Convert the following string into its ASCII hex representation.
Don't use 0x or h with the hex values. The hex for "1+z" is 31 2B 7A

Google

47 6F 6F 67 6C 65

Convert the following string into its ASCII hex representation.
Don't use 0x or h with the hex values. The hex for "1+z" is 31 2B 7A

Electric

45 6C 65 63 74 72 69 63

Convert the following ASCII hex representation into its string.
The hex 31 2B 7A is the string 1+z

53 74 75 64 69 6F

Studio

Convert the following ASCII hex representation into its string.
The hex 31 2B 7A is the string 1+z

4D 69 63 72 6F 73 6F 66 74

Microsoft

Convert the following binary number into a signed decimal value.

1111 1010 0000 1111

-1521

Convert the following binary number into a signed decimal value.

1010 1010 0101 1111

-21921

Convert the following binary number into an unsigned decimal value.

1100 1000 0100 0001

51265

Convert the following binary number into an unsigned decimal value.

1110 0001 0010 1000

57640

The following data segment starts at memory address 0x1000 (hexadecimal)

```
.data
printString BYTE "Assembly is fun",0
```

```
moreBytes BYTE 10 DUP(0)
dateIssued DWORD ?
dueDate DWORD ?
elapsedTime WORD ?
```

What is the hexadecimal address of dueDate?

- 0x101A
- 0x1030
- 0x1010
- 0x101E

The following data segment starts at memory address 0x2200 (hexadecimal)

```
.data
printString BYTE "MASM is fun",0
moreBytes BYTE 25 DUP(0)
dateIssued DWORD ?
dueDate DWORD ?
elapsedTime WORD ?
```

What is the hexadecimal address of dueDate?

- 0x2225
- 0x220C
- 0x2229
- 0x2241

The following data segment starts at memory address 0x4100 (hexadecimal)

```
.data
printString BYTE "Do not add decimal to hex",0
someBytes WORD 36 DUP(0)
moreBytes BYTE 10, 20, 30, 40, 50, 60, 70, 80, 90
questionAddr DWORD ?
ignoreMe WORD ?
```

What is the hexadecimal address of questionAddr?

- 0x411A
- 0x416B

- 0x4162
- 0x4207

After the following MASM code is executed:

```
mov    eax,52
mov    ebx,17
mov    ecx,23
add    eax,ebx
sub    eax,ecx
```

What is the value in the eax register (in decimal)? 46

What is the value in the ebx register (in decimal)? 17

What is the value in the ecx register (in decimal)? 23

After the following MASM code is executed:

```
mov    eax,212
mov    ebx,19
mov    edx,0
div    ebx
```

What is the value in the eax register (in decimal)? 11

What is the value in the ebx register (in decimal)? 19

What is the value in the edx register (in decimal)? 3

Suppose that result is declared as DWORD, and the following MASM code is executed:

```
   mov eax,7
   mov ebx,5
   mov ecx,6
label5:
   add eax,ebx
   add ebx,2
   loop label5
   mov result,eax
```

What is the value stored in the memory location named result?

67

Please place the following steps of the instruction execution cycle in their proper order.

Step 1:

Fetch the instruction at the address in the Instruction Pointer into the Instruction Register.

Step 2:

Increment the Instruction Pointer to point to next instruction.

Step 3:

Decode the instruction in the Instruction Register.

Step 4:

If the instruction requires memory access, determine the memory address, and fetch the operand from memory into a CPU register, or send the ope

Step 5:

Execute the instruction.

Step 6:

If the output operand is in memory, the control unit uses a write operation to store the data.

