

# CODEWARDS

## Hora de código

---

Introducción a la programación

# Hora de código

---

## Introducción a la programación

**Objetivo de la clase:** Introducir a los estudiantes a los conceptos básicos de la programación.

---

**Actividades en la computadora:** Aprendizaje de comandos y conceptos básicos de programación

Comandos: `move`, `rotate`, `right`, `left`, `load`, `put`

Objeto: `cat`

Estructura de ciclo: `loop ... end`.

---

**Materiales necesarios:**

1. Computadora individual para cada estudiante con acceso a Internet y al simulador.

---

## Parte 1. Condiciones generales

- Para organizar la clase con el uso del simulador, el profesor debe ofrecer a cada estudiante un sitio de trabajo con conexión a Internet y con acceso a la web <https://codewards.org/hoc-cats>.
- Todos los navegadores y sistemas operativos son compatibles con el simulador.
- Lo más eficiente es organizar el trabajo con el simulador de manera individual para cada estudiante, haciendo énfasis en la importancia de avanzar por los niveles en orden según la ruta escogida (elegida por el propio estudiante).
- Al finalizar el trabajo con el simulador, los estudiantes pasan a la parte final, en donde se les agradece su participación en la “Hora de código” con la propuesta de imprimir/guardar el certificado de logro.

## Parte 2. Comienzo del trabajo

Al comenzar el trabajo, es esencial que la atención de los estudiantes se enfoque en lo siguiente:

1. El código se construye con la ayuda de diferentes objetos gráficos (botones-pictogramas) que se encuentran en la parte inferior de la pantalla. Los estudiantes también tienen acceso a un redactor de texto para escribir el código. Éste permite escribirlo directamente con el teclado, para sentirse como un programador de verdad.
2. El simulador tiene un sistema de tareas educativas y pistas con las cuales los estudiantes aprenden por su cuenta sobre los mecanismos de creación y escritura del código.

Antes de comenzar a trabajar con el simulador, hay que acordar las siguientes reglas con la clase (en casos de que surja alguna duda):

- A. “Leer la condición e intentar de nuevo.”
- B. “Pregunta primero a otras tres personas y después a mí”, - es decir, se debe preguntar a tres compañeros y si no pueden ayudar, entonces se pregunta al profesor.
- C. “Los “fuertes” ayudan a los que van un poco más atrasados (ayudan pero no resuelven por ellos).”

## Parte 3. Prácticas con la computadora

Explique que la manera más simple de controlar objetos es por un sistema de comandos:

QUIÉN + QUÉ (tenemos que hacer) + CÓMO (cuántos pasos, en qué dirección, etc.)

