Solving Art Gallery Problem with Triangulation and Graph Coloring

Ensof Barhami 13516096
Program Studi Teknik Informatika
Sekolah Teknik Elektro dan Informatika
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia
13516096@students.itb.ac.id

***Abstract*—Art Gallery Problem is a well-known problem in computational geometry. It was first posed in 1973 by Victor Klee. The original problem asked for the minimum number of guards sufficient to see every point of the interior of an n-vertex simple polygon. In 1975 Václav Chvátal proved that $\left\lfloor \frac{n}{3} \right\rfloor$ guards are always sufficient for any n vertex simple polygon. Three years later Fisk gave a simple proof using graph coloring. This paper presents an implementation of Fisk Algorithm to solve Art Gallery Problem using simple Ear Clipping Triangulation Method and Greedy Graph Coloring which totally runs in quadratic (O(n^2)) time.**

***Keywords*—Art Galery Problem, Computational Geometry, Ear Clipping, Graph Coloring, Simple Polygon, Triangulation.**

I. INTRODUCTION

Art Gallery Problem is one of problem in computational geometry that extend its problem statement very fast. Several proofs were proposed in the same decade as that question was asked. In the beginning the original problem only asked about the static guards in a simple polygon, in how many guards is needed to cover a whole Art Gallery. It was then developped into more advances problem such as Guarded Guard Problem, 3-D Polytopes, Fortress Problem, and a complete visibility problem called Prison Yard Problem. The last problem is quiet similar with the original Art Gallery. The different is Prison Yard Problem is not only asked the visibility of interior polygon, but also the exterior. It is then solved with a really well-known Four-Color Theorem.

The first theorem proposed in 1975 by Václav Chvátal succesfully answer simple Art Gallery problem. It is stated that $\left\lfloor \frac{n}{3} \right\rfloor$ guards are always sufficient for any n vertex simple polygon. Steve Fisk then proved it with another approach with triangulation. But, in 1980 Kahn, Klawe, and Kleitman found that there is a special case in this problem. If the polygon is orthogonal only $\left\lfloor \frac{n}{4} \right\rfloor$ guards are always sufficient for any n vertex simple polygon. Later on, many theorems appear to solve modified Art Gallery Problem. However, this paper only focus on simple and non orthogonal polynom and solve it with the approach that use Graph Coloring and Triangulation with Ear Clipping method. Eventhough there is another Triangulation method called constrained Delaunay with divide and conquer approach (O($n\log n$)) which runs faster than Ear Clipping (O(n^2)), i prefer use Ear Clipping method from [1] due to it is easier to implement and recall that this problem does not receive large input. Even if a polygon has 1000 vertices this algorithm still runs under 1 second.

II. POLYGON, ART GALLERY, GRAPH COLORING, AND 2D TRIANGULATION

A. POLYGON

A polygon P is usually defined as a collection of n vertices $v_1, v_2, . . . , v_n$ and n edges $v_1v_2, v_2v_3, . . . , v_{n-1}v_n$ such that no pair of nonconsecutive edges share a point. We deviate from the usual practice by defining a polygon as the closed finite connected region of the plane bounded by these vertices and edges. The collection of vertices and edges will be referred to as the boundary of P, denoted by ∂ P; note that $\partial P \subseteq P$. The term "polygon" is often modified by "simple" to distinguish it from polygons that cross themselves [2].

Polygons can be divided into several classification as follow[4]:

1. Regular or Irregular

   A regular polygon has all angles equal and all sides equal, otherwise it is irregular

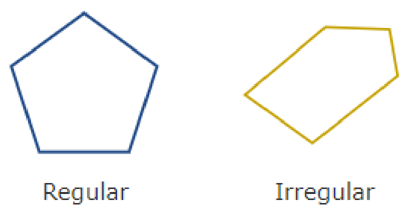

Figure 1. Regular and Irregular Polygon [4]

2. Concave or Convex

   A convex polygon has no angles pointing inwards. More precisely, no internal angle can be more than 180°. If any internal angle is greater than 180° then the polygon is concave.

Figure 2. Convex and Concave Polygon[4]

3. Simple or Complex

A simple polygon has only one boundary, and it doesn't cross over itself. A complex polygon intersects itself! Many rules about polygons don't work when it is complex.

Figure 3. Simple and Complex Polygon[4]

A. ART GALLERY PROBLEM

Art Gallery with n corners can be define as a simple polygon with n vertices or sometimes called n-gon.

