the power supply must be substantially larger.

D. no extra hardware is needed.

311. Using memory offsets in load and store instructions, multiple registers for sequential loads and significantly reducing the number of end-of-loop branch instructions characterizes (338)

**A. Loop unrolling**

B. Compiling the code backwards

C. Having more registers available

D. Issuing multiple instruction per clock

312. Which is a common way of exposing parallelism in computer code? (338)

A. Register renaming

B. Branch Prediction

**C. Loop unrolling.**

D. Data matching

313. Dynamic pipeline scheduling is done by (339)

A. Oil companies.

**B. The processor**

C. The programmer

D. The compiler

314. A processor that can issue more than one instruction per clock cycle is (339)

A. Dynamite.

B. Dynamic.

**C. Superscalar.**

D. Super sized.

315. Which of the following are not used in dynamic scheduling? (340)

**A. Crossbar architecture**

B. reorder buffer

C. Commit unit

D. Reservation station

316. Out of order execution happens when (341)

A. The reorder buffer is empty

B. The reservation stations are full

**C. All data is available for a subsequent instruction**

D. The Supreme Court intervenes

317. Which of the following makes it safe to do out-of-order execution? (341)

A. Cache snooping

B. dual-ported memory

C. The reservation station

**D. The commit unit**

318. What decreases with more advanced multiple-issue strategies? (343)

A. Throughput

B. Power supply voltage

C. Power consumption

**D. Energy efficiency**

Answers

1. A

2. C

3. D

4. B

5. D

6. C

7. A

8. C

9. A, G

10. A, G

11. D

12. D

13. B

14. A

15. C

16. A

17. B

18. A, B

19. B

20. A

21. B, F

22. B, C

23. A

24. C

25. D

26. A

27. A

28. D

29. D

30. D

31. A

32. A

33. A

34. D, G

35. B, F

36. B

37. D

38. A

39. A

40. C

41. C

42. D

43. A

44. A

45. A

46. A

47. D

48. C, E

49. B

50. A

51. B

52. B

53. B

54. B

55. A

56. A

57. A

58. C

59. D

60. C

61. B

62. A

63. B

64. B

65. D

66. C

67. B, E

68. B

69. B

70. B

71. B, D

72. B, C

73. B

74. C

75. A, B

76. B, C

77. B, F

78. C, E

79. C, D

80. D, E

81. E, G

82. B, F

83. A, D

84. B, F

85. B, C

86. C, G

87. C, D

88. B

89. B

90. D, E

91. B, E

92. E

93. A, D

94. C, G

95. C, F

96. A, B

97. A, B

98. C

99. B, D

100. A, B

101. B, C

102. C, G

103. B, G

104. E, F

105. B, F

106. B, C

107. A, C

108. C, E

109. B, G

110. C, E

111. E, G

112. B, E

113. D, G

114. B, F

115. E, G

116. B, C

117. E, G

118. C, E

119. C, G

120. A, F

121. B

122. A, D

123. E, G

124. B, E

125. C

126. A, E

127. D, E

128. E, G

129. A

130. C

131. B, E

132. D

133. A, C

134. E, F

135. C

136. A, E

137. B, F

138. C, D

139. B

140. A, D

141. D, E

142. B, E

143. A, C

144. C, F

145. A, C

146. B, F

147. D, E

148. D

149. C, E

150. B, D

151. D, E

152. F, G

153. E

154. C, F

155. A, B

156. E, F

157. C, D

158. B

159. D, G

160. A, E

161. D, F

162. A, G

163. B, C

164. B, F

165. B

166. A, D

167. C, D

168. F, G

169. E, G

170. C, G

171. A

172. D, F

173. D

174. B

175. A

176. F, G

177. B

178. D, E

179. A, D

180. A

181. A, G

182. B, G

183. B, G

184. D

185. B, C

186. D, E

187. B, E

188. A, B

189. D, F

190. A, C

191. A, B

192. A, D

193. A, B

194. C

195. B, D

196. C, E

197. B, D

198. C, F

199. A, B

200. D, E

201. E, G

202. B, C

203. B, C

204. B, E

205. A

206. A, E

207. A, B

208. A, E

209. C, E

210. B

211. A, D

212. B, D

213. A, B

214. A, C

215. A

216. E, G

217. B

218. C

219. A

220. A, B

221. A

222. A, C

223. D

224. A, B

225. A, C

226. D, F

227. D

228. D

229. B

230. C, F

231. A

232. A, C

233. D

234. C, G

235. C, F

236. D, G

237. A

238. C

239. A, F

240. B, D

241. B

242. D, E

243. E, G

244. D

245. D

246. A, B

247. A, C

248. C

249. C

250. A

251. A

252. A

253. D

254. E

255. F

256. E

257. C, F

258. A

259. B

260. B

261. A, G

262. E

263. B

264. C

265. B, G

266. D, E

267. E, F

268. A

269. B

270. B, E

271. E, G

272. C

273. C

274. A

275. F

276. B, F

277. D

278. B, G

279. A

280. A, B

281. D, F

282. A, D

283. F

284. B, C

285. B

286. A, G

287. C, G

288. B

289. B

290. B

291. B

292. B

293. C

294. B

295. C

296. A

297. B

298. A, F

299. A

300. D, G

301. E

302. B

303. A, G

304. F, G

305. A, E

306. D, E

307. C, F

308. E

309. C

310. B

311. A

312. C

313. B

314. C

315. A

316. C

317. D

318. D