Select the pseudo-code that most closely corresponds to the following assembly code. Assume that the variables a, b, c, and d are initialized elsewhere in the program.

```
.data
; General purpose variables
a       DWORD     ?
b       DWORD     ?
c       BYTE      ?
d       BYTE      ?
upperLevel   DWORD     18
lowerLevel   DWORD      3
; Strings
yes     BYTE     "Yes",0
no      BYTE     "No",0
maybe   BYTE     "Maybe",0

.code
   main PROC
   mov   eax, 0
   mov   ebx, a
startLoop:
   cmp   eax, ebx
   jge   endOfProgram
```

```
   mov   edx, OFFSET no
   call  WriteString
   inc   eax
   jmp   startLoop
   mov   edx, OFFSET maybe
   call  WriteString
endOfProgram:
   exit
main ENDP
END main
```

- ```
  if (a < b)
     print (no);
  else
     print (maybe);
  ```

- ```
  while (a < 18)
     print (no);
  else
     print (maybe);
  ```

- ```
  for (k = 0; k < a; k++)
     print (no);
  ```

- ```
  while (a > 0)
     print (no);
  ```

A common programming error is to inadvertently initialize ECX to zero before beginning a loop (when using the LOOP instruction).

- True
- False

The MOVZX instruction is only used with unsigned integers.

- True
- False

Which of the following is **NOT** a valid MOV operation? Table 4-1 might be helpful. (check any/all that apply)

- [ ] MOV mem,reg
- [x] MOV mem,mem

- [ ] MOV mem,imm
- [ ] MOV reg,imm
- [x] MOV imm,imm
- [ ] MOV reg,reg
- [ ] MOV reg,mem
- [x] MOV imm,mem

Which of the following are valid uses of the XCHG instruction? (check any/all that apply)

- [x] XCHG reg,reg
- [ ] XCHG mem,mem
- [x] XCHG mem,reg
- [ ] XCHG imm,imm
- [ ] XCHG imm,reg
- [ ] XCHG imm,reg
- [ ] XCHG reg,imm
- [x] XCHG reg,mem

The fomal name of the LOOP instruction is

- ( ) Loop According to EBX Counter
- ( ) Like Object Oriented Programming
- (•) Loop According to ECX Counter
- ( ) Loop According to CSI Counter

Adding 5 to 0FBh in an 8-bit register sets the Zero flag.

- (•) True
- ( ) False

The following instructions will set the Carry flag:

```
mov al,0FEh
sub al,2
```

- ( ) True

- (•) False

The MOVSX instruction is only used with unsigned integers.

- ( ) True
- (•) False

Which library procedure writes a single character to standard output?

WriteChar

The USES operator, coupled with the PROC directive, lets you list the names of all registers modified within a procedure.

- (•) True
- ( ) False

Which library procedure writes an unsigned 32-bit integer to standard output in hexadecimal format?

WriteHex

Which library procedure locates the cursor at a specific row and column on the screen?

Gotoxy

There are several important uses of runtime stacks in programs (select all that apply):

- [x] When the CALL instruction executes, the CPU saves the current subroutine's return address on the stack.
- [x] The stack provides temporary storage for local variables inside subroutines.
- [x] When calling a subroutine, you pass input values called arguments by pushing them on the stack.
- [x] A stack makes a convenient temporary save area for registers when they are used for more than one purpose. After they are modified, they can be restored to their original values.

Which library procedure generates a 32-bit pseudorandom integer in a caller-specified range?

RandomRange

The linker combines object files into an executable file.

- (•) True
- ( ) False

Which register contains the starting address of data when calling DumpMem?

- ( ) EAX
- ( ) EBX

- ECX
- EXI

- ESI
- EXD

Which library procedure returns the number of milliseconds elapsed since midnight?

GetMSeconds

A stack is also called a FIFO structure (First-In, First-Out) because the last value put into the stack is always the first value
taken out.

- True
- False

Which register contains an integer before calling WriteDec?

- EWD
- EBX

- EAX
- ECX

- EDX
- EXA

Which of the following code sequences assigns the value 0x10 to EBX? (select all that are correct)

- [x]
```
mov  edx,20h
push edx
mov  ecx,10h
push ecx
pop  ebx
pop  edx
```
- [ ]
```
mov  ecx,10h
mov  edx,20h
push ecx
push edx
pop  ebx
pop  edx
```

- [ ]
```
push 20h
mov  ecx,10h
```

```
push ecx
pop  eax
pop  ebx
mov  edx,20h
push edx
mov  ecx,10h
push ecx
pop  ebx
pop  edx
```

By default, labels are visible only within the procedure in which they are declared.

- True
- False