### *Concepto #1:*

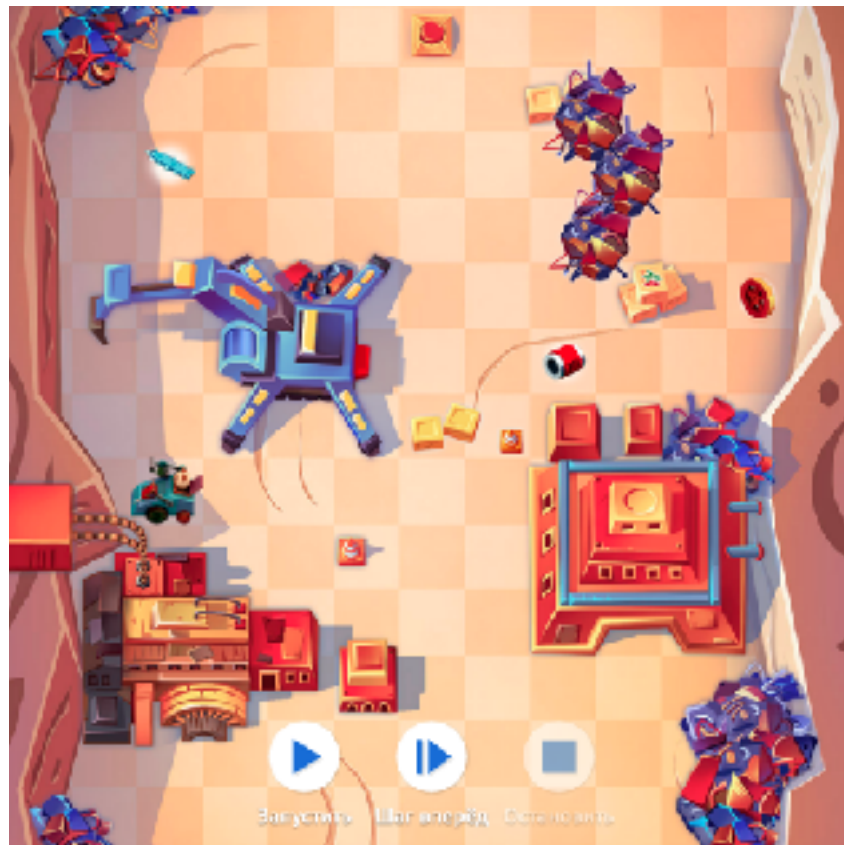
---

## **Quién + Qué + Cómo**

- Un estudiante de cada grupo pasa al pizarrón (con la tarjeta del robot). Los otros estudiantes dan comandos que hay que escribir en el pizarrón (en INGLÉS) para “programar” al alumno, quien deberá seguir los comandos. Por ejemplo: kid.go 4steps.
- Después, los estudiantes pasan a las computadoras y se les dan las instrucciones de cómo usar el programa.

En la plataforma en línea hay cómics que ayudan a conectar a los niños y niñas con la historia.

## Tarea #1



### Tarea:

¡Otras tres piezas más y tendremos todo lo necesario para poder armar! Escribe el código para recoger las piezas que faltan.

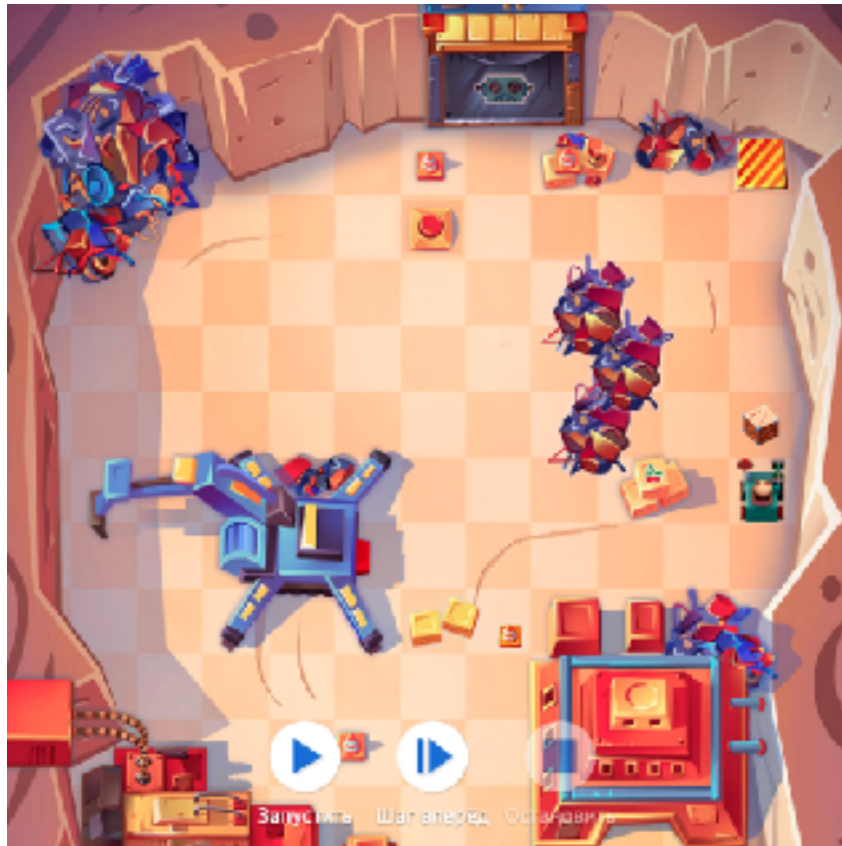
### Pista:

1. ¡Trata de no estrellarte con nada! Necesitamos al robot completo.
2. Para girar usa el comando rotate.
3. Para girar a la izquierda usa el argumento left. Para girar a la derecha - right.

