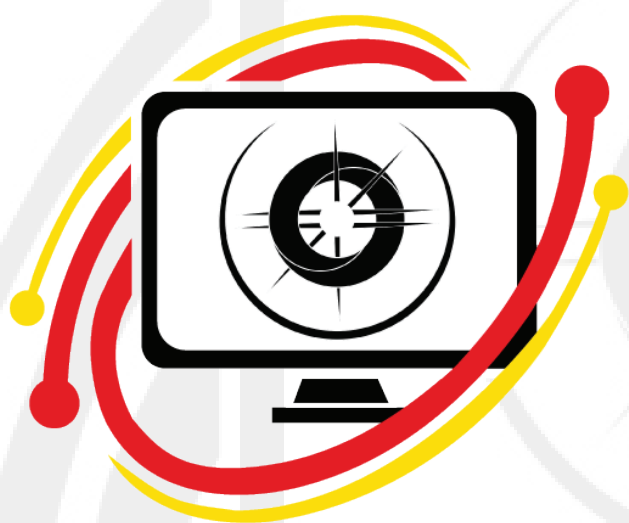


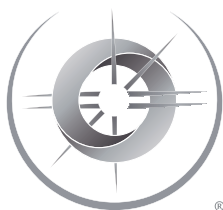
I Olimpiada aragonesa de informática (2021-2022)



