**The art gallery problem**

Anastopoulou Nicky (class B)  
Lygatsika Joanna-Maria (class A)  
Pilot School Faculty Varvakeion  
Athens, 2012

1. **Introduction**

The art gallery problem or museum problem is a well studied visibility problem in computational geometry. This raised from Victor Klee, American professor of mathematics in 1973.

The question we must answer is how many guards are needed (the sufficient and necessary number) to keep all together, any part of the museum at any time. We define as guard a person or a device. The guards are positioned at fixed points, they can observe in all directions at any distance, but cannot see through walls.  
The plan of the gallery or the museum is represented by a simple polygon and each guard is represented by a point of the polygon e.g.  a vertex.

Václav Chvátal (born 1946), Professor in the Department of Computer Science and Computer Software, Concordia University of Montreal, Canada, gave the first solution in 1975.

Chvátal's art gallery theorem gives an upper bound on the minimum number of guards. It states that guards are always sufficient (enough) and sometimes necessary to guard (keep) a simple polygon with n vertices.

Three years after the original solution of Chvátal, S. Fisk gave a very brief and significantly simple proof of the theorem. (Steve Fisk [1946-2010] was professor of mathematics at the Bowdoin College). His proof is regarded as exciting and breathtaking. The fact is that Fisk based his short proof on two important mathematical topics:

1. The triangulation of a polygon and
2. The claim that the graph of a triangulated polygon can be 3-colored.

The proof method proposed by Fisk is algorithmic. That was the occasion to extend the specific problem to a number of similar ones, with very interesting algorithmic problems and applications.

In this article we shall present a brief summary of the theoretical background of the problem. Moreover, we use an algorithm in the system of The Visual Module Vpython, Version 5.72, which gives the limit of Chvátal, using the method proposed by Fisk. This system was proposed to us by Professor John Emiris and doctor Fragoudakis Christodoulos of the University of Athens.

1. **The Problem**

The problem is called «**The art gallery problem» G(M)** and the question we need to answer is:

|  |
| --- |
| *Given a polygonal museum P, how many guards will be needed and where to place them, so we can be sure it will be guarded?* |

This is a complicated problem, in the newer versions of which, the guards can move along a wall or along a section between vertices. Also, the watching can be either within or outside the polygon.

The solution of the problem, as mentioned, will be done through an algorithm in the system of The Visual Module Vpython, Version 5.72, which gives the limit of Chvátal using the method of Fisk. Consequently, much of the difficulty of the problem involves the optimization of the algorithm.

The applications of the problem are numerous such as the intrusion detection in buildings, the air traffic control, the strategic planning of military missions, monitoring progress, designing computer games type ARCADE, as well as surgical applications.

Certainly in this article we do not go into technical details of the current evolution of the problem. We study the algorithm of placing guards and the question of the number of them, needed to guard a polygonal museum and give an implementation of the algorithm in Vpython.

* 1. **Introduction to the Problem: Consideration of**

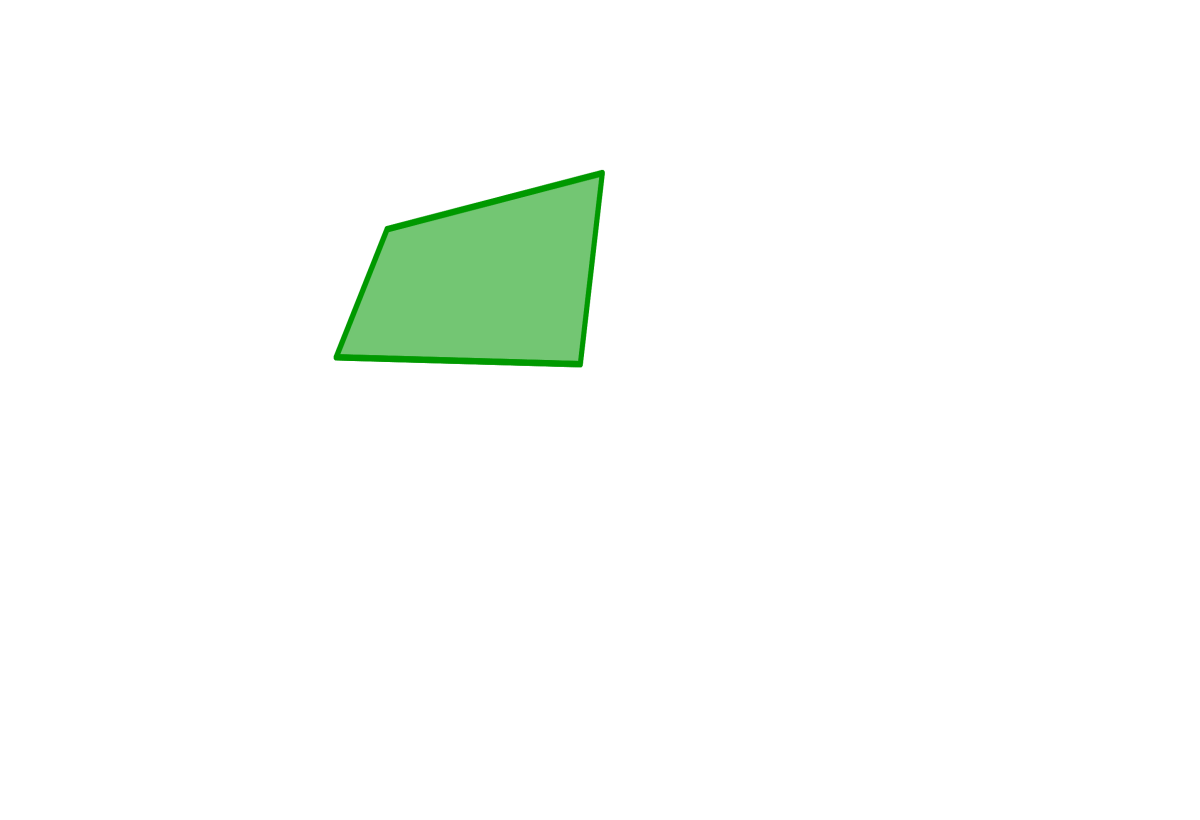
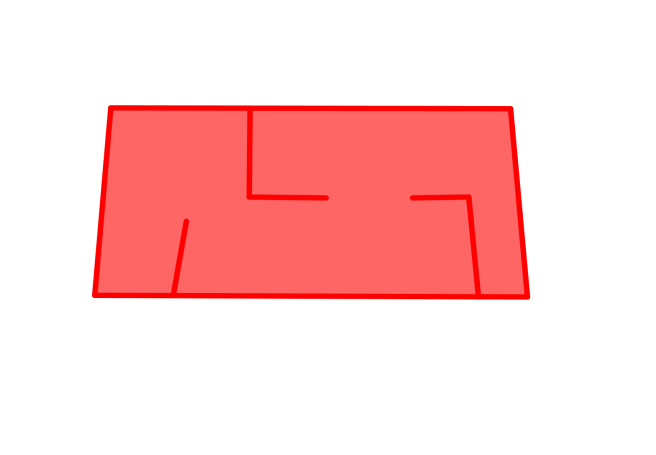
**problem’s data.**

**The Hall.**

We consider that the layout (plan) of the hall is the vertical projection of a three-dimensional object in two dimensions.

