

| | | | |
|-----------------------|---|----------|------------------|
| Materia | LOGICA DE PROGRAMACION | Carrera | ING. EN SISTEMAS |
| Profesor | BLANCA ESTRADA RENTERIA | Semestre | 1° |
| PROYECTO FINAL | | | |
| Observaciones: | Todos los proyectos deberán cumplir con las instrucciones generales y debe de hacer una presentación tipo ejecutiva donde presente su proyecto. Máximo número de integrantes de equipo 4 | | |

Instrucciones Generales:

Deberá entregar el código fuente y el compilado del proyecto en un CD con todos los archivos necesarios para su ejecución y sin virus. Además de un manual de usuario que muestre su funcionamiento y todos aquellos documentos que crea necesarios como investigación adicional.

Entregar engargolada la documentación con lo siguiente:

- Portada
- Resumen descriptivo
- Temas investigados para la realización del proyecto.
- Bitácora de trabajo.
- Conclusiones
- Referencias consultadas

Además, el documento elaborado debe estar incluido en el CD en formato PDF, y el nombre del archivo deben ser los apellidos del jefe del equipo.

Proyecto copiado de Internet o de alguna otra fuente causará anulación de la calificación final del curso, así como proyectos con errores de compilación o que no se realice lo que se pide.

TEMAS A EVALUAR:

Se evaluará que el proyecto no tenga errores de compilación y/o sintaxis, errores de ejecución. Además, debe cumplir con todos los temas que marca el programa de la materia, como se describe a continuación:

1. Portada
 - a) Imagen con título de la UAA
 - b) Título del proyecto
 - c) Nombre de la materia
 - d) Nombres de los integrantes
 - e) Carrera, semestre y grupo
 - f) Nombre del profesor
 - g) Fecha de entrega

2. Resumen descriptivo

- a) Fortalezas
- b) Debilidades

Estos puntos los deben redactar con sus propias palabras, indiquen que fortalezas tiene su proyecto y cuales debilidades, justifiquen sus comentarios.

3. Temas Investigados para la realización del proyecto

4. Bitácora de trabajo

Hacer una tabla donde incluyan el nombre del integrante del equipo, actividades realizadas y tiempo (días) que se llevó en realizar dicha actividad (sean honestos y escriban lo que hizo cada uno)

5. Conclusiones

Con sus palabras expliquen de forma breve que fue lo que aprendieron con el desarrollo del proyecto.

6. Bibliografía o Referencias consultadas

Escribir la bibliografía que fue consultada para realizar el proyecto (libros, páginas de Internet, etc.) Recuerden que un formato adecuado para citar fuentes de libros, revistas, páginas web, periódicos, etc.

CODIGO

- Todo el programa debe ser modular, es decir, debe estar dividido en funciones.
- No debe utilizar variables globales a menos que se justifique su.
- Debe aplicar todos los temas vistos en el curso: Tipos de datos adecuados a la información que contendrán, ciclo y condicionales, vectores y matrices, Funciones.
- Utilizar los menús necesarios para acceder a cada parte del programa.
- Validar las capturas (que no existan errores de ejecución, es decir, que no “true” el programa).
- Cada sección de código del programa debe estar perfectamente bien comentada, es decir, cada función debe tener comentarios acerca de lo que se realiza.
- Los nombres de las variables y funciones deben ser de acuerdo al dato que almacenan o la acción que realizan

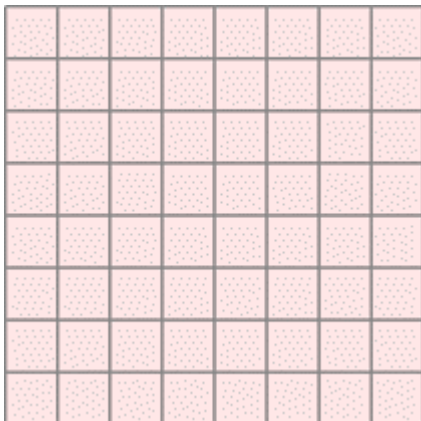
Cada uno de los proyectos que a continuación se describen, se realizaran en equipos de más de 4 personas, pudiendo ser de manera individual; el proyecto se entrega en la fecha definida en grupo, si se detectaran copias de código se anulan los proyectos de ambos equipos quedando así la materia con calificación reprobatoria, debido a que hay que aprobar la práctica y la teoría por separado.

