MINF18
TD OS Reporting
Devoir 1
Members:
Vo Hung Son
Tran Quang Nhat

1. **What is the goal?**

What is the reasonably expected output considering #if 0 transformed to #if 1? We will not activate the code
at first, so that it can compile until we have implemented it.

Answer:
Only the first block will be processed until someone changes the 1 to a 0. Then the other block will be compiled. This is a convenient way to temporary switch blocks of code in and out while testing different algorithms.

Explain why it is a mistake to try:
+ to read a character before being warned that a character is available
because Write "ch" to the console display,
and return immediately. "writeDone"
is called when the I/O completes.
+ to try to write before being warned that the previous writing is complete.
because the previous writing must be complete then "writeDone" is called when the I/O completes so to try to write before the error will be occur.

5. From characters to strings
5.2: **The number of characters written must be returned. Beware of the int *value argument of ReadMem: why can**
not we just pass a pointer pointing inside the buffer to?
Answer:
because A pointer is a variable that stores a memory address. Pointers are used to store the addresses of other variables or memory items. Pointers are very useful for another type of parameter passing, usually referred to as **Pass By Address**. Pointers are essential for dynamic memory allocation.

The choice of the file in which write this function is yours, several places are appropriate. You must motivate your
choice in the report:

Define int c opyS trin gFrom Mach ine (int from , char * to , unsigned size) in
synchconsole.h: `int copyStringFromMachine(int from , char * to , unsigned size);`
implement this method in **SynchConsole**.cc
`//(5.2) Write a procedure similar to strcpy:`

```
void SynchConsole::copyStringFromMachine (int from, char *to,
unsigned size) {
        unsigned i;
        for(i=0;i<size && machine->mainMemory[from+i] !='\
0';i++) {
                to[i] = machine->mainMemory[from+i];
        }
        to[i] = '\0';
}
```

5.3 Why would not it be reasonable to allocate, to a buffer, the same size than the MIPS string? Depending on how
you allocated the buffer, make sure that the buffer is released after the system call is complete

Answer:
This depends on your platform and compiler settings. Simple rule of thumb: if you're in doubt about whether your code will overflow the stack, write it in a way which can't.

6. But how to stop?
What happens if you remove the call to Halt() at the end of the main function of putchar.c? Decrypt the error
message and explain it. What to change to avoid this error?
Answer: the process will not be blocked.

```
#include "syscall.h"

int
main ()
{
    Halt ();

    /* not reached */
    return 0;
}
```

How to take into account the return value return n of the main function if this is declared as a integer value?
Look for test "who" call the main function.
Answer:
putchar.c,shell.c,matmult.c call to main function.

7. Reading Functions
What do you do in case of end of file?
Answer:
EOF is -1 because that's how it's defined. The name is provided by the standard library headers that you #include. They make it equal to -1 because it has to be something that can't be mistaken for an actual byte read by getchar(). getchar() reports the values of actual bytes using positive number (0 up to 255 inclusive), so -1 works fine for this.