Input and output in the MiniMIPS assembly language

The default I/O mode of MiniMIPS is via a system call instruction, syscall. It is specified by a function integer between 1 and 10, where 1, 2, 3, and 4 are for output, 5, 6, 7 and 8 are for input, and 9 and 10 are for system control use. The following example is to output an integer.

```
addi $a0, $zero, 4    # prepare for printing out integer 4
addi $v0, $zero, 1    # request a system call for function 1 – output an integer
syscall
```

When the instruction syscall is executed, the integer is printed out in a console window popped from the SPIM simulator. The following example is to input an integer into register $v0

```
addi $v0, $zero, 5    # request a system call for function 5 – input an integer
syscall               # an integer is returned in register $v0
```

You need to input an integer from within the console window.

Take a look at the following table for the correspondence between an integer and its function for syscall.

($v0)	Function	Arguments	Result
1	Print integer	Integer in $a0	Integer displayed
2	Print floating-point	Float in $f12	Float displayed
3	Print double-float	Double float in $f12,$f13	Double float displayed
4	Print string	Pointer in $a0	Null-terminated string displayed
5	Read integer		Integer returned in $v0
6	Read floating-point		Float returned in $f0
7	Read double-float		Double-float returned in $f0,$f1
8	Read string	Pointer in $a0, length in $a1	String returned in buffer at pointer
9	Allocate memory	Number of bytes in $a0	Pointer to memory block in $v0
10	Exit from program		Program execution terminated