| | | | |
|-----------------------|-------------------------|-------------------|---------------------|
| Materia | PROGRAMACION I | Carrera | ING. EN ELECTRONICA |
| Profesor | BLANCA ESTRADA RENTERIA | Semestre | 2° |
| PROYECTO FINAL | | | |
| Nombre del Proyecto: | 4 EN LINEA | Fecha de Entrega: | |

Instrucciones Generales 4 en línea:

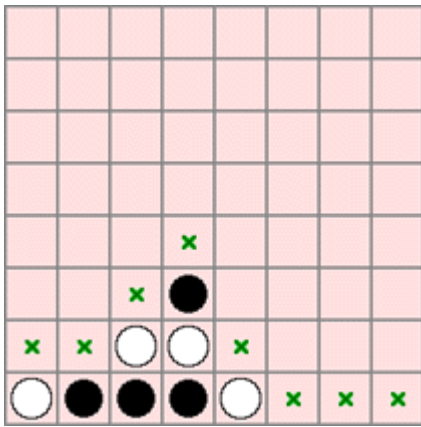
Posición inicial y objetivo del juego

El 4 en Línea pertenece a la familia de juegos populares ya que sus sencillas reglas se pueden aprender rápidamente ofreciendo un gran entretenimiento durante un largo tiempo. El objetivo de este juego consiste en colocar cuatro fichas en una fila continua vertical, horizontal o diagonalmente. Se juega sobre un tablero de 8x8 casillas que al empezar está vacío.



Cómo colocar las fichas

Ambos jugadores sitúan sus fichas (una por movimiento) en el tablero. La regla para colocarlas consiste en que la estas siempre "caen hasta abajo". Es decir una ficha puede ser colocada bien en la parte inferior de una columna o bien sobre otra de alguna otra columna. La siguiente imagen muestra un ejemplo de la posición de una partida en curso donde las cruces verdes señalan las casillas donde el jugador puede colocar una nueva ficha.



Cómo finalizar el juego

La partida termina si una de las siguientes condiciones se cumple:

- Uno de los jugadores coloca cuatro o más fichas en una línea continua vertical, horizontal o diagonalmente. Este jugador gana la partida.

Todas las casillas del tablero están ocupadas y ningún jugador cumple la condición anterior para ganar. En este caso la partida finaliza en empate.

| | | | |
|-----------------------|-------------------------|-------------------|------------------|
| Materia | LOGICA DE PROGRAMACION | Carrera | ING. EN SISTEMAS |
| Profesor | BLANCA ESTRADA RENTERIA | Semestre | 1° |
| PROYECTO FINAL | | | |
| Nombre del Proyecto: | Cuadrados de sumas | Fecha de Entrega: | |

Instrucciones Generales:

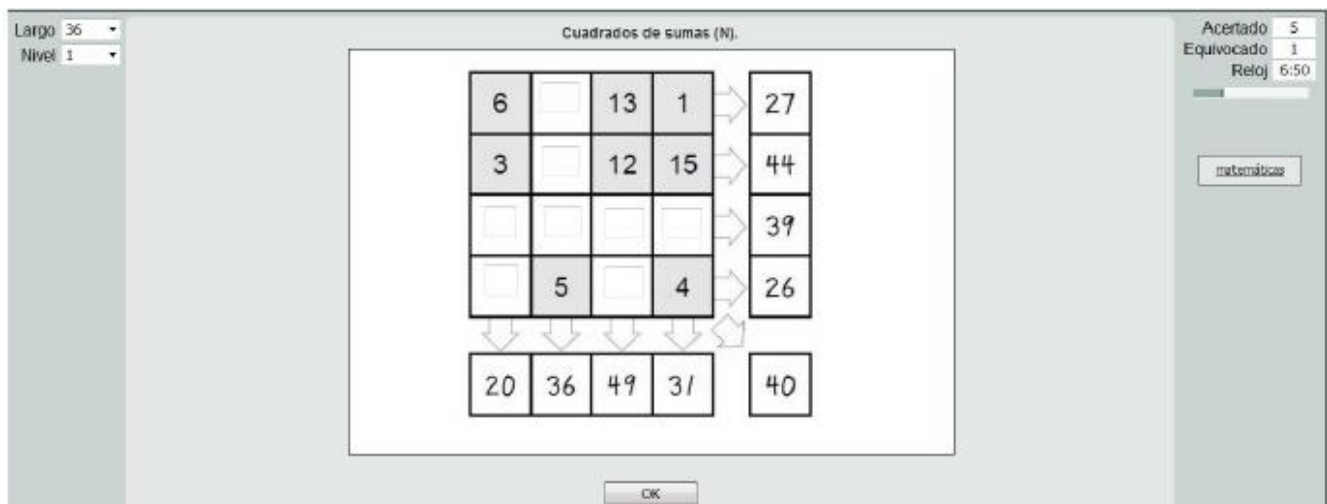
<http://www.actiludis.com/?p=58611>

Es una herramienta que les permite a los niños aprender sumatorias a manera de juego, el objetivo es implementar un tablero de $N \times N$, donde N puede variar de 3 a 6, el tablero contendrá solo algunos números que se ven involucrados en las sumatorias, los cuadros, deben ser completados de acuerdo a la sumatorias que aparecerán del lado derecho, abajo y esquina inferior derecha.

Además de mostrar el cuadro de sumas que se está resolviendo, deberá.

- Generar el tamaño del cuadro de manera aleatoria
- Contar el tiempo que tarda en resolver el cuadrado.
- Validar que el cuadro este completamente resuelto
- Validar que el cuadrado este correcto a terminar la edición del mismo.

El jugador puede resolver tantos cuadros como requiera, cada vez que se obtiene un nuevo cuadro, se debe generar el tamaño de manera aleatoria.



Puede revisar la siguiente liga para que le vea cómo funciona:

<https://www.thatquiz.org/es/practicetest?ux79b0vy1mtlo>

| | | | |
|-----------------------|-------------------------|-------------------|------------------|
| Materia | LOGICA DE PROGRAMACION | Carrera | ING. EN SISTEMAS |
| Profesor | BLANCA ESTRADA RENTERIA | Semestre | 1° |
| PROYECTO FINAL | | | |
| Nombre del Proyecto: | Maratón de C | Fecha de Entrega: | |

Instrucciones Generales:

1. PRESENTACION DEL JUEGO EN MODO GRAFICO

I. Inicio de Juego

II Vector fijo de al menos 50 preguntas, se inicializa al comenzar el juego

III Salir

2. Número de jugadores dos. Pedir Nombres de los jugadores y desplegarlos durante el juego

3. Tablero con 9 casillas

| | | | | | | | | |
|--------|---------------|----------|---------------------------|------------------------|----------|-----------|------------|-----|
| INICIO | declaraciones | Vectores | Sentencias Condicionales. | Sentencias Repetitivas | Matrices | Funciones | Biblioteca | FIN |
|--------|---------------|----------|---------------------------|------------------------|----------|-----------|------------|-----|

| | | | | | | | | |
|--|--|--|--|--|--|--|--|--|
| | | | | | | | | |
| | | | | | | | | |

3. Dependiendo de la casilla, se realizará una pregunta sobre algún tema y se presentarán tres opciones de respuesta para que el jugador elija una.

Casilla 1. Declaraciones e inicialización de variables y constantes de todo tipo de datos.

Casilla 2. Vectores

Casilla 3. Sentencias condicionales

Casilla 4. Sentencias repetitivas

Casilla 5. Manejo de matrices

Casilla 6. Manejo de Funciones

Casilla 7. Funciones de distintas bibliotecas

4. Se le debe indicar al jugador si acertó o no en su respuesta (puede ser con sonidos) e independientemente de que el jugador acierte o no, debe hacerse cambio de turno.