The model is represented by the museum's ground plan, which is a convex or non convex polygon, whose edges do not intersect each other. Of course the hall of the museum may have obstacles. For a real museum obstacles are the columns, walls which divide the hall, and the exhibits. For our problem, every obstacle will be faced as a "hole" of the polygon.

**Definition 1**: A polygon, whose the extension line of any edge leaves the polygon in the same semi plane is called **convex**, otherwise is called **non convex**. Furthermore, each polygon, convex or non convex, whose edges do not intersect each other, is defined as **simple**. Otherwise, it is defined as **non-simple**. **Pn** denotes the polygon, where **n** is the number of edges.



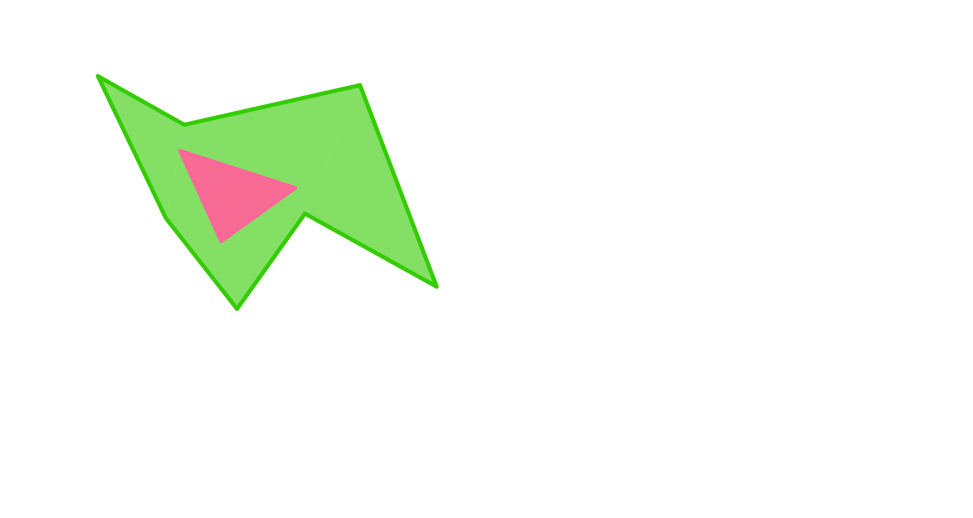


Figure 1: **Simple polygons, convex and non convex.**

**The Guard.**

A guard, in the present work, is a fixed point. A point can be rotated but not change position. There are, however, cases where the guard can be moved, but this is not the subject of this study.

**The Visibility.**

**The status of visibility**: A guard sees any point of the museum, if the segment line defined by the point and the guard and does not intersect any of the edges of the polygon.

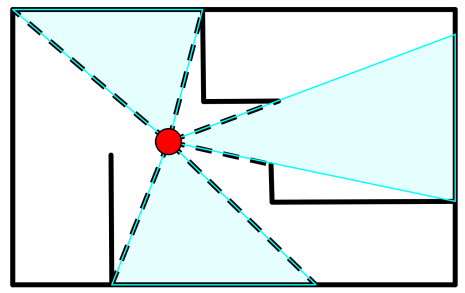
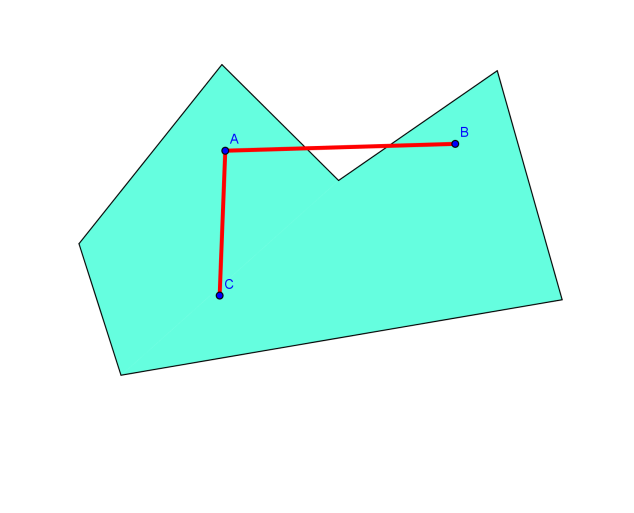


Figure 2: **Visibility of a guardian**



**For example**: In the figure 2a, point B of the polygon is not visible from the guard A because the segment AB intersects the edges of the polygon, while the point C is visible because the segment AC does not intersect any edge of the polygon.

Figure 2a

**Definition 2**: We say that **a room (hall) is been guarded**, when any interior point of it, is visible from at least one guard. The minimum number of guards that suffice to guard a room **Pn** will be denoted by **G(Pn)**.

**Types of Polygons and the Number of Guards.**

As we have mentioned, there are two types of simple polygons, the convex and non convex. Therefore, depending on the type of polygon, the necessary number of guard is calculated. Specifically:

* When the polygon **Pn** is **convex**, the minimum number of **G(Pn)=1**.

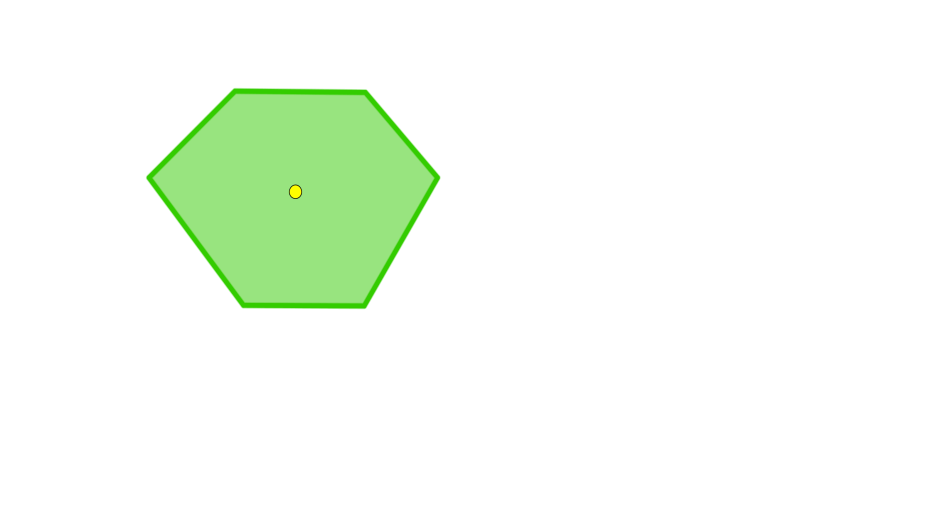


Figure 3: **The polygon Pn is convex, so the minimum number of G(Pn)=1.**

* When the polygon is **non convex**, the minimum number **G(Pn)** is calculated through the algorithm with the program of The Visual Module Vpython.