Let us say that a point x ∈≤ P sees or covers a point y ∈ P if the line segment xy is a subset of P: xy ⊆ P. Note that xy may touch ∂ P at one or more points; that is, line-of-sight is not blocked by grazing contact with the boundary. For any polygon P, define G(P) to be the minimum number of points of P that cover all of P: the minimum k such that there is a set of k points in P, $\{x_1, . . . , x_k\}$, so that, for any y ∈ P, some x_i, $1 \leq i \leq k$, covers y. Finally, define g(n) to be the maximum value of G(P) over all polygons of n vertices[2].

Klee's original art gallery problem was to determine g(n): the covering points are guards who can survey 360° about their fixed position, and the art gallery room is a polygon. The function g(n) represents the maximum number of guards that are ever needed for an n-gon: g(n) guards always suffice, and g(n) guards are necessary for at least one polygon of n vertices. We will phrase this as: g(n) guards are occasionally necessary and always sufficient, or just necessary and sufficient[2][9].

B. GRAPH COLORING

Graph coloring has a common sub problem called vertex coloring. It is a problem that we need to perform coloring every vertices in a graph so that no two adjacent vertices have the same color. The problem usually asks what is the minimum color we need to perform such a task. The minimum color that we need is called Chromatic Number. From graph theory, we know that a complete graph of n vertices has n Chromatic Number. Unfortunately, we can not gently solve this problem if the graph is not a complete graph. Until this paper is written, there is no efficient algorithm for graph coloring problem as this problem is a known NP Complete Problem[3].

However, we are only talking about planar graph and tree in this paper. So the Chromatic Number is easier to determine. Because every random planar graph has the *maximum* Chromatic Number which is only 4. It is known as Four Color Theorem, the first problem that is proven by computer [6].

C. 2D TRIANGULATION

2D Triangulation is a division of planar graph into triangles. It is important to know that every polygon admit a triangulation.

Triangulation Theorem a polygon of n vertices may be partitioned into n - 2 triangles by the addition of n - 3 internal diagonals.

Proof The theorem is trivially true for n = 3. Let P be a polygon of n > 3 vertices. Let v2 be a convex vertex of P, and consider the three consecutive vertices v_1, v_2, v_3. (We take it as obvious that there must be at least one convex vertex.) We seek an internal diagonal d. If the segment v_1v_3 is completely interior to P (i.e., does not intersect ∂ P), then let d = v_1v_3. Otherwise the closed triangle v_1, v_2, v_3 must contain at least one vertex of P. Let x be the vertex of P closest to v_2, where distance is measured perpendicular to v_1, v_3 and let d = v_1x. In either case, d divides P into two smaller polygons P1 and P2. If Pi has ni vertices, i = 1, 2, then n1 + n2 = n + 2 since both endpoints of d are shared between P1 and P2. Clearly ni ≥ 3, i = 1,2, which implies that ni < n , i = 1, 2. Applying the induction hypothesis to each polygon results in a triangulation for P of (n1 - 2) + (n2 - 2) = n - 2 triangles, and (n1 - 3) + (n2 - 3) + 1 = n - 3 diagonals, including d [2].

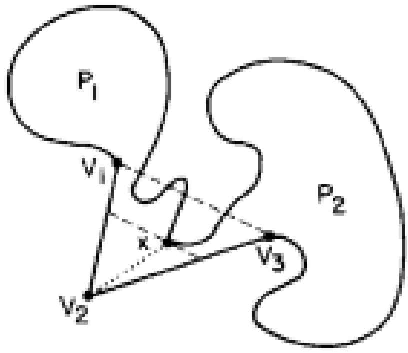

Figure 4. The line segment xv2 is an internal diagonal [2]

Note that the resulting triangulation of any simple polygon is a tree. No cycle will be produced because if it has a cycle, This cycle encloses some vertices of the polygon, and therefore it encloses points exterior to the polygon. This contradicts the definition of a simple polygon.

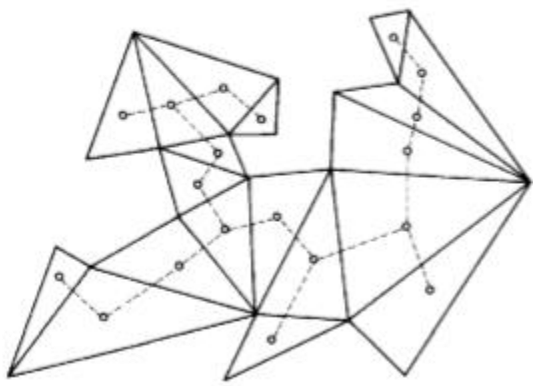

Figure 5. Triangulation of a polygon produces a tree[2]

One of a method for triangulation that will be used in this paper is Ear-Clipping method.