5. Si el jugador acertó, se le debe generar un número aleatorio entre 1 y 3 que determine cuantas casillas avanza, si no acertó en su próximo turno se le vuelve hacer una pregunta sobre un tema de la misma casilla.

6. Se debe ver en pantalla en que casilla esta cada jugador y mostrar en pantalla el número aleatorio que corresponde a cuantas casillas avanza.

7. Felicitar al jugador que gana.

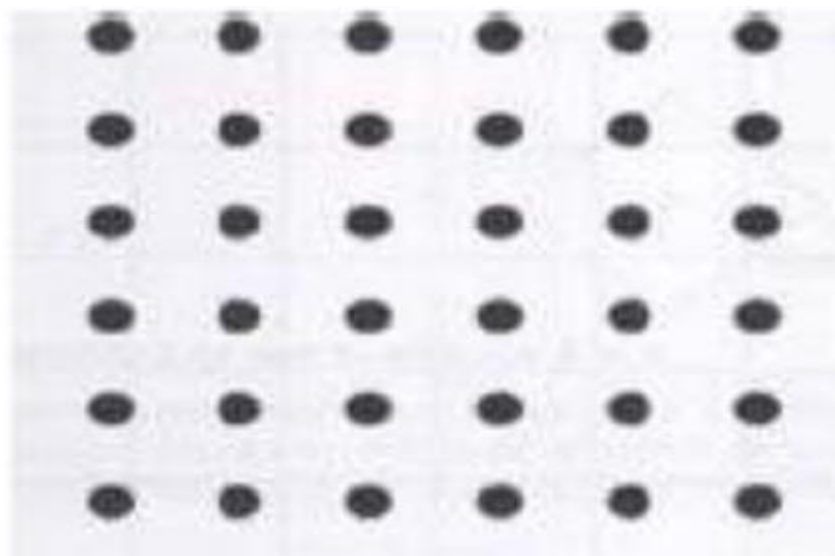
| | | | |
|----------------------|-------------------------|-------------------|------------------|
| Materia | LOGICA DE PROGRAMACION | Carrera | ING. EN SISTEMAS |
| Profesor | BLANCA ESTRADA RENTERIA | Semestre | 1° |
| PROYECTO FINAL | | | |
| Nombre del Proyecto: | Uniando Vértices | Fecha de Entrega: | |

Instrucciones Generales:

Es un juego para dos personas.

Material: El material es una hoja de papel cuadriculado, de cuadrados grandes, sobre la que se marcan 36 puntos y dos rotuladores de distinto color, uno para cada jugador.

Objetivo: El objetivo del juego es formar el máximo número de cuadrados, uniando vértices contiguos de la cuadrícula.



Reglas de juego:

1. Se echa a suertes el jugador que comienza a jugar.
2. Cada jugador, por turno, une dos vértices consecutivos de la cuadrícula mediante un segmento, en horizontal o vertical, pero nunca en diagonal.
3. Un jugador se atribuye un cuadrado cuando traza el cuarto lado. En este caso, escribe la inicial de su nombre dentro del cuadrado.
4. Siempre que un jugador forma un cuadrado, tiene derecho a realizar una jugada más.
5. Gana el jugador que ha formado más cuadrados.

El objetivo es que programe este típico juego utilizando matrices.

Al finalizar el juego, debe indicar que jugador fue el que gano y con cuantos cuadros lo realizo.

| | | | |
|----------------------|-------------------------|-------------------|------------------|
| Materia | LOGICA DE PROGRAMACION | Carrera | ING. EN SISTEMAS |
| Profesor | BLANCA ESTRADA RENTERIA | Semestre | 1° |
| PROYECTO FINAL | | | |
| Nombre del Proyecto: | KenKen | Fecha de Entrega: | |

Instrucciones Generales:

Inicialmente desarrollado por un profesor de matemáticas japonés. Tetsuya Miyaote, éste lo ideó para ayudar a sus alumnos a aprender aritmética.

Reglas del Juego:

1. En cada rectángulo hay que hacer la operación que indica el símbolo.
2. En cada fila y columna del cuadrado tiene que haber los números 1, 2, 3 y 4.

