homework 2

1. what operation is always needed before storing (push operation) something to the stack?

increasing the stack pointer value

2. consider the following c code. let us assume that the function `sum()` is crrently executing in the processor. which main memory address is currently available in the return address register `ra`?

```
int sum(int a, int b) {
    return (a + b);
}

void main() {
    int x = 10;
    int y = 20;
    int z = sum(x, y);
    z++;
    x = x + 2;
}
```

memory address that stores the line `z++;` is stored in the return address register `ra`

3. which of the following is true for volatile type qualifier?

should be used to represent register of memory-mapped peripheral the value of volatile variable could be changed by external devices

4. match the correct binary values for different steps of 2's complement calculation for decimal number 20

step 1: convert the decimal to binary

20 (base 10) = 00010100 (base 2)

step 2: invert all the bits

00010100 (base 2) = 11101011 (base 2)

step 3: add 1 to get the 2's complement

11101011 (base 2) = 11101100 (base 2)

5. which are the possible reasons for choosing c over java for embedded systems?

- faster code execution
- low memory requirement
- simpler to write and easier to maintain

6. write the mips assembly code for the assignment statement as given below

```
C[250] = C[240] - b;
```

`C` is an array stored in the memory. assume that register `$s2` is storing the variable `b`, and `$s3` is storing the base address of the array `C`. use register `$t1` for storing temporary values.

```
lw $t0, 960($s3)    # load C[240] into $t0 (240 * 4 = 960)
sub $t1, $t0, $s2   # subtract b from $t0 and store in $t1
sw $t1, 1000($s3)   # store the result in C[250] (250 * 4 = 1000)
```

- lw $t0, 960$(s3) loads the value of C[240] from memory into register $t0. Since each memory address stores 4 bytes (1 word), we need to multiply the index by 4 to get the correct memory offset.
- sub $t1, $t0, $s2 subtracts the value of register $s2 (which stores the value of b) from the value in $t0. The result is stored in $t1.

- sw $t1, 1000(s3) stores the value in \$t1 into the memory location corresponding to C[250]. Again, we multiply the index by 4 to get the correct memory offset.

7. write the machine code in binary for the assembly code for the assembly codes in question Write the machine code in binary for the assembly codes for the assembly codes in question (a) and mark the different segments (e.g., op, rs, rt etc.) on the machine code. Use the MIPS instruction reference guide on Canvas. An example format is below

```
[000000][10001][10010][01000][00000][100000]
opcode: 000000
rs: 10001
rt: 10010
rd: 01000
shamt: 00000
funct: 100000
```

```
lw:   100011 10011 01000 1110100000000000
      op     rs    rt    immediate

subu: 000000 01000 10010 01001 00000 100011
      op     rs    rt    rd    shamt  funct

sw:   101011 10011 01001 1111101000000000
      op     rs    rt    immediate
```

8. why do we use registers in addition to the main memory?

alu can access the registers faster compared to the main memory

9. let's assume that `x = 8`, which of the following can we calculate using local left shift operation?

all numbers in decimal

$x \times 12$ and $x \times 8$ can be calculated using logical left shift.

$x \times 32$ and $x \times 16$ would require shifting by more than 32 bits, which is not possible in most processor architectures since they have a fixed number of bits for the data registers.