Olimpiada Aragonesa de Informática

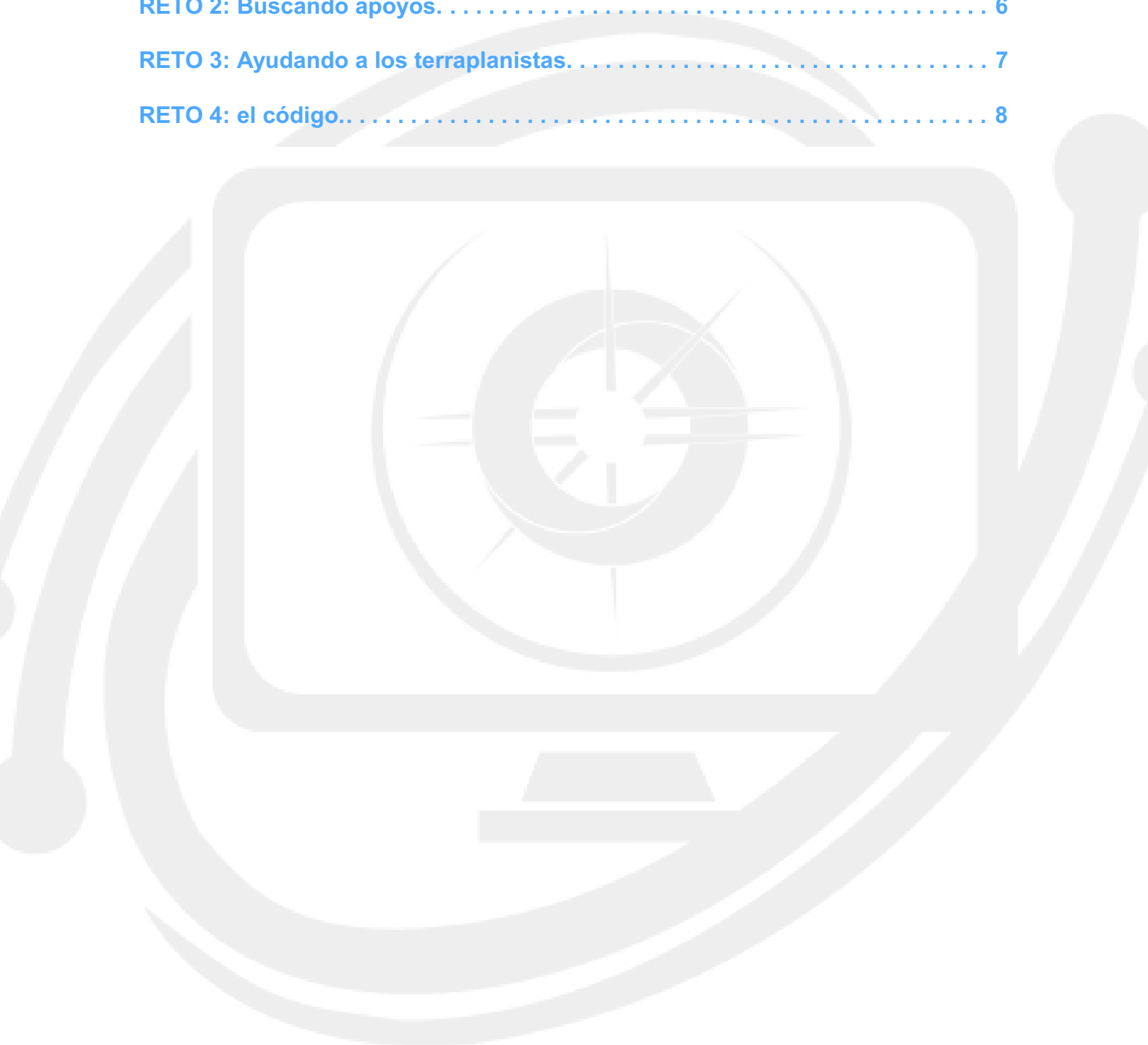
ALGORITMIA
1-2 secundaria

Organiza



Colegio Profesional
de Ingenieros Técnicos
en Informática de Aragón

| | |
|--|---|
| INTRODUCCIÓN..... | 3 |
| RETO 1: Velas solares..... | 5 |
| RETO 2: Buscando apoyos..... | 6 |
| RETO 3: Ayudando a los terraplanistas..... | 7 |
| RETO 4: el código..... | 8 |



INTRODUCCIÓN

La Agencia Espacial Española planea realizar una misión al planeta Marte con el objetivo de ser los primeros humanos en llegar allí. Es por ello que se desea reclutar a las mentes más agudas para que les ayude a resolver una serie de problemas que les ha surgido en distintas fases del proyecto.

Si estáis leyendo este documento estáis de enhorabuena, porque habéis sido preseleccionados para esta misión.

A lo largo de esta prueba os iremos enunciando una serie de problemas que deberéis resolver usando uno de los siguientes lenguajes de programación:

- C
- C++
- Java
- Python 3

Además de un lenguaje de programación, necesitareis vuestro ingenio del que estamos seguros que tenéis de sobra.

Antes de comenzar permitidnos unos consejos:

1. **ESTO NO ES UN EXAMEN**, vuestras notas no se van a ver afectadas por lo que hagáis aquí.
2. **NO HAY UN ORDEN PARA RESOLVER LOS RETOS**, si no encontráis la forma de solucionar un reto en **ESE** momento, no os frustréis, no os obcequéis, pasad al siguiente y cuando hayáis resuelto los demás, volved a ese.
3. **LEED ATENTAMENTE LOS ENUNCIADOS DEL RETO**, para resolver el problema tenéis que saber que es lo que se pide, como os van a pasar los datos y en que formato se pide la salida.
4. **¿CÓMO SE GANA?**, resolviendo el mayor número de retos posibles. Y entre los que resuelvan un problema, el que haga el programa **MÁS** óptimo.

5. No es necesario que siempre uséis el mismo lenguaje de programación, podéis resolver cada rato con el lenguaje que deseéis.



RETO 1: Velas solares.

Los instrumentos de las naves espaciales necesitan energía para poder funcionar. Para ello tienen que desplegar unos paneles solares inmensos que se encuentran plegados en la bodega de la nave.

Los paneles solares en el espacio son finos como una hoja de papel y también se pueden doblar por la mitad. Los ingenieros de la Agencia Espacial Española necesitan saber cuántos pliegues van a tener que hacer a los paneles para ocupar espacio van a ocupar el espacio que han destinado para ellos. Eso si, **SIN PASARSE**.

Para resolver el problema nos van a dar 2 números, ambos números entre 1 y 1000. El primero representa el grosor del panel solar medido en micras (10^{-6} metros) y el segundo es el tamaño del espacio en donde han pensado meter el panel doblado medido en metros.

Y la salida de nuestro programa será el número de veces que hemos tenido que doblar para llegar a ese tamaño.

Ejemplo:

| ENTRADA | SALIDA ESPERADA |
|---------|-----------------|
| 1 1 | 19 |
| 2 1 | 18 |
| 1 10 | 23 |

El programa no debería tardar más de un segundo en resolver el problema.

RETO 2: Buscando apoyos.

El coste de una misión a Marte es muy costoso, tanto económicamente como políticamente hablando. En la comunidad de Barataria, un político, llamado Filemón Pi, ha fundado un partido político de terraplanistas que intentan desbaratar la misión y buscan apoyos entre la población rápidamente. Su lema es “los votantes son sacrificables; sus votos, no”. Para asegurarse el voto de cada votante, envía a sus esbirros, Carpanta y Otilio, a entrevistar y convencer a cada persona de la comunidad para que no apoyen la misión espacial.