****

Figure 4: **In a non-convex n-gon, the G (Pn) is calculated by the algorithm.**

Following, we present the solution of the problem through the algorithm, i.e. the steps to be performed, which find the minimum number of guards, who are needed in all cases of a no-convex polygon.

* 1. **Solving the Problem: The steps of the**

**algorithm.**

The goal is to find a generalized procedure (i.e. an algorithm) for every possible form of convex or non convex n-gon, which gives the minimum number of G(Pn)and the positioning of guards, in any simply polygon.

The **steps of the algorithm** by which we solve the problem are:

1. Triangulation of the polygon
2. Design of dual graph
3. 3-coloring of vertices of the polygon
   * 1. **The triangulation strategy**

The main consideration is to simplify the problem. We could divide the non-convex polygon into convex polygons, and place a guard within each such polygon (see below figure 5).

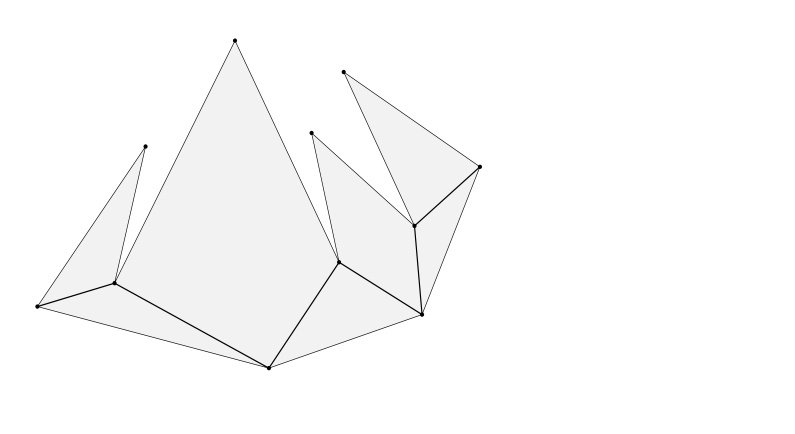
However, this is not always the best choice as shown in Fig.5. For this reason, we use smaller units, i.e. the simplest convex polygons, which are nothing else but the triangles. This method is called **triangulation**, i.e. the partition of a polygon into triangles. The role of the triangulation in this problem is as significant as the role of prime factorization in number theory.

Figure 5: **The partition of a polygon in polygons does not give a satisfactory answer to the problem G(M).**

**Definition 3: Triangulation of a polygon** is called a set of triangles with the following properties:

* The triangles are defined by the vertices of the polygon.
* The sum of triangles is equal to the polygon.
* Any two triangles either do not intersect, or have a common vertex, or a common edge.

The triangulation of the polygon is obtained by adding the diagonals joining two vertices of the polygon, excluding those which pass through external points.

**The Number of Distinct Triangulations.**

Generally, there is a large number of distinct triangulations in each polygon. The problem of finding that number started in 1751 by a letter of Euler to Goldbach.

The number of the triangulations of a convex polygon eventually was found by Gabriel Lamé (see Theorem 1). This number of triangulations of a polygon Pn found by him [L], for every n> 2, is called also **Catalan number**, since, virtually, the type who gives the number of distinct triangulations is a type of combinatorial theory that was found by Catalan.

**Theorem 1**: For any convex polygon of n vertices, the number of distinct triangulations is:

Where, ,μ, κ

**For example**: If the polygon has 4 vertices, then the number of distinct triangulations is:

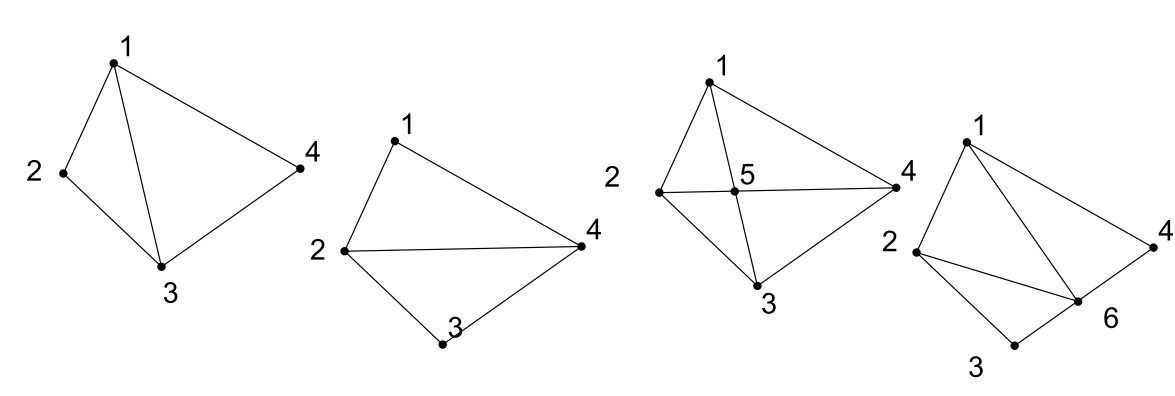


Figure 6: The two triangulations of a quadrangle.

In Figure 6 we see the two triangulations of a quadrilateral {(1, 2, 3) (1, 3, 4)} and {(1, 2, 4) (2, 3, 4)}.

There are special types of polygons, receiving a unique triangulation, e.g., a triangle or a non-convex quadrilateral.

**Simple Polygon Triangulation**

**Theorem 2**: Every simple polygon with n vertices can be triangulated. The triangulation contains n-2 triangles and n-3 diagonals.

(The proof of this theorem is not presented here. Note for historical reasons, that there is another proof of this, based on the type of Euler). [E]

The theorem 2 is necessary for the construction of the triangulation algorithm.

The best triangulation algorithm in terms of time and spatial complexity is one that has been described by O 'Rourke in [R] or [E]. In [E], the terminology is more friendly and clearer.