### Código correcto:

1. `cat.rotate right`
2. `cat.move 5`
3. `cat.rotate right`
4. `cat.move 5`
5. `cat.rotate right`
6. `cat.move 3`
7. `cat.rotate left`
8. `cat.move 1`
9. `cat.move 3`
10. `cat.rotate left`
11. `cat.move 2`

## Tarea #2



### Tarea:

No podemos pasar. Tenemos que quitar la caja que nos obstruye la ruta hacia el punto de control. Hay que usar los comandos load (cargar) y put (poner).

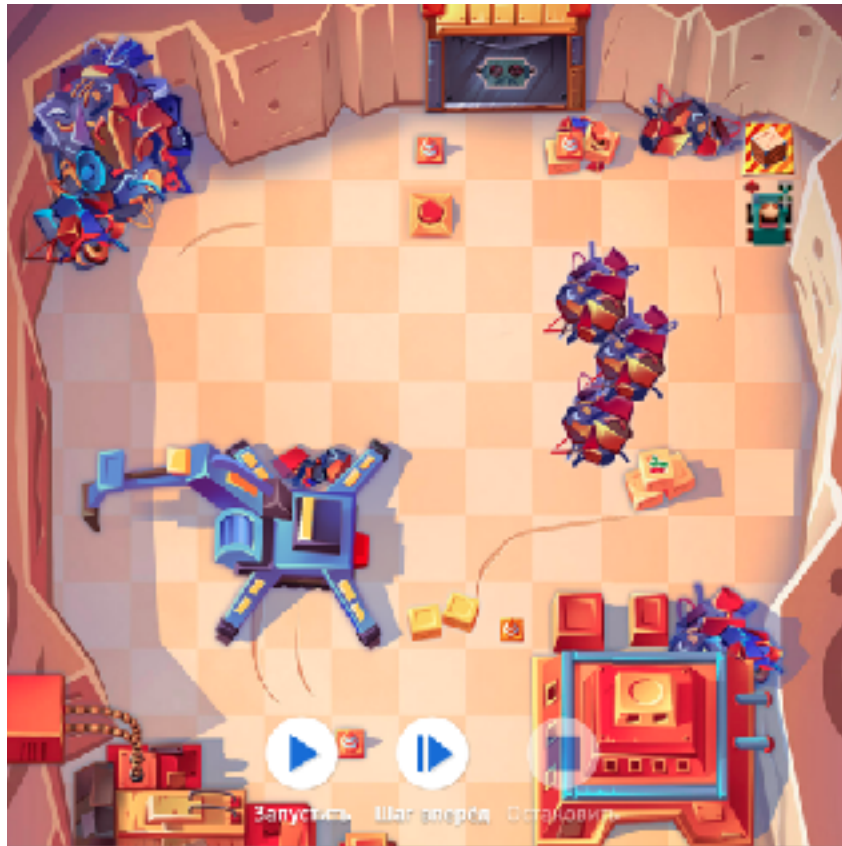
### Pista:

1. Para cargar la caja usa el comando load, para descargar - put.
2. Para recoger la caja o descargarla, hay que pararse enfrente.
3. Coloca la caja en la casilla señalada.

### Código correcto:

1. `cat.load`
2. `cat.move 4`
3. `cat.put`

## Tarea #3



### Tarea:

¡Perfecto! Hemos recogido suficientes piezas. Es hora de ir a la fábrica y usarlas para hacer algo útil que instalaremos en el robot. Abre la puerta y entra.

### Pista:

1. Me parece que hay que poner algo pesado en el botón.
2. Intenta poner la caja en el botón. Usa los comandos load (cargar) y put (poner).
3. Para poner la caja en el botón, hay que pararse enfrente.

### Código correcto:

1. `cat.load`
2. `cat.rotate left`
3. `cat.move 4`
4. `cat.put`
5. `cat.rotate right`
6. `cat.move 2`

## Tarea #4



### Tarea:

Es hora de armar la herramienta. Saca las 3 piezas y colócalas en las 3 cajas que están en la cinta transportadora.

### Pista:

1. Coloca las piezas recogidas en las cajas que están en la cinta transportadora. Una pieza en cada caja. ¡Es muy simple!
2. Para sacar la pieza usa el comando put.
3. Sólo es posible sacar una pieza a una caja vacía.