Ayer, Carpanta y Otilio encontraron a Tranquilino Fernández, un posible votante, en el páramo central de Barataria. Tranquilino no parece muy espabilado, con lo que puede ser un votante perfecto para el partido de Filemón. Para averiguar si es lo suficientemente poco espabilado, le han puesto un sencillo examen de aritmética: Tranquilino debe calcular la suma de varias parejas de números enteros positivos. Pero el problema es que Carpanta y Otilio tampoco saben mucho de matemáticas, así que el problema consiste en crear un programa que sumen dos números enteros positivos que pueden ser muy grandes.

Entrada: Se debe leer desde la entrada estándar lo siguiente: en la primera línea, el número de parejas de números que hay que sumar, X (tal que $1 \leq X \leq 100$). En las $2 \cdot X$ siguientes líneas habrá un entero positivo N en cada línea (de manera que $1 \leq N \leq 1030$).

Salida: Se deben escribir en la salida estándar X líneas conteniendo cada una la suma de una de las parejas de números leídas en dos líneas consecutivas de la entrada.

Ejemplo:

| ENTRADA | SALIDA ESPERADA |
|---------|-----------------|
| 3 | 4 |
| 2 | 10000000 |
| 2 | 101000 |
| 1 | |
| 9999999 | |
| 100000 | |
| 1000 | |

El programa no debería tardar más de un segundo en resolver el problema.

RETO 3: Ayudando a los terraplanistas.

Como ya hemos visto, es de conocimiento general que los terraplanistas que, además de afirmar que la tierra es plana, no son muy buenos en ciencias, sobretodo en matemáticas. Sin embargo, a pesar del esfuerzo de los sistemas educativos no consiguen que los terraplanistas avancen en sus capacidades matemáticas.

Es por ello, que el departamento de educación de la Agencia Espacia Española, en colaboración con las distintas Universidades de Aragón y la organización de las olimpiadas informáticas de Aragón han decidido colaborar para ayudar a los terraplanistas a mejorar en matemáticas y ya de paso que vean por sí mismos que la tierra es redonda.

Para ello han decidido entrenar a los terraplanistas mediante el cálculo mental. Para ello, un profesor les irá dictando una lista de números **CARDINALES DE UN SOLO DÍGITO**, que termina cuando dice “no hay más números”.

Tras terminar, el programa debe dar el resultado de la suma como un entero para que los alumnos puedan saber si han realizado correctamente el cálculo mental.

ENTRADA

Uno
Tres
Nueve
Cinco
Seis
Cero
No hay mas números

SALIDA ESPERADA

24

NOTA: Los alumnos no lo saben, pero el resultado de este problema es la solución el numero de fuerzas G que un astronauta puede llegar a soportar.

El programa no debería tardar más de un segundo en resolver el problema.

RETO 4: el código.

Se os va a facilitar una información súper secreta procedente de la Agencia Espacial Española (AEE), se trata de unos códigos que facilitan la ignición del cohete. Para ello se deberá extrayendo la letra correcta de cada palabra obtendrás el código de ignición del cohete.

Para ello deberéis escribir un programa un que elimine el enésimo carácter de una cadena de entrada.

Entrada: La primera línea de entrada contiene un único entero N ($1 \leq N \leq 1000$) que es el número de conjuntos de datos que siguen.

Cada conjunto de datos consta de una única línea de entrada que contiene M , un espacio y una cadena formada solo por letras mayúsculas y espacios. M será menor o igual que la longitud de la cadena. Se garantiza que la longitud de la cadena sea menor o igual a 80.

Salida: El programa debe devolver la cadena con el carácter eliminado.

Ejemplo:

| ENTRADA | SALIDA ESPERADA |
|-----------------|-----------------|
| 4 Estación | Estcion |
| 3 Satélite | Saelite |
| 5 Gravedad | Gravead |
| 1 Internacional | Nternacional |
| 2 Casco | Csco |

El programa no debería tardar más de un segundo en resolver el problema.



I OLIMPIADA

ARAGONESA DE INFORMÁTICA

- COLEGIO PROFESIONAL DE INGENIEROS TÉCNICOS
EN INFORMÁTICA DE ARAGÓN -

INSCRIPCIONES EN:
www.olimpiadaaragonesainformatica.es



Organiza:



Colegio Profesional
de Ingenieros Técnicos
en Informática de Aragón

Patrocina:



Colabora:



Escuela de
Ingeniería y Arquitectura
Universidad Zaragoza

universidad
SANJORGE
GRUPO SANVALERO



ESCUELA DE
ARQUITECTURA
Y TECNOLOGÍA



Escuela Universitaria
Politécnica - Teruel
Universidad Zaragoza



Escuela Universitaria
Politécnica - La Alfranca
Centro adscrito
Universidad Zaragoza



**GOBIERNO
DE ARAGON**