**The Partition into y-monotone Polygons.**

As mentioned at the beginning of this section, the aim is to divide the "room" (the non-convex polygon) into convex polygons, which can be triangulated easily, by drawing internal diagonals between vertices. However, the partitioning of a polygon into convex polygons is not feasible algorithmically. Thus, it becomes necessary to partition a non-convex polygon firstly in y-monotone polygons, a restricted class of polygons.

Note that a triangle is always a y-monotone polygon.

**Definition 4**: A polygon is called y-monotone, if any horizontal line, i.e. perpendicular to the y -axis (scanning line) has at most two common points with the edges of the polygon.

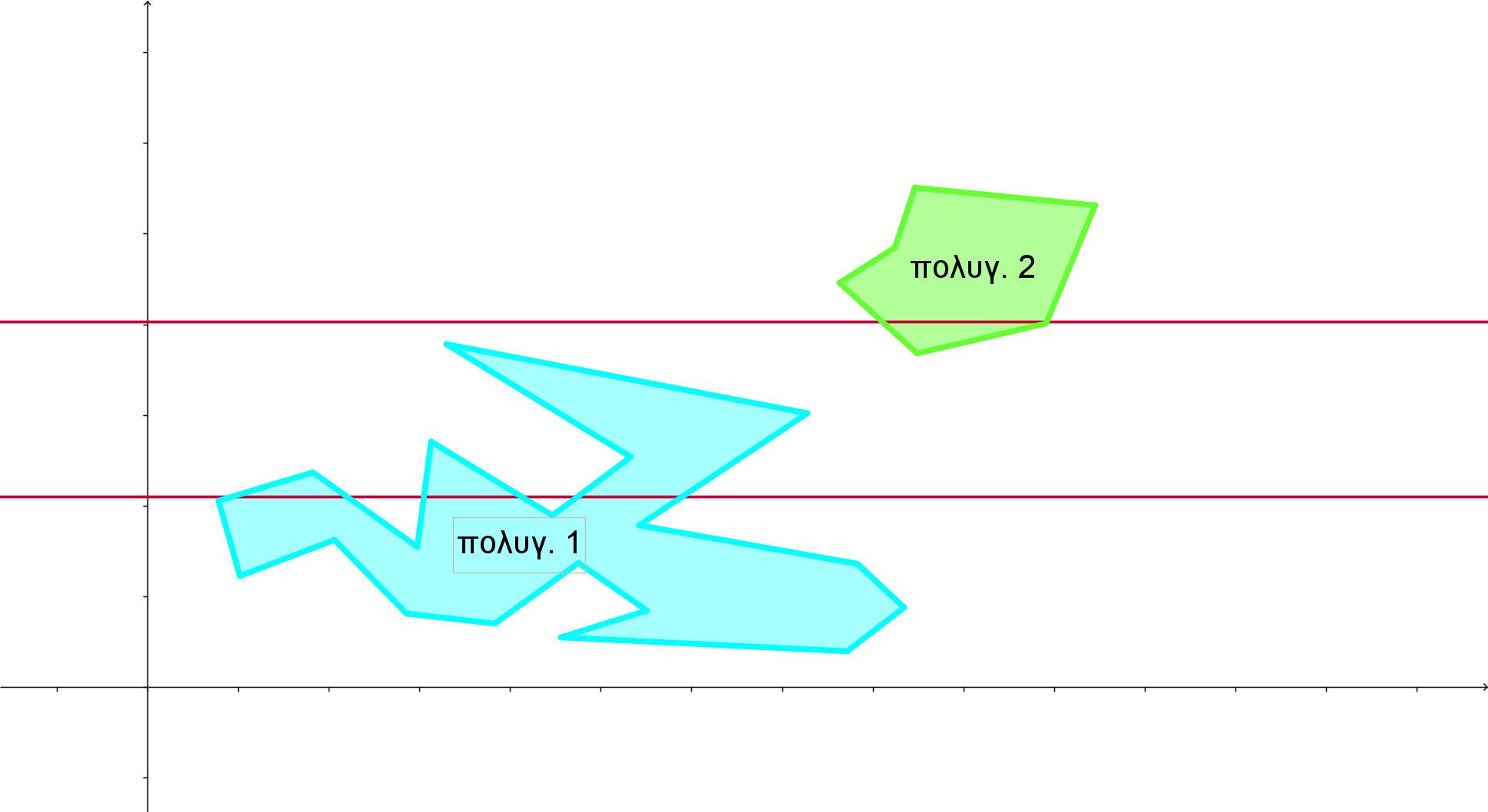


Figure 7: **The polygon 1 is not y-monotone while polygon 2 is y-monotone**

The scanning line starts moving from the vertex of the polygon with the largest ordinate. The scanning line moves downward, to the lowest vertex of the polygon, i.e. the scanning begins from large to small values of ordinates.

**Sorting of Polygon Vertices.**

* + 1. **Formation of Dual Graph.**

**Definition 5**: The dual graph is a structure consisting of **entities** and **relationships** that are formulated with points and edges.

The dual graph is useful for finding those vertices of the triangles, where we shall position the guards: which vertex we begin from and to which one we go after, etc.

The purpose of the formation of the dual graph is **the designation of a «path» in the polygon**.

To form the dual graph, we define a point in the interior of each triangle of the triangulation (not identical with its vertices or located on its edges). The triangles of the triangulation are called” nodes” of the graph. Then we join with a line segment, in the interior of the polygon, the points of two adjacent triangles, which share one common edge. That is, any common edge of two triangles will be crossed by a section (side) of the graph. A node is joined at least with one other node of the graph. The sides of the graph do not intersect the edges of the polygon and reflect the relationship of the two triangles in the interior of the polygon, i.e.to share a common edge.

The dual graph is always a «tree» i.e. an acyclic graph, because there is a unique path from one triangle of the triangulation to another, if the polygon is simple.