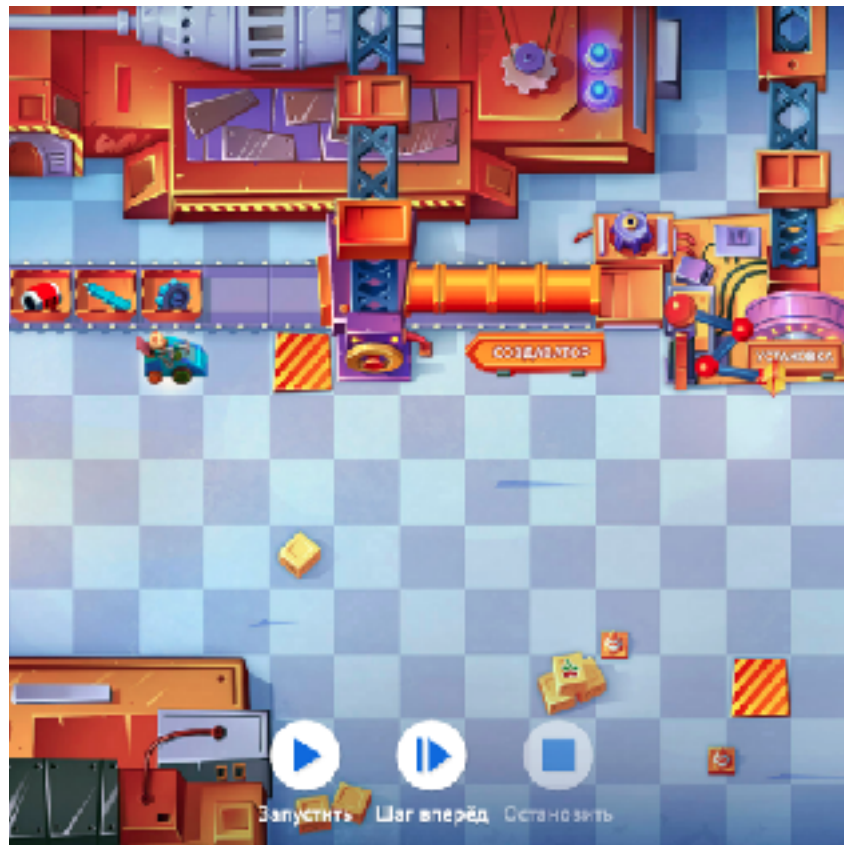
### Código correcto:

1. cat.rotate left
2. cat.move 2
3. cat.put
4. cat.rotate right
5. cat.move 1
6. cat.rotate left
7. cat.put
8. cat.rotate right
9. cat.move 1



```
10.cat.rotate left  
11.cat.put
```

## Tarea #5



### Tarea:

Ahora puedes fabricar algo para nuestro robot. Gira el botón del “Creatorator” con el comando spin (girar algo), para que todas las piezas se coloquen adentro. Después de crear una herramienta, muévete a la sección de montaje.

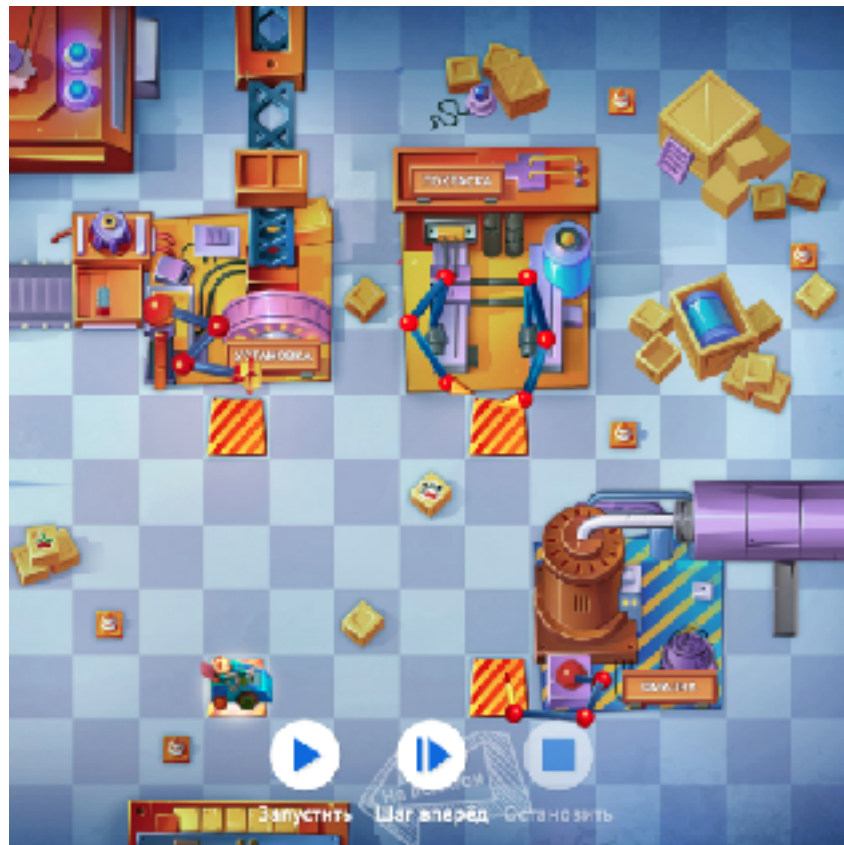
### Pista:

1. Algo nuevo y útil aparecerá sólo cuando las tres piezas sean cargadas en el “Creatorator”.
2. Un giro del botón mueve la cinta una casilla.
3. El botón se puede girar sólo desde el punto de control con el comando spin (girar algo).

### Código correcto:

1. cat.move 2
2. cat.spin 5
3. cat.rotate right
4. cat.move 2
5. cat.rotate left
6. cat.move 7
7. cat.rotate right
8. cat.move 3

## Tarea #6



### Tarea:

¡Vamos a instalar la herramienta! Hay que instalar, pintar y engrasar cada pieza. Usa la función loop (bucle o ciclo) para repetir comandos específicos de manera automática. Después de la instalación, muévete hacia la salida.

### Pista:

1. Hay que instalar tres tipos de herramientas en el robot. Para eso hay que acercarse tres veces a cada máquina. Para los comandos repetitivos usa loop (bucle o ciclo).
2. Tienes que comprender qué comandos deben repetirse y en qué orden. Entonces verás claramente cómo usar el ciclo. Por cierto, el ciclo siempre comienza en el mismo punto.
3. Después de armar las herramientas, muévete a la salida.
4. El ciclo debe comenzar con el comando loop y acabar con el comando end.

### Código correcto – variante 1:

1. `cat.rotate left`
2. `loop 3`
3. `cat.move 4`
4. `cat.rotate right`
5. `cat.move 4`

```
6.      cat.rotate right
7.      cat.move 4
8.      cat.rotate right
9.      cat.move 4
10.     cat.rotate right
11.     end
```

**Código correcto – variante 2:**

```
1.     cat.rotate left
2.     loop 11
3.         cat.move 4
4.         cat.rotate right
5.     end
6.     cat.move 4
7.     cat.rotate left
8.     cat.move 1
```

## Tarea #7



### Tarea:

Nuestra siguiente herramienta: el taladro. Sigue tu ruta destruyendo piedras. Usa el comando drill (taladrar).

### Pista:

1. Fíjate en la posición de las piedras.
2. Parece ser que sería mejor usar la función loop (ciclo o bucle) para optimizar el código.
3. Mmm, parece que lo podemos usar hasta dos veces en el código.

### Código correcto:

```
1. cat.rotate left
2.   loop 4
3.     cat.move 1
4.     cat.drill
5.   end
6. cat.move 2
7. cat.rotate left
8.   loop 3
9.     cat.drill
10.    cat.move 2
11.  end
```

## Tarea #8



### Tarea:

¡Vamos a probar el escáner! Hay tres objetos escondidos en la niebla. ¡Hay que encontrar en dónde están! Escanea la zona por sectores. Se pueden ver los puntos de control, úsalos para parar y escanear. Para ello, usa el comando detect (detectar).

### Pista:

1. Para escanear un sector se necesita girar completamente sobre uno mismo.
2. Aparecerán cruces rojas en el mapa en los sitios en donde están los objetos.

### Código correcto:

```
1. cat.move 2
2. loop 3
3.     cat.move 4
4.     cat.rotate right
5. loop 4
6.     scanner.detect
7.     cat.rotate right
8. end
9. end
```

## Tarea #9



### Tarea:

Vamos a comprobar de qué están hechos los objetos que encontramos. Para ello, acércate a cada uno y escanéalo (comando scan) y observa qué se puede hacer con él. Utiliza la condición “if..., else” (si..., si no...) con la construcción “if” (si).

### Pista:

Si es wood (madera), entonces saw (serrar).

Si es rock (piedra), entonces drill (taladrar).

### Código correcto:

```

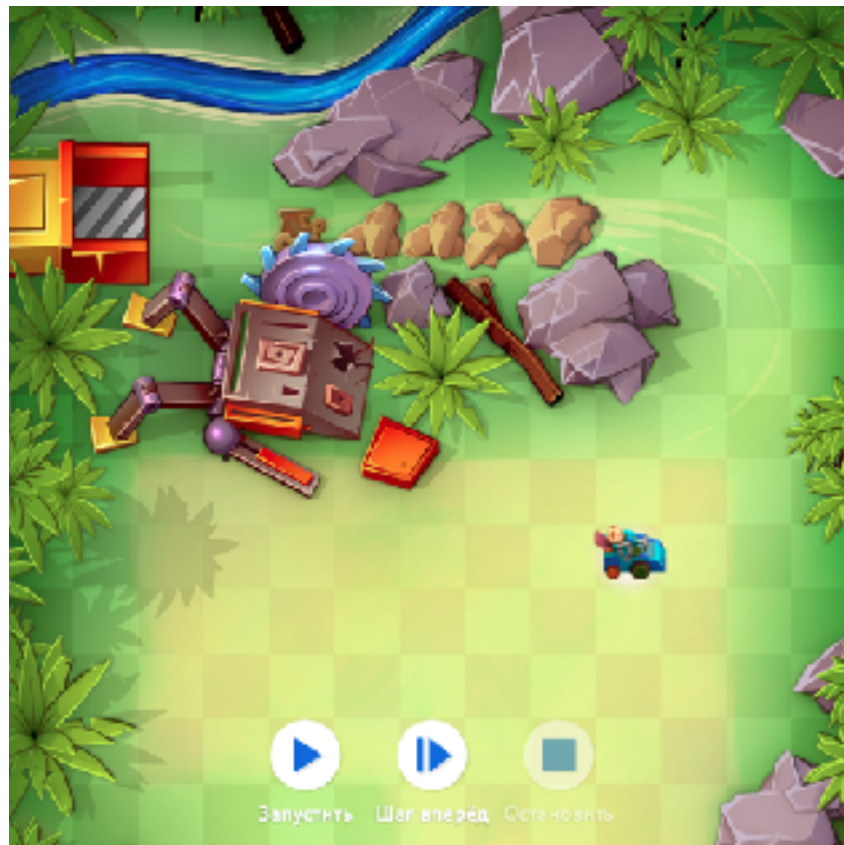
1. if scanner.scanWood
2.   cat.saw
3. else
4.   cat.drill
5. end
6. cat.move 3
7. cat.rotate right
8. cat.move 2
9. if scanner.scanWood
10.  cat.saw
11. else
12.  cat.drill
13. end

```

```
14. cat.move 3
15. cat.rotate right
16. cat.move 4
17. if scanner.scanWood
18.   cat.saw
19. else
20.   cat.drill
21. end
```



## Tarea #10



### Tarea:

Ya estamos casi al final, falta la última prueba. Nos falta pasar por el túnel, pero está bloqueado. Usa la construcción if (si), para quitar los obstáculos y poder pasar.

### Pista:

Si es wood (madera), entonces saw (sierra).  
Si es rock (piedra), entonces drill (taladra).

### Código correcto:

```

1. cat.move 2
2. cat.rotate left
3. cat.move 5
4. cat.rotate left
5. cat.move 2
6. loop 5
7. if scanner.scanWood
8.   cat.saw
9.   cat.move 1
10. else
11.   cat.drill
12.   cat.move 1
13. end
14. end
15. cat.move 3
  
```

## Parte 4. Reflexión.

### Juego activo “¡Te programaré!”

Al final de la clase, hay que repartir 3 tipos de tarjetas (colores) para empezar el juego:

1. objeto,
2. comandos (acciones),
3. argumentos (números).

Los estudiantes tienen una tarea: programar al profesor para que dé dos pasos hacia adelante. Él lo podrá hacer sólo si los equipos presentan los tres comandos de sus tarjetas correctamente.

Los alumnos escogen un objeto y componen un programa basado en el material aprendido (las tarjetas se pueden pegar o adherir al pizarrón con imanes, un estudiante = una tarjeta).