[7] An ear of a polygon is a triangle formed by three

consecutive vertices v_0, v_1, and v_2 for which v_1 is a convex vertex (the interior angle at the vertex is smaller than π radians), the line segment from v_0 to v_2 lies completely inside the polygon, and no vertices of the polygon are contained in the triangle other than the three vertices of the triangle. In the computational geometry jargon, the line segment between v_0 and v_2 is a diagonal of the polygon. The vertex v_1 is called the ear tip. A triangle consists of a single ear, although you can place the ear tip at any of the three vertices. A polygon of four or more sides always has at least two nonoverlapping ears. This suggests a recursive approach to the triangulation. If you can locate an ear in a polygon with n ≥ 4 vertices and remove it, you have a polygon of n−1 vertices and can repeat the process. A straightforward implementation of this will lead to an O(n^3) algorithm. With some careful attention to details, the ear clipping can be done in O(n^2) time. The first step is to store the polygon as a doubly linked list so that you can quickly remove ear tips. Construction of this list is an O(n) process. The second step is to iterate over the vertices and find the ears. For each vertex v_i and corresponding triangle $v_{i-1}v_iv_{i+1}$ (indexing is modulo n, so $v_n = v_0$ and $v_{-1} = v_{n-1}$), test all other vertices to see if any are inside the triangle. If none are inside, you have an ear. If at least one is inside, you do not have an ear. The actual implementation I provide tries to make this somewhat more efficient. It is sufficient to consider only reflex vertices in the triangle containment test. A reflex vertex is one for which the interior angle formed by the two edges sharing it is larger than 180 degrees. A convex vertex is one for which the interior angle is smaller than 180 degrees. The data structure for the polygon maintains four doubly linked lists simultaneously, using an array for storage rather than dynamically allocating and deallocating memory in a standard list data structure. The vertices of the polygon are stored in a cyclical list, the convex vertices are stored in a linear list, the reflex vertices are stored in a linear list, and the ear tips are stored in a cyclical list.

Once the initial lists for reflex vertices and ears are constructed, the ears are removed one at a time. If v_i is an ear that is removed, then the edge configuration at the adjacent vertices v_{i-1} and v_{i+1} can change. If an adjacent vertex is convex, a quick sketch will convince you that it remains convex. If an adjacent vertex is an ear, it does not necessarily remain an ear after v_i is removed. If the adjacent vertex is reflex, it is possible that it becomes convex and, possibly, an ear. Thus, after the removal of v_i , if an adjacent vertex is convex you must test if it is an ear by iterating over over the reflex vertices and testing for containment in the triangle of that vertex. There are O(n) ears. Each update of an adjacent vertex involves an earness test, a process that is O(n) per update. Thus, the total removal process is O(n^2).

III. Several Proofs on Art Gallery Theorem

A. Early Trial

Recall g(n) from previous topic. g(n) is the maximum number of guards that are ever needed for an n-gon: g(n) guards always suffice, and g{n) guards are necessary for at least one polygon of n vertices. If the polygon is regular, it is obvious that g(n) is always one. It is also obvious that g(3) = g(4) = 1 . The value of g(5) as drawn in figure 6 is also 1. What about g(6) ? It is already two guards are needed.

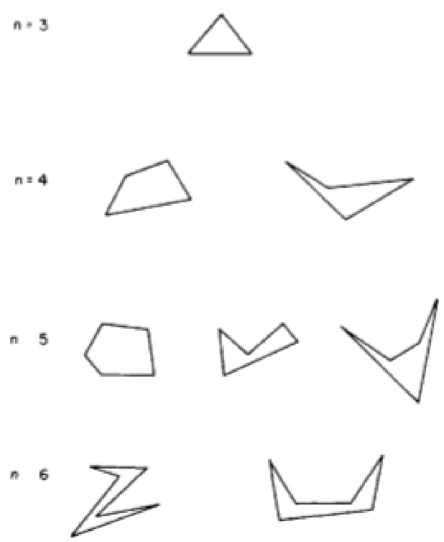

Figure 6. Polygons with 5 or fewer vertices can be covered by a single guard, but some 6-vertex polygons require two guards.[2]

Here is Chvátal who stated that the worst case shape of a polygon in this problem can be obtained if we draw a “comb” with k prongs or simply n = 3k vertices so that, there is always one sharp shape in every 3 vertices. This established that one guard will see 3 vertices at once. In the end, Chvátal said that g(n) ≥ $\left\lfloor \frac{n}{3} \right\rfloor$.