Ejemplo:

| | | | |
|----|----|----|----|
| 5+ | 6+ | | 3 |
| | 2 | 4+ | 5+ |
| 5+ | | | |
| 4+ | | 6+ | |

Solución:

| | | | |
|----|----|---|---|
| 5+ | 6+ | | 3 |
| 1 | 4 | 2 | 3 |
| 4 | 2 | 3 | 1 |
| 2 | 3 | 1 | 4 |
| 3 | 1 | 4 | 2 |

| | | | | |
|-----|----|-----|--|----|
| 2- | | 2÷ | | 9+ |
| 24× | 4- | | | |
| | 2÷ | 48× | | 4- |
| | | 75× | | |
| 3- | | | | 2 |

| | | | | |
|----|---|----|---|----|
| 2- | | 2÷ | | 9+ |
| 5 | 3 | 2 | 1 | 4 |
| 4 | 5 | 1 | 2 | 3 |
| 2 | 1 | 4 | 3 | 5 |
| 3 | 2 | 5 | 4 | 1 |
| 1 | 4 | 3 | 5 | 2 |

El objetivo es que programe este juego, los valores deben ser aleatorios, el tamaño de la tabla puede variar entre 3 y 6.

| | | | |
|-----------------------|-------------------------|-------------------|------------------|
| Materia | LOGICA DE PROGRAMACION | Carrera | ING. EN SISTEMAS |
| Profesor | BLANCA ESTRADA RENTERIA | Semestre | 1° |
| PROYECTO FINAL | | | |
| Nombre del Proyecto: | Cinco caminos | Fecha de Entrega: | |

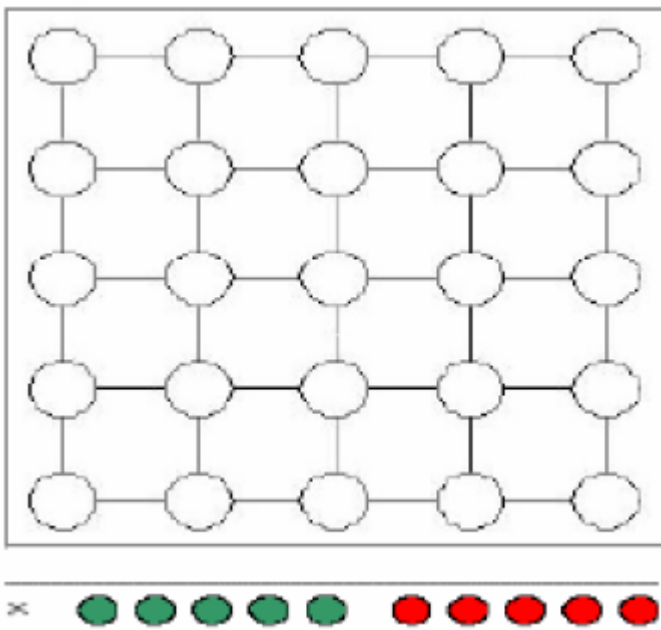
Instrucciones Generales:

Es un juego del norte de china y se suele jugar con dulces y cuando se captura una pieza del contrario se la comen realmente. El ganador también se come los dulces que quedan al finalizar la partida.

Reglas del Juego:

- Cada jugador tiene 5 fichas y se colocan alineadas en los bordes del tablero y frente a frente.
- Los jugadores mueven sus fichas alternativamente sorteando el jugador que inicia el juego.
- Se mueve la ficha a una posición vacía en línea recta.
- Se captura una ficha contraria si al mover nuestra ficha:
 - La ficha contraria está en la misma fila que dos fichas nuestras.
 - Las fichas nuestras están juntas y además pegadas a la contraria.
 - Las otras dos posiciones de la línea están vacías.
- Ejemplos válidos de captura de ficha negra:
 - vacía-vacía-blanca-blanca-negra
 - vacía-negra-blanca-blanca-vacía
- Ejemplos no válidos de captura de ficha negra:
 - negra-vacía-blanca-blanca-negra
 - vacía-blanca-negra-blanca-vacía

Tablero:

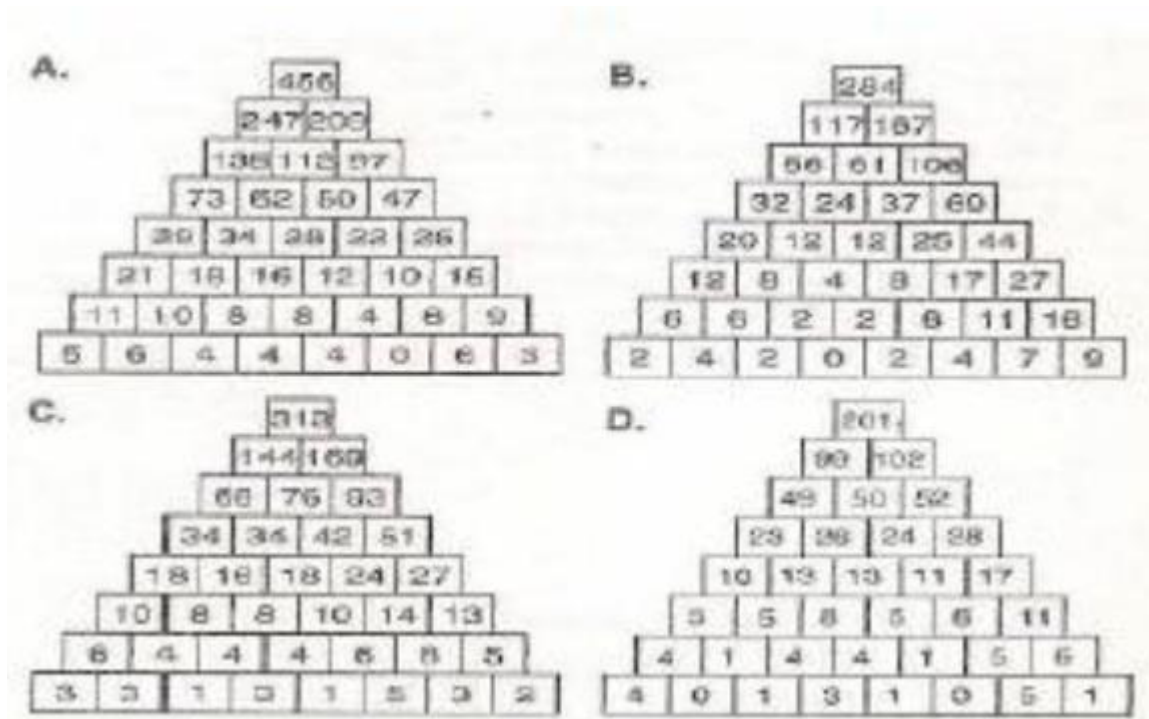


CENTRO DE CIENCIAS BÁSICAS
DEPARTAMENTO DE SISTEMAS ELECTRONICOS
ACADEMIA DE PROGRAMACION

| | | | |
|-----------------------|-------------------------|-------------------|------------------|
| Materia | LOGICA DE PROGRAMACION | Carrera | ING. EN SISTEMAS |
| Profesor | BLANCA ESTRADA RENTERIA | Semestre | 1° |
| PROYECTO FINAL | | | |
| Nombre del Proyecto: | Pirámide Algebraica | Fecha de Entrega: | |

Instrucciones Generales:

En muchas revistas de pasatiempos aparecen estos acertijos. Se trata de pirámides que se rellenan teniendo en cuenta que, en cada casilla, el número es la suma de los dos que se encuentra por debajo de él. Ejemplo:



El objetivo es programar este pasatiempo, elementos a considerar

1. Se le mostrara al usuario una pirámide con algunos números para que él se guíe
2. El deberá de llenar totalmente la pirámide
3. Una vez llenada la pirámide, tendrá que validar si la pirámide es correcta e indicarle al usuario si es correcta o no
4. De no ser correcta, se le debe permitir editar nuevamente la pirámide para hacer cambios, si no la pudo resolver, entonces debe mostrarle la solución.