

Một số bài tập lập trình với ASY

Tips 04 2023 ...

Bài viết sẽ mô tả việc lập trình để vẽ hình với ASY.

