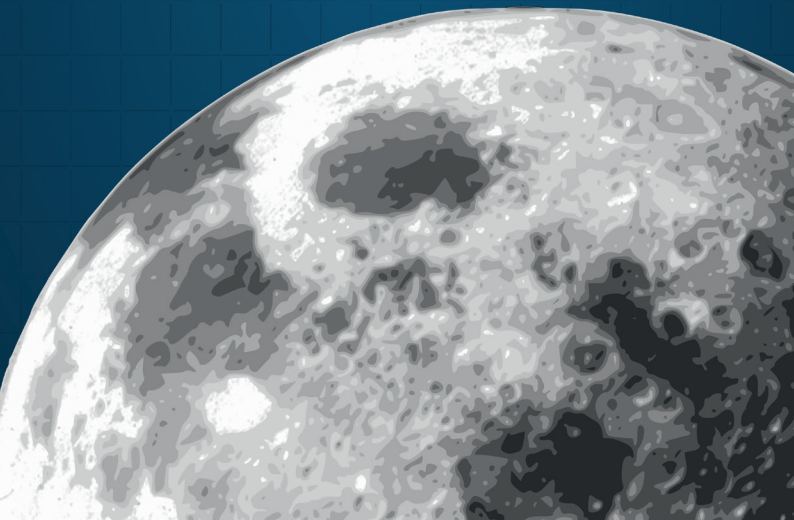


11001

ESPAÑOL



En 1969, millones de personas seguían los acontecimientos que sucedían a 384.000 km de la Tierra pegados a sus televisores. Tres minutos antes de que el Eagle se posara en la **Luna**, su computadora de a bordo lanzó varias alarmas. Debido a un fallo en los manuales, un radar no necesario para el aterrizaje estaba encendido cuando debería estar apagado. Enviaba señales erróneas a la computadora de a bordo, que tenía que encargarse además de todas las operaciones propias del alunizaje. Afortunadamente, el software que gestionaba la computadora de a bordo fue lo suficientemente inteligente como para detectar el problema y la computadora avisó a los astronautas diciendo: "estoy sobrecargada con más tareas de las que debería estar haciendo en este momento, así que me voy a centrar solamente en las tareas importantes, es decir, las que tienen que ver con aterrizar". Recordemos que era una computadora con menos potencia de cálculo que una calculadora. Sin este novedoso e inteligente diseño, el pequeño paso para el hombre y gran salto para la Humanidad habría acabado en siniestro total.

Hicieron falta más de 30 años antes de que la NASA reconociera el mérito de **Margaret Hamilton**, la directora del centro de ingeniería del software del MIT encargada de desarrollar el software de a bordo de todo el programa Apollo, el único que ha sido capaz de permitirnos pisar otros mundos y volver para contarlos.

Atención: Cuando se haga referencia a un "jugador" en este manual también debe entenderse, por supuesto, como "jugadora".

MOON es un juego educativo en el que simularemos el funcionamiento de una **computadora sencilla**.

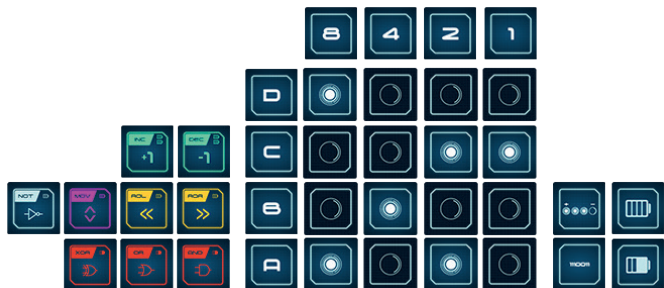
Ayuda a los astronautas del Eagle a completar su misión mientras aprendes a contar en binario, realizas operaciones **lógicas** y descubres cómo funciona una computadora **por dentro** al mismo tiempo que pasas un rato **divertido**.

MOON está recomendado para edades a partir de **10 años**, de **1 a 4 jugadores**, con una duración de entre **15 y 45 minutos** (dependiendo de la dificultad elegida).

PREPARA UNA PARTIDA

1. Sitúa los **4 registros** de la CPU (A, B, C y D) y sus correspondientes bits **apagados** en el medio de la mesa. Este será el tablero central.
2. Coloca las cartas de **operación** a la izquierda del tablero central, ordenadas en función de su coste de energía: primero las que cuestan 2 unidades de energía (INC, DEC), luego las que cuestan 1 (NOT, ROL, ROR, MOV) y finalmente las que cuestan 1/2 (OR, AND, XOR).
3. Baraja el mazo de **cartas objetivo** y colócalo boca abajo a la derecha de los registros. Estas cartas representan los cálculos que necesitan hacer los astronautas para alunizar.
4. Coge tres **unidades de energía** y ponlas junto a ti.

MOON simula el funcionamiento de una **computadora real** empleando operaciones y datos idénticos a los que se usan en microprocesadores reales. Por lo tanto, es conveniente repasar cómo contar en binario antes de poder jugar:



CONTAR EN BINARIO

Tanto los módulos individuales de RAM como los **registros** de la CPU tienen diferentes bits que sirven como **contadores binarios**. Cada posición tiene un número asociado (1, 2, 4 y 8 en registros de 4 bits).

Si todos los bits de todas las posiciones están **apagados**, se estará representando el número **cero**.



Si hay bits encendidos, hay que **sumar** los números de los recuadros que tienen sus **bits encendidos** para saber qué número se representa.



Por ejemplo, esta combinación representa el número **3** porque los bits de las posiciones **1** y **2** están activados, así que $1 + 2 = 3$.

Esta combinación representa el número **9** porque los bits de las posiciones **1** y **8** están activados, así que $1 + 8 = 9$.



JUEGO COOPERATIVO

Prepara la partida como se explica en la página 3 de este manual. Si es tu primera partida, toma 3 unidades de energía y no utilices las operaciones OR, AND y XOR. Más adelante, podrás adecuar la dificultad del juego a tu nivel.



Para ganar, debes ayudar a los astronautas en su misión **resolviendo** todas las cartas objetivo del mazo. Las cartas objetivo contienen **combinaciones de bits** que tienes que lograr en el registro A de la CPU.



Al principio de la partida, toma las tres primeras cartas del mazo de cartas objetivo y cópialas en los registros B, C y D. Luego, descártalas.



Después, toma una carta del mazo de cartas objetivo y colócala delante del mazo por la cara en la que se puede ver la combinación de bits objetivo.

En cada ronda, podrás realizar tantas operaciones como desees en función de las unidades de **energía** de que dispongas (recuerda que hay operaciones como OR que requieren 1/2 unidad de energía, mientras que otras como INC requieren 2).



No es obligatorio gastar todas las unidades de energía en una ronda, pero **no** se podrán **acumular** para la siguiente.

Utiliza tu energía para realizar **operaciones** sobre los registros A, B, C y D de la CPU y lograr resolver el objetivo. Recuerda que una carta objetivo no se dará por resuelta hasta que su valor no se encuentre en el **registro A** de la CPU.

Tanto si has logrado resolver la carta objetivo como si no, al terminar la ronda tendrás que hacer **avanzar** las cartas objetivo una



posición, tomar una **nueva carta** objetivo del mazo y colocarla boca abajo delante del mazo.

Es en este momento cuando recuperarás toda la energía que tenías al principio de la ronda.

Si al finalizar una ronda una carta objetivo avanza hasta la **5ª posición**, habrás tardado demasiado y la misión lunar habrá fracasado.



Esto puede ocurrir incluso si no quedan cartas objetivo en el mazo pero tardas más de **5 rondas** en resolver las últimas cartas.

Es decir, las cartas objetivo van avanzando hacia delante al final de cada ronda independientemente de que queden cartas en el mazo o no.

Si, por el contrario, logras resolver **todos** los objetivos del mazo sin que esto ocurra, habrás **triunfado** ayudando a los astronautas a alunizar.

Además, hay dos cartas objetivos especiales que no tienen una combinación de bits sino un **"bug"**.

Estas cartas especiales **no se pueden resolver** y cuando salgan bloquearán una de las posiciones dentro de la lista de objetivos pendientes para el resto de la partida.



Para convertir los bits de los registros de la CPU en otros valores que necesitemos durante el juego, usamos las operaciones. Veamos cómo se usan las **operaciones** INC, DEC, ROL, ROR, MOV, NOT, OR, AND y XOR.

INC

Esta operación se usa sobre **1 registro** y cuesta **2 unidades** de energía. **Suma** uno al valor almacenado en el registro:



Si el registro almacena el valor máximo (todos los bits encendidos), se da **la vuelta** al contador:



DEC Esta operación se usa sobre **1 registro** y cuesta **2 unidades** de energía. **Resta** uno al valor almacenado en el registro:



Si el registro almacena el valor mínimo (todos los bits apagados), se da **la vuelta** al contador:



ROL Esta operación se usa sobre **1 registro** y cuesta **1 unidad** de energía. **Desplaza** cada bit a la siguiente posición hacia la

izquierda, salvo el último bit que ocupa la posición libre que ha dejado el primero:



En muchas ocasiones, equivale a **multiplicar por 2** el valor del registro:



ROR Esta operación se usa sobre **1 registro** y cuesta **1 unidad** de energía. Desplaza cada bit a la siguiente posición hacia la derecha, salvo el primer bit que ocupa la posición libre que ha dejado el último:

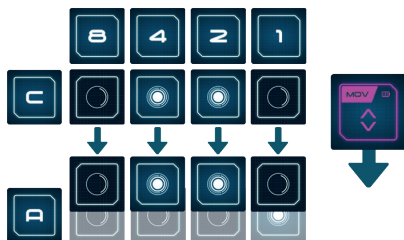


En muchas ocasiones, equivale a **dividir entre 2** el valor del registro:



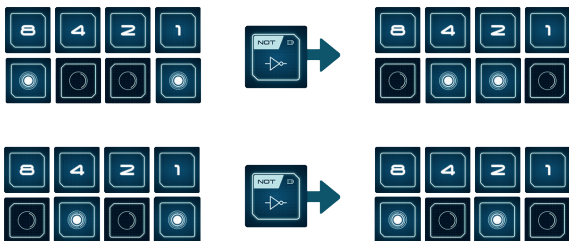
MOV Esta operación se usa sobre **2 registros** o sobre 1 registro y la memoria **RAM** y cuesta **1 unidad** de energía (1/2 en modo competitivo).

Copia todos los bits de un registro a otro **sobrescribiendo** lo que contenía el destino (es muy útil para copiar un valor en tu RAM y recuperarlo después sin que nadie te lo pueda modificar con sus operaciones).



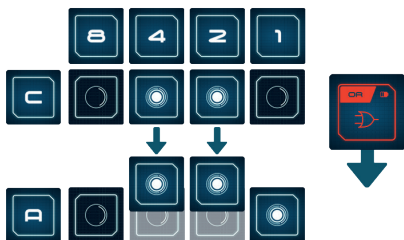
NOT Esta operación se usa sobre **1 registro** y cuesta **1 unidad** de energía.

Niega, es decir, pone lo **contrario** de lo que almacena cada uno de los bits del registro: si había un bit encendido, pone un bit apagado y si había un bit apagado, pone un bit encendido (damos la vuelta a todos los bits del registro):



OR Esta operación se usa sobre **2 registros** y cuesta **1/2 unidad** de energía.

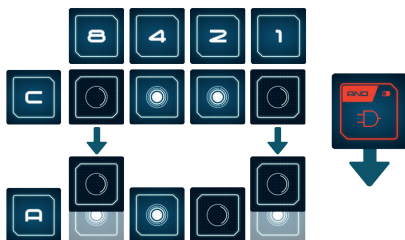
Copia solo los bits encendidos de un registro a otro.



AND

Esta operación se usa sobre **2 registros** y cuesta **1/2 unidad** de energía.

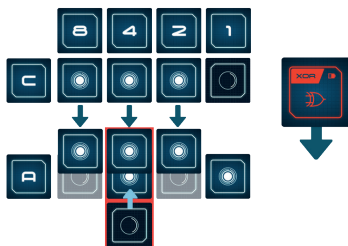
Copia solo los bits **apagados** de un registro a otro.



XOR

Esta operación se usa sobre **2 registros** y cuesta **1/2 unidad** de energía.

Copia solo los bits **encendidos** de un registro a otro, pero si el bit ya estaba encendido previamente, lo **apaga**.



DIFICULTAD

MOON permite ajustar su dificultad de varias maneras:

1. Definiendo el nivel de **energía** disponible por ronda. Sugerimos que para las primeras partidas se usen 3 unidades de energía (fácil) y se reduzca progresivamente (normal: 2.5; difícil: 2; master: 1.5).
2. Definiendo cuántas cartas "**bug**" contendrá el mazo de cartas objetivo (fácil: ninguna; normal: 1; difícil: 2; master: 2).
3. Definiendo el estado **inicial** de los **registros**. En el nivel fácil, al principio de la partida las primeras 3 cartas objetivo del mazo se copian en los registros B, C y D y se dan por resueltas. En nivel normal, se hace lo mismo con las primeras 2 cartas objetivo y los registros B y C. En el nivel difícil, se copia la primera carta objetivo en el registro B y en el nivel master, se pone el valor 1 en el registro A y el resto de registros a 0.
4. Añadiendo las cartas de **eventos** al mazo de cartas objetivo.

EVENTOS

Añade las cartas de evento al mazo de cartas objetivo para hacer las partidas más emocionantes:

Las cartas **RESET** registro **reinician** el registro indicado:



Las cartas **ERROR** registro **inutilizan** el registro indicado.



Las cartas **ERROR** operación **inutilizan** la operación indicada.



Las cartas **OK** pueden **reparar** un ERROR en un registro o en una operación (pero no pueden guardarse para después).



MODO COMPETITIVO

Antes de jugar en modo competitivo, conviene haber jugado en modo colaborativo. Revisa las secciones previas para conocer el funcionamiento básico del juego.

Prepara la partida como se explica en la página 3 de este manual. Cada jugador **elige un color**, toma la carta de **RAM** de ese color y coloca todas las posiciones de su módulo de RAM apagadas. Reparte las unidades de **energía** a cada jugador en función su nivel: 4 para fácil, 3 para normal, 2.5 para difícil, y 2 para master.

Baraja el mazo de **cartas objetivo** y colócalo boca abajo a la derecha de la CPU. Cada jugador toma una carta objetivo del mazo, la

ve y la coloca junto a su módulo de RAM **sin mostrar** la combinación de bits correspondiente al resto de jugadores.



En cada turno, cada jugador podrá hacer tantas operaciones como unidades de energía disponga. No es obligatorio jugarlas todas, pero **no** es posible **ceder** energía que no se desee usar a otros jugadores.

Cualquier jugador puede modificar los valores de todos los bits en los 4 registros de la CPU en su turno, pero **no** podrá **copiar** o **modificar** los valores almacenados en la **RAM** del resto de jugadores.

Si un jugador logra durante su turno que el **registro A** de la CPU contenga su combinación de bits objetivo, **mostrará** su carta objetivo al resto de jugadores, la guardará junto a su módulo de RAM y tomará otra carta objetivo del mazo.



Al finalizar el mazo de cartas objetivo, **ganará** quien más objetivos haya logrado resolver.

En el modo competitivo, hay dos **cambios** con respecto al modo cooperativo:

1. La operación MOV requiere **1/2 unidad** de energía (en lugar de 1 unidad de energía).
2. Las cartas "bug" sirven para poder **ver la carta objetivo de otro jugador** en cualquier momento de la partida. Cuando un jugador toma esta carta del mazo, la enseña al resto de jugadores y la guar-

da para su posterior uso durante alguno de sus turnos. Al ser usada, la carta bug **pasa a estar en posesión** de quien ha tenido que mostrar su carta objetivo y podrá usarla cuando lo desee contra otro jugador durante alguno de sus turnos. Estas cartas **no cuentan** para el recuento de objetivos resueltos al finalizar la partida.

HACKERS

Opcionalmente, puedes hacer que cada jugador tenga características **especiales** en las partidas en modo competitivo. Mira el dorso de las cartas de RAM para ver qué hacker quieres ser:



Verde: puede usar **INC** o **DEC** consumiendo solo 1 unidad de energía.



Amarillo: puede usar **ROL** o **ROR** consumiendo solo 1/2 unidad de energía.



Morado: puede hacer dos **MOV** sin consumir energía en cada turno.



Rojo: puede hacer dos operaciones lógicas **OR**, **AND**, **XOR** sin consumir energía en cada turno.

BITS EXTRA

El diseño modular de MOON permite **ampliar** el número de bits de su CPU, incrementando sustancialmente la complejidad del juego.

El juego base incorpora **8 bits extra** para poder ampliar los registros A, B, C y D a 5 bits o 6 bits de tamaño.



Dado que las cartas objetivo solamente contienen combinaciones de 4 bits, será necesario emplear **dos cartas objetivo** para indicar las combinaciones de 5 o 6 bits que hay que resolver.

En el modo **cooperativo**, sacaremos dos cartas objetivo al resolver un objetivo:

- Resolvemos **un objetivo**.
- Mostramos el **siguiente** objetivo.
- Al terminar la ronda, colocamos otra carta objetivo.

El objetivo estará formado por todos los bits **más a la derecha** del conjunto de las dos cartas. Por ejemplo, jugando con registros de 6 bits, serán los 4 bits de la carta de la derecha y los 2 bits más a la derecha de la carta de la izquierda:



En el modo **competitivo**, pondremos cada parte del objetivo **a cada lado** de nuestro módulo de RAM para evitar que se mezclen.



Este juego ha sido creado dentro del proyecto COMPUS, cofinanciado por el Programa Erasmus+ de la Unión Europea y desarrollado por la Universidad de Deusto (España), AGR Priority (España), Fundación Educativa ACI - Esclavas SC-Fatima (España), Școala Gimnazială Ferdinand I (Rumania) y OEIIZK: Osrodek Edukacji Informatycznej i Zastosowań Komputerow w Warszawie (Polonia). Para saber más acerca del proyecto, visita <http://compus.deusto.es>

El apoyo de la Comisión Europea para la elaboración de esta publicación no implica la aceptación de sus contenidos, que es responsabilidad exclusiva de los autores. Por tanto, la Comisión no es responsable del uso que pueda hacerse de la información aquí difundida.



Co-funded by the
Erasmus+ Programme
of the European Union