Question 1

1 / 1 pts

Which of the following virtual machine commands would implement a call of a Jack function, `myFunc`, that takes a single (1) int parameter but does not return any value (void)?

- ⦿
  ```
  call myFunc 1
  pop temp 0
  ```

- ○
  ```
  call myFunc 0
  ```

- ○
  ```
  call myFunc 1
  ```

- ○
  ```
  call myFunc 0
  pop temp 1
  ```

在Jack虚拟机中,函数调用时为了保证调用后栈的平衡,需要将参数弹出栈

但没有返回值的话,可以使用"pop temp 0"命令弹出一个单元格。
这是为了保持栈的平衡。

Question 2

1 / 1 pts

If variable x has value 2 and y has value 2, what is the value of the following Jack expression?

(x + 3) / (4 - y)

5/2

- 2

> rs 2

> All values are 16-bit integers so division is integer division.

因为Jack language 是从左到右操作的

Question 3

1 / 1 pts

Which of the following are valid operators in the Jack language?

- ☐ >=
- ☐ &&
- ☐ +=
- ☑ =
- ☑ |
- ☑ /
- ☑ <
- ☐ !
- ☐ !=
- ☑ *

Question 4
1 / 1 pts

If variable x has value 2 and y has value 1, which values could result from the following Jack expression?

x + 3 * 4 - y

2   1

- [ ] 17
- [x] 19 ✓
- [x] 11 ✓
- [ ] 5
- [x] 15
- [x] 13 ✓

> Operator precedence is undefined so the compiler writer can produce code that executes the operators in any one of 6 orders. However, most Jack compilers would evaluate the expression from left to right if there was a choice.

Question 5
1 / 1 pts

If you write the following call in a Jack function in class Bob:

    MyClass.Useful(34,"55")

but the function MyClass.Useful was declared to take three parameters, an int, an Array and a BankAccount, what happens?

- [ ] The program runs and a segmentation violation occurs when the missing BankAccount parameter is accessed.
- [ ] The program runs and a segmentation violation occurs when "55" is used as an array
- [ ] The class Bob compiles but the program does not assemble correctly
- [x] The program runs but does not work as expected

> The return address will be used as the address of the BankAccount object ...

- [ ] The class Bob does not compile
- [x] The class Bob compiles and the program assembles correctly.

从左到右:
① 2+3=5
5×4=20
20-1=19
算术顺序:
② 2+3×4=14
14-1=13
右到左:
③ 4-1=3
3×3=9
9+2=11

④: (x+3)(4-y)
= 5·3=15

segmentation violation 分段错误:
由于访问了一个不存在其内存段中的地址而引起的,
或试图在只读的内存段中进行写操作.