Figure 7. a comb with 5 prongs[2]

The formula g(n) ≥ $\left\lfloor \frac{n}{3} \right\rfloor$ could be interpreted as: one guard is needed for every three vertices. Phrased in this simple form, it is natural to wonder if perhaps a guard on every third vertex is sufficient. Figure 8 shows that such a simple strategy will not suffice: x_m in the figure will not be covered if guards are placed on all vertices i with i = m(mod 3). A second natural approach is to reduce visibility of the interior to visibility of the boundary: if guards are placed such that they can see all the paintings on the walls, does that imply that they can see the interior? Not necessarily, as Fig. 9 shows: guards at vertices a, b, and c cover the entire boundary but miss the internal triangle Q. A third natural reduction is to restrict the guards to be stationed only at vertices. Define a vertex guard to be a guard located at a vertex; in contrast, guards who have no restriction on their location will be called point guards. Define g_v(n) to be the number of vertex guards necessary and sufficient to cover an n-gon. Certainly there are particular polygons where the restriction to vertices weakens the guards' power: Fig. 10 shows one that needs two vertex guards but a single point guard placed at x suffices to cover the entire polygon. But g(n) summarizes information about all polygons, so this particular case has no more impact on our question than does the existence of n-gons needing only one guard have on the value of g(n). It turns out that in fact $g_v(n) = g(n)$ and that the reduction is appropriate[3].

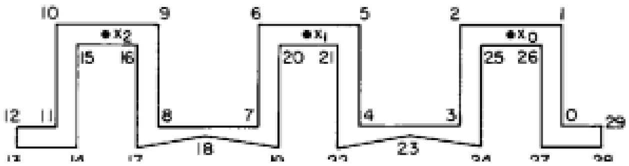

Figure 8.[2]

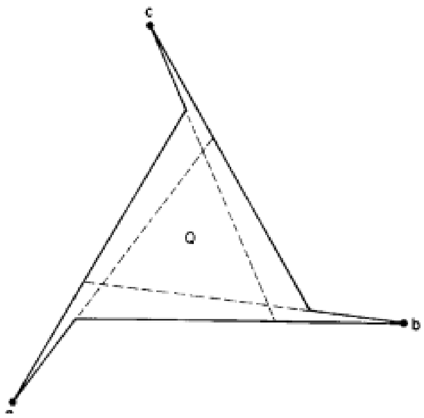

Figure 9.[2]

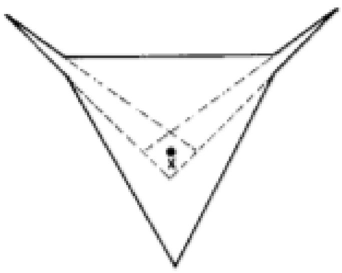

Figure 10.[2]

A. Václav Chvátal

Actually Chvátal proof also used triangulation similar with Fisk, but he use induction instead of graph coloring.

Define a fan as a triangulation with one vertex (the fan center) shared by all triangles. Chvatal took as his induction hypothesis this statement:

Induction Hypothesis: Every triangulation of an n-gon can be partitioned into g $\leq \left\lfloor \frac{n}{3} \right\rfloor$ fans.

For the basis, note that n > 3 since we start with an n-gon, and that there is just a single triangulation possible when n = 3, 4, and 5, each of which is a fan; see Fig. 11. Thus the induction hypothesis holds for n <6. Given a triangulation with n > 6, our approach will be to remove part of the triangulation, apply the induction hypothesis, and then put back the deleted piece. We will see in the next section that there is always a diagonal (in fact, there are always at least two) that partitions off a single triangle. But note that this only reduces n by 1, and if we were unlucky enough to start with n = 1 or 2 (mod) 3, then the induction hypothesis partitions into g = $\lfloor(\boldsymbol{n-1})/\boldsymbol{3}\rfloor = \left\lfloor \frac{\boldsymbol{n}}{\boldsymbol{3}} \right\rfloor$ fans, and we will in general end up with g + 1 fans when we put back the removed triangle. The moral is that, in order to make induction work with the formula [n/3j, we have to reduce n by at least 3 so the induction hypothesis will yield less than g fans, allowing the grouping of the removed triangles into a fan.

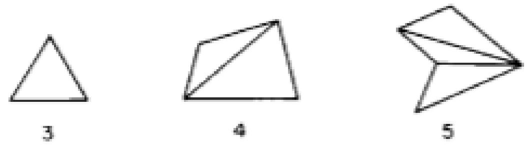

Figure 11[2]

The problem, it is not always there exist a diagonal that partitions off 4 edges of the polygon and reduces n by 3. But, Chvatal then came up with brilliant lemma

Lemma 1 For any triangulation of an n-gon with n > 6, there always exists a diagonal d that cuts off exactly 4, 5 or 6 edges.

After long applying induction hypothesis, Chvatal published Art Gallery Theorem in 1975.

Theorem 1 $\left\lfloor \frac{n}{3} \right\rfloor$ guards are occasionally necessary and always sufficient to cover a polygon of n vertices.

B. Fisk

Three years after Chvátal, Fisk (1978) came up with sufficiency proof with a very simple idea of Triangulation and 3-Coloring. This prove surely simpler than the previous proof proposed by Chvátal. Here are his steps to make sure that it is always sufficient to have $\left\lfloor \frac{\boldsymbol{n}}{\boldsymbol{3}} \right\rfloor$ guards in a simple n-vertices polygon.

Let say we have a symple polygon P with 10 vertices

1. Triangulate the simple polynom.

   As stated before in triangulation theorem, every polynom can be triangulated.

2. Color every vertices with only 3 colors. So that, no two adjacent vertices that have the same color.

   Why 3 colors ? because it is obvious that every two adjacent triangles inside the triangulated polygon share the same two vertices and of course we need two colors. Then, we only need to add one more color to the other vertex of each triangle. We can ensure that one other vertex doesnt meet in one point so we cant give the same color to them. With induction, we can do this coloring step with every two adjacent triangle greedily and use only 3 colors.

3. Choose the minimum number of total vertices that have the same color.

   For example in the resulting triangulated polygon we have 3 green vertices, 3 red vertices, and 4 black vertices. Then we can choose either green or red.
4. Place guards in every vertices with the choosen color.

Why does this step works ?

Let a, b, and c be the number of occurrences of the three colors in a coloring, with a < b < c. The total number of nodes is n, so a + b + c = n. If a > $\frac{n}{3}$, then the sum of all three would be larger than n. Therefore, a < $\left\lfloor\frac{n}{3}\right\rfloor$ (since a must be an integer). We know that our 10-vertices triangulated polygon's the least frequently used color is red. Since a triangle is the complete graph on three nodes, each triangle has all three colors at its vertices. Thus every triangle has a red node and thus a guard in one of its corners. Moreover, since the triangles form a partition of P, every point in the polygon is inside some triangle, and since triangles are convex, every point is covered by a red guard. Thus the guards cover the entire polygon, and there are at most $\left\lfloor\frac{n}{3}\right\rfloor$ of them. This establishes that $\left\lfloor\frac{n}{3}\right\rfloor$ guards are sufficient to cover the interior of an arbitrary polygon. Together with the necessity proved earlier, we have that g(n)= $\left\lfloor\frac{n}{3}\right\rfloor$.

IV. Modified Art Gallery Problem

There are lot of family of the original problem of art gallery[5].
Here are some problem that i already observe from [2][5] .

1. Orthogonal Art Gallery

The same with art galley, but the polygon is always orthogonal. It means that every vertex have 90 degree angle. The solution is given 1980 by Kahn, Klawe, Keitman with Quadrilateralization . Stated that g(n) is $\left\lfloor\frac{n}{4}\right\rfloor$.

2. Fortress Problem

It is asked how many guards are needed to guard exterior part of a polygon. g(n) is $\left\lceil\frac{n}{2}\right\rceil$ for regular one, and $\left\lceil\frac{n}{4}\right\rceil$+1 for orthogonal shape.

3. Prison Yard Problem

It is a combination of the original Art Gallery and Fortress Problem. How many guards are needed to cover interior and exterior of a polygon. There are several solution in this problem, most of it using a very well known Four Color Theorem as a basic concept. Because after triangulation, there lots of "vertices" outside the polygon that needs to be covered. The adding vertices allow our triangulated polygon being a planar graph. Then it is obvious that Four Color theorem will be used.

4. Modified the Guard State

Other variations on guards include:

   a. Point guard means the guard can be placed anywhere in the polygon.
   b. Edge guard means the guard can be placed anywhere along an edge of the polygon.
   c. Mobile guard means the guard is allowed to patrol along a line segment lying in the polygon.

5. Polygon with holes

   If h is the number of holes in a polygon, then

   a. O'Rourke (1987) stated that ⌊(n+2h)/3⌋ vertex guards are sufficient for a polygon with n vertices and h holes.
   b. Hoffmann, Kaufmann, Kriegel 1991 : ⌊(n+h)/3⌋ point guards are sufficient for a polygon with n vertices and h holes.

IV. Implementation of Fisk Theorem

I use algorithm in [1] to perform triangulation. I modified some code and add a backtracking algorithm to perform 3-coloring in C language from [8] and modified some function. Triangulation that is used here is Ear-Clipping method.

Algorithm :

1. Initialize the ear tip status of each vertex.
2. While n > 3
   a. Locate an ear tip v2
   b. Save v1 and v3 as diagonal (adjacent vertex)
   c. Delete v2.
   d. Update ear tip of v1 and v3.
3. coloring triangulated polygon.
4. Place guards in the minumum vertex with the same color.

Here are some function needed in this algorithm

1. Ear Init().
   Initializes the data structures, and a preparation to clip off the ears one by one.

2. bool diagonal(a,b)
   Returns true iff (a,b) is a proper internal diagonal.

3. bool diagonalie(a,b)
   Returns TRUE iff (a,b) is a proper internal *or* external diagonal of P, *ignoring edges incident to a and b*.

4. bool lefton,left,collinear (a,b,c)
   check whether the posision of c is strictly to the left of the directed line through a to b.

5. bool between (a,b,c)
   Returns TRUE iff point c lies on the closed segement ab. First checks that c is collinear with a and b.

6. bool intersect (a,b,c,d)
   Returns TRUE iff segments ab and cd intersect, properly or improperly.

7. bool intersectprop (a,b,c,d) Returns true iff ab properly intersects cd: they share a point interior to both segments. The properness of the intersection is ensured by using strict leftness.

V. Application

Computational geometry has a wide application such as :

- route planning for GPS: determining location, speed, and direction
- integrated circuit design
- designing and building objects such as cars, ships, and aircraft

- computer vision, to determine lines of sight and designing special effects in movies
- robotics, to plan motion and visibility.

Art Gallery Problem and all its family as a computational geometry problem itself can be used in optimization problem such as planning GPS, and of course an application to its original problem is to minimize the guards (can be anything) needed to cover a big system or environtment.

V. Conclusion

There lots of algorithm can be used to solve Art Gallery Problem. 3-coloring and Triangulation used in this paper also has some algorithm that can be implemented. I prefer to use the easiest to understand and implement. That is why the implementation leaves much room for improvement, especially in time complexity. However, proofs and algorithm discussed in this paper can be used to solve a well-known computational geometry problem, Art Gallery Problem.

VI. Appendix

The author's implementation of the algorithm discussed in this paper that is from [1], [2] and [8] can be accessed on Github (https://github.com/ebarhami/ArtGalleryProblem). It is written in C. The copyright of the code from [1] and [8] also is still written in the code.

.

VII. Acknowledgment

The writer thanks Dr. Judhi Santoso M.Sc. as the writer's lecturer in Discrete Mathematics class, for guidance in preparing this paper and teaching materials especially graph theory in Discrete Mathematics. Writer also would like to thank Mr Joseph O'Rourke, the author of Art Gallery Theorems and Algorithms. The book has greatly help in preparing this paper.

References

[1] O'Rourke , Joseph ,Computational Geometry in C Second Ed. Baltimore : Cambridge University Press, 1987, pp 1-30
[2] O'Rourke , Joseph , Art Gallery Theorems and Algorithms. Baltimore : Oxford University Press, 1997, pp 1-44
[3] http://www.geeksforgeeks.org/graph-coloring-set-2-greedy-algorithm/ (Retrieved December 3rd 2017)
[4] https://www.mathsisfun.com/geometry/polygons.html (Retrieved December 3rd 2017)
[5] D.G. Aliaga, "Art Gallery Theorems and Algorithms, Computer Science Department Purdue University Presentation Slides" unpublished.
[6] http://mathworld.wolfram.com/Four-ColorTheorem.html (Retrieved December 1st 2017)
[7] Eberly, David, "Triangulation by Ear Clipping" unpublished
[8] http://www.geeksforgeeks.org/backttracking-set-5-m-coloring-problem/ (Retrieved December 4th 2017)
[9] S. Halim and F. Halim, Competitive Programming 3. Singapore: Lulu, 2013, pp. 269-271 and 338.

Pernyataan

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 4 Desember 2017

Ttd

Ensof Barhami 13516096