

# Votación Hexagonal

Autor: Ludwig Alvarado

Dentro de la Universidad Tecnológica de Algoritmos y Desarrollo Optimizado (UTADEO), se están organizando las próximas elecciones para representantes estudiantiles al consejo directivo. Debido a la polémica que hubo el año pasado por un posible fraude, la universidad decidió seleccionar 6 puntos físicos vigilados para que los estudiantes puedan ir a votar que se pueden ver en la figura 1.

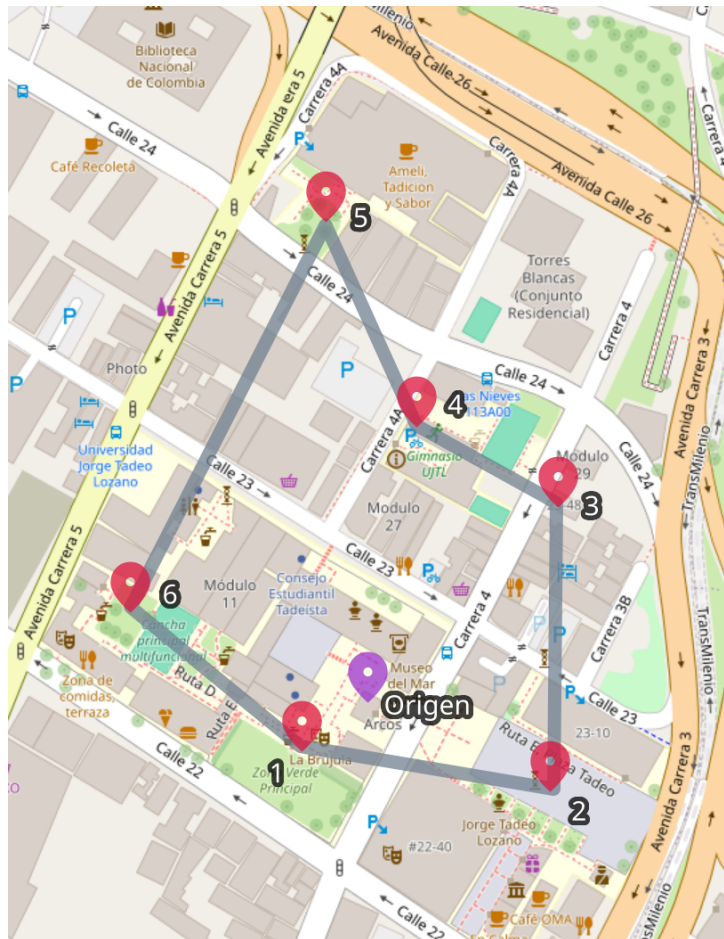


Figura 1: Puestos de votación disponibles, datos cartográficos de OpenStreetMap, edición por autoría propia.

Debido a que este año la universidad asignó un presupuesto mucho mayor, te contrataron a ti como programador para mostrar los resultados de las elecciones, distancia del puesto de votación más lejano al origen, y el Índice Energético de Votación (IEV) definido como:

$$IEV = \lceil \sqrt[3]{V} \rceil!$$

Donde  $V$  es la cantidad de votos totales en una mesa de votación. Para este año se postularon 3 estudiantes para ser el nuevo representante al consejo directivo de la universidad, ellos son; Andrea ( $A$ ), Bobby ( $B$ ) y Claudia ( $C$ ).

## Entrada del programa

Vas a recibir 6 líneas de datos, en cada línea vas a encontrar las coordenadas  $(x, y)$ ,  $(-1000 < x, y < 1000)$  de cada puesto de votación con respecto al origen  $(0, 0)$  seguido de la cantidad de votos que tuvo Andrea, Bobby y Claudia en esa mesa de votación,  $(1 \leq A, B, C \leq 500)$ .

## Salida del programa

Se deben imprimir el total de votos de todas las mesas de votación, la cantidad de votos que tuvo cada candidato en todas las mesas, el porcentaje de votos de cada candidato, la mesa más lejana al origen y el IEV del puesto de votación con más votos recogidos.

Se debe seguir el siguiente formato:

```
Total Votos: #numeroVotos
Votos Andrea: #votos #porcentaje
Votos Bobby: #votos #porcentaje
Votos Claudia: #votos #porcentaje
Mesa más lejana: #numeroMesa #distancia
IEV: #numeroMesa #valorIEV
```

## Caso de prueba de ejemplo

| Prueba #1         |                           |
|-------------------|---------------------------|
| Entrada           | Salida                    |
| -1 -1 78 50 90    | Total Votos: 2248         |
| 5 -1 100 90 115   | Votos Andrea: 798 35.5%   |
| 3 7 20 15 10      | Votos Bobby: 755 33.59%   |
| 0 8 40 55 30      | Votos Claudia: 695 30.92% |
| -2 11 470 450 370 | Mesa más lejana: 5 11.18  |
| -7 2 90 95 80     | IEV: 5 39916800           |

## Notas

La mesa de votación que tuvo más votos fue la número 5 con un total de 1290. Al sacar la raíz cúbica se obtiene  $\sqrt[3]{1290} \approx 10,8858...$ , al aplicar la función techo, es decir:  $\lceil \sqrt[3]{1290} \rceil = 11$ , al aplicar el factorial de este último número se obtiene el IEV:  $11! = 39916800$ .

## Aspectos Importantes

1. La prueba se resolverá en **2 momentos**: primero en papel, sin editor de código (**valdrá 3.0 puntos**). Segundo, pasará el algoritmo desarrollando en papel al editor de código (**valdrá 2.0 puntos**).
2. Hagan uso de los conceptos aprendidos hasta el tema de **ciclos**.
3. Los datos están insertados como un string, puede hacer uso de la función `split()` para obtener la información relevante, por ejemplo, si quiero guardar en tres variables diferentes la cadena `1 2 3`, puedo hacer:

```
1 >>> x, y, z = map(int, input().split())
2 >>> print(x, y, z)
3 1 2 3
4
```

4. Puede hacer el uso de las funciones `ceil` y `cbrt` (raíz cúbica) del módulo `math`, además de `round()`.
5. Si tienen dudas con respecto al uso de funciones pregunten al profesor si es permitida, de lo contrario si la usa y no es permitida (-1.0 puntos).
6. No importe ninguna librerías externas, no las necesita, en caso de hacerlo (-1.0 puntos).
7. No preocuparse por validaciones de errores, asuma que los datos se suministran de manera correcta.
8. Prestar especial atención al formato de entrada y al formato de salida. Cualquier diferencia respecto al formato de salida especificado se considerará como prueba fallida.