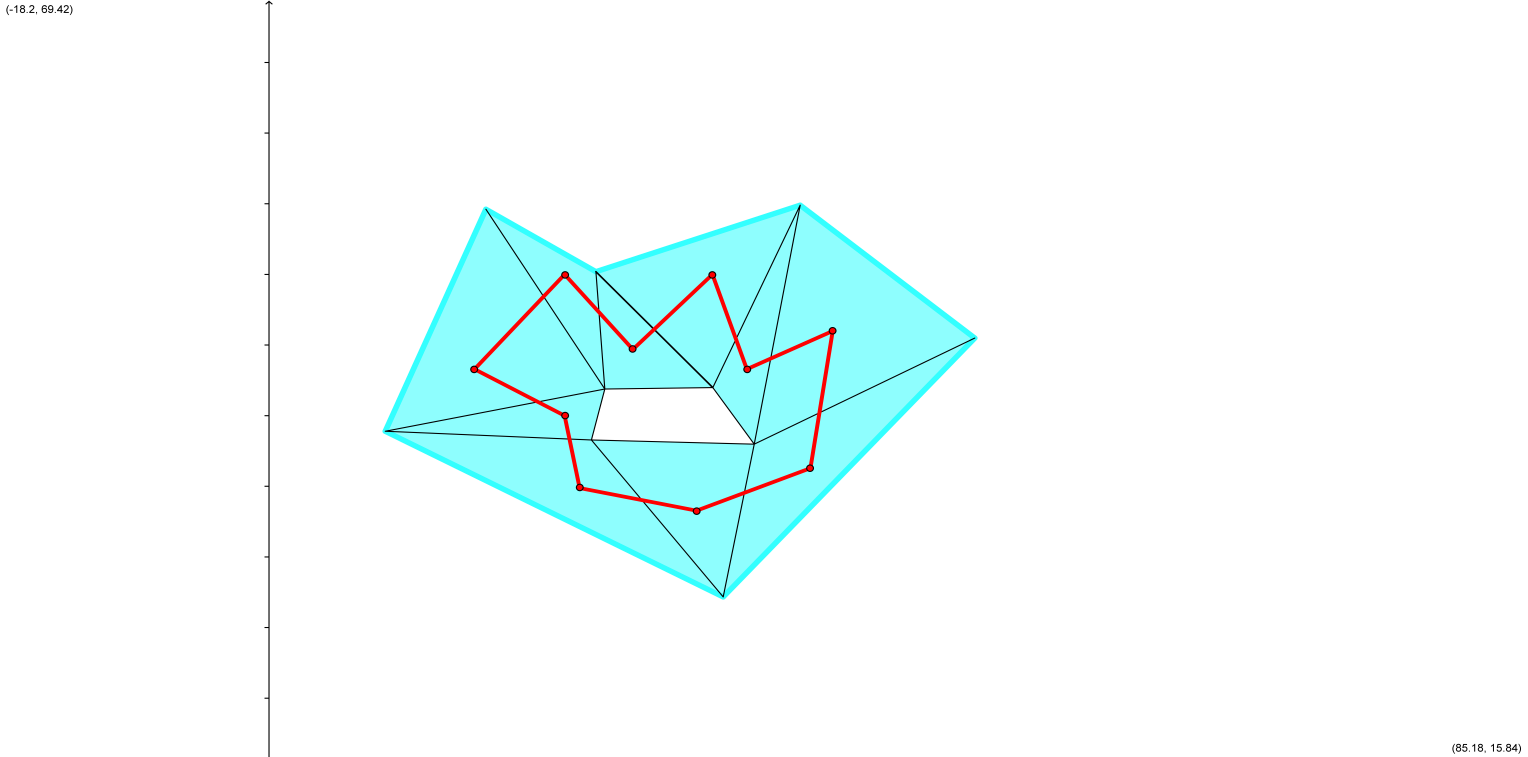
**Note**: If the polygon is not simple, i.e. when it has «holes» in its interior, its triangulation dual graph is a closed graph. That is, all points are joined together. In this case we call it «clique» (see figure 11).

Figure 11: **Representation of dual-graph «clique»**

In our example, the dual graph is as follows:

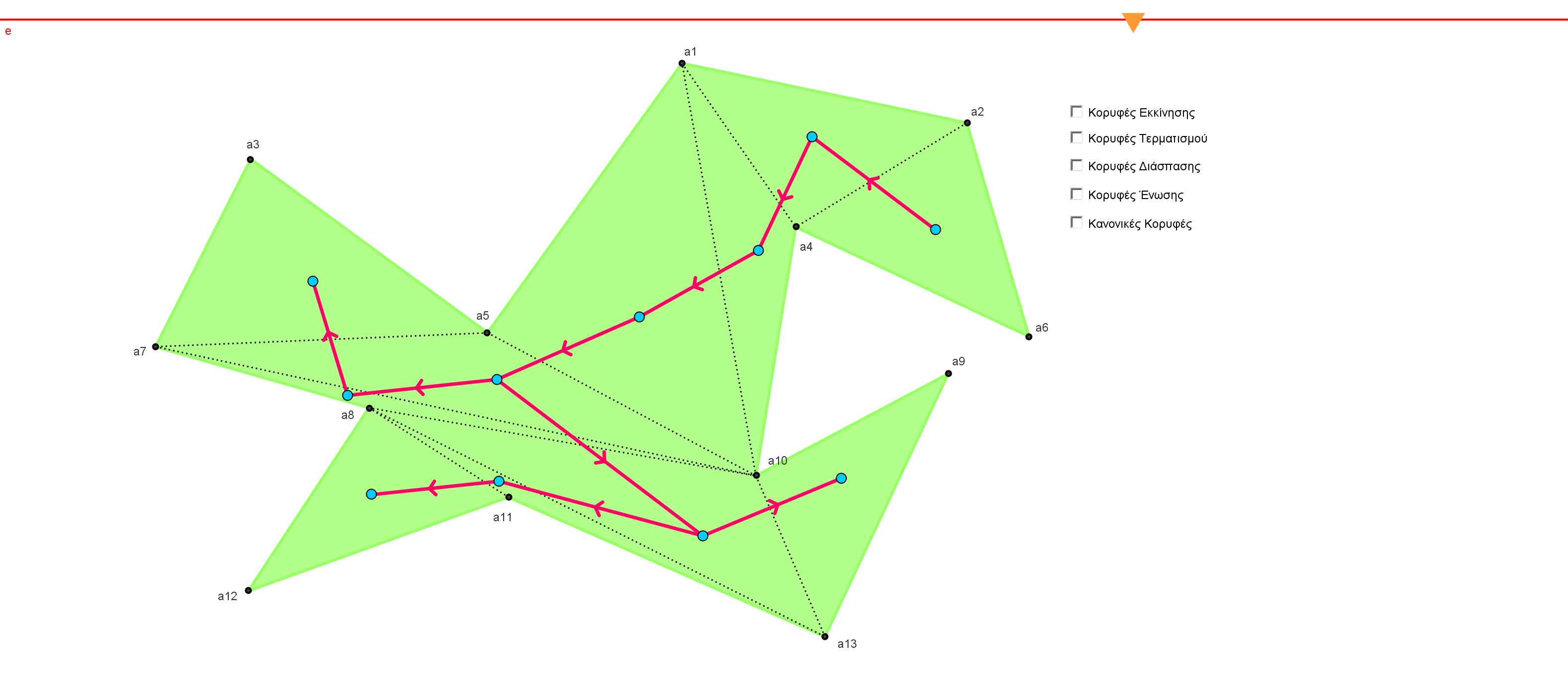


Figure 12: **Representation of the dual graph**

Based on the path designated by the dual graph we can begin the

3-coloring of vertices.

* + 1. **3-COLORING OF VERTICES.**

To find the vertices, in which we must place guards, we rely on the

3- Coloring of vertices of the polygon.

Starting from one of the nodes of the graph which shares only one edge, we continue to the next adjacent node, along the path that is indicated by the graph.

In each node, we color the three vertices in three different colors, so that any two adjacent vertices have different colors.  
Of course, we can go back to a previous node, but we will not change the color of the vertices of the triangle that have been already colored. In this way, we continue, until we have gone through all nodes of the graph, coloring all the vertices of the polygon.

**Note**: As we proceed to each next node (except the first), only one vertex is uncolored. This is because we have already colored the two vertices of the previous adjacent node, which shares a common edge and, therefore, two common vertices. It suffices to color the 3th vertex with the color which does **not** appear in the common edge.

Having 3-colored all vertices of the polygon, we find the number of occurrence of each of the three colors. **Our aim is to identify the color which appears the fewest times**. **The guards, positioned on the vertices, having that color, cover (keep) the entire museum.**

**Example**: in the figure below, after the triangulation of the polygon and the 3- coloring of the vertices, we count how many times each of the colors appears in the vertices:

**red color 4** times, **green color 4** times and **yellow color 3** times. The yellow one appears the less often, so we place the 3 guards on yellow vertices and the problem is solved with 3 guards.

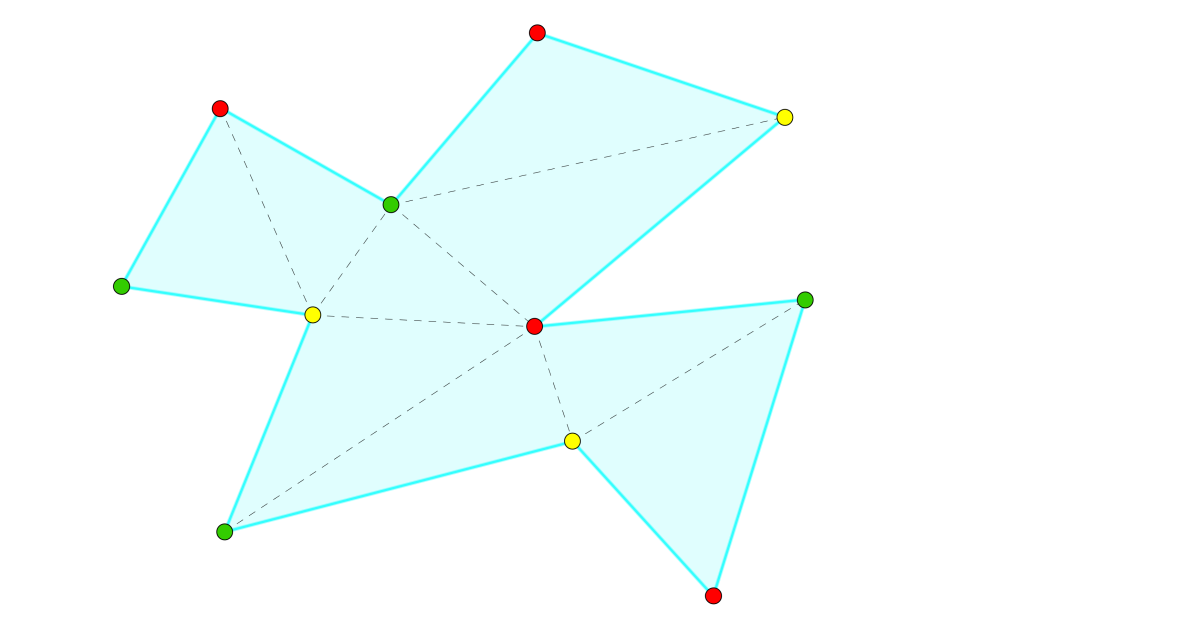


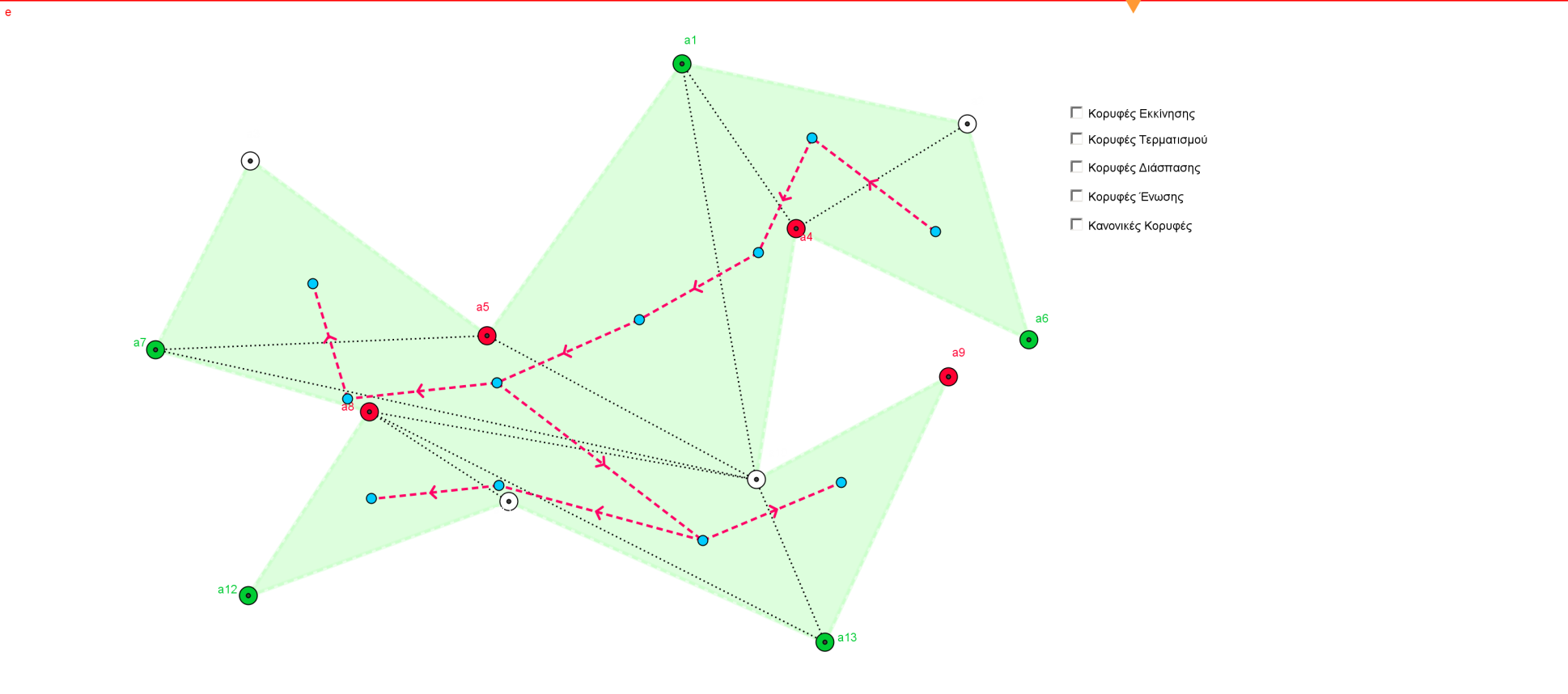
Figure 13: **The 3-coloring of the vertices of a   
 polygon.**

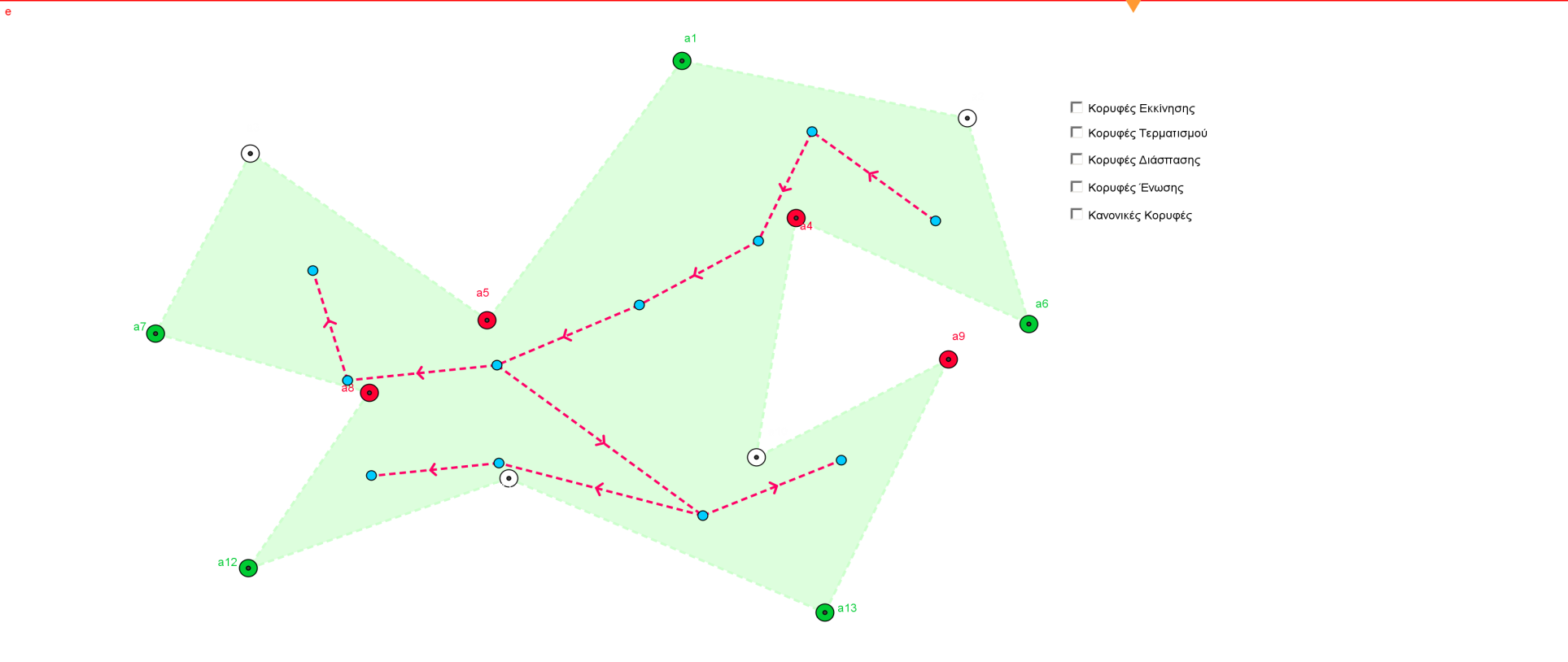
The edges of this polygon is n=11 and according to the theorem of Chvátal, that guards are always sufficient (enough) and sometimes necessary to (guard) keep a simple polygon with n vertices, we have. Thus, we see that through the process of 3-coloring (Fisk’ proof), the result coincides with that, given by the theorem of Chvátal.

In conclusion, the following lemma is powerful:

**Lemma**: It is always possible to 3-color a simple polygon with n vertices, and therefore to keep a polygonal museum, with guards at most.

In our example we have:



****

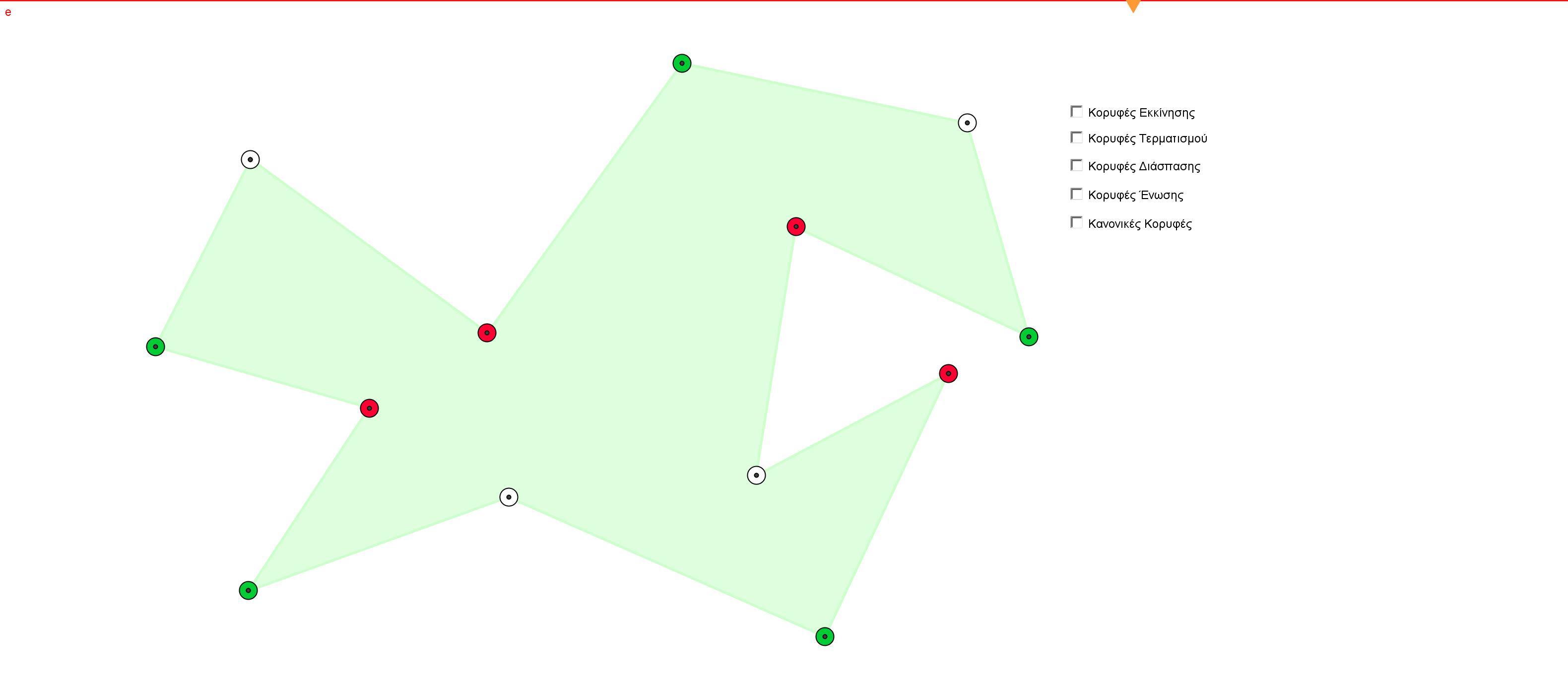


Figure 14: **The 3-coloring of vertices**

Counting how many times each of the three colors appears on the vertices, we have:

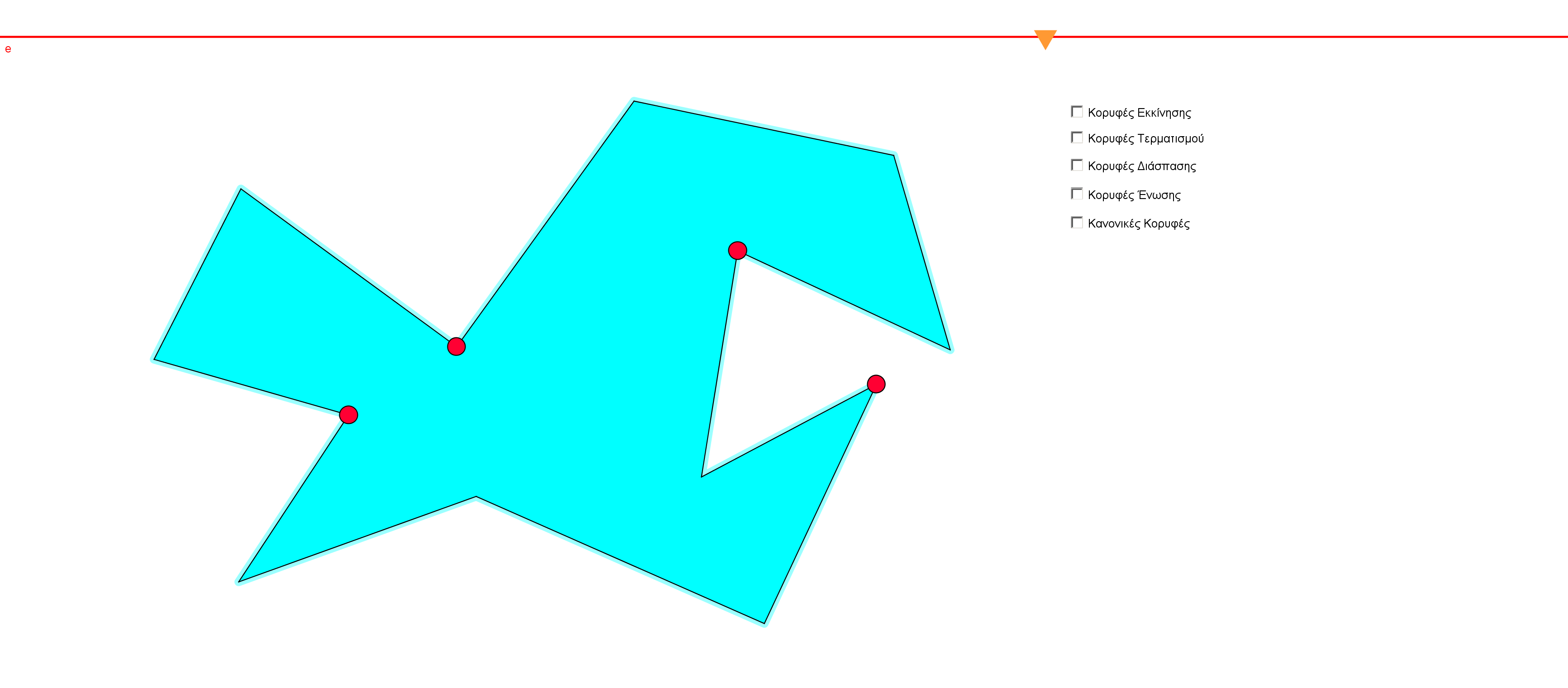
**green color 5** times, **white color 4** and **red color 4** times.

Therefore, the number of guards, which is necessary to keep the museum hall, is4, **G(Pn)=4**.

**The location in which we must place them is either white colored vertices or red colored vertices.**

|  |
| --- |
| Given any polygonal museum Pn, where n is the number of vertices, guards are sufficient to keep the museum and they must be placed on the vertices of the polygon, which are colored with that **color that appears the fewest times**. |

1. **THE SOLUTION OF THE PROBLEM**

1. **BIBLIOGRAPHY**

**[C]**: Chvátal V., A Combinatorial Theorem in Plane Geometry, Journal of Computorial Theory (B) 18 (1975), pp 39-41.

**[Ε]**: Εμίρης Ιωάννης, Υπολογιστική Γεωμετρία, Εκδόσεις Κλειδάριθμος, 2008.

**[F]:** Fisk S., A Short Proof of Chvátal’s Watchman Theorem, Journal of Combinatorial Theory, Series B 24 (1978) pg 374.

**[L]:** Lamé G., Un polygone convexe étant donné, de combine de manières peut-on le partager en triangles au moyen de diagonales? Journal de Mathématiques Pures et Appliquées, 3 (1838), 505-507.

**[LP]:** Lee D.T. and Preparata F. P., An optimal algorithm for finding the kernel of a polygon, ACM 26 (1979), 415-421.

**[LRS]**: Jesús A. De Loera, Jörg Rambau, and Francisco Santos, Triangulations, Algorithms and Computation in Mathematics, Springer-Verlag Berlin, Heidelberg, 2010.

**[R]:** O’Rourke Joseph, Art Gallery Theorems and Algorithms. Oxford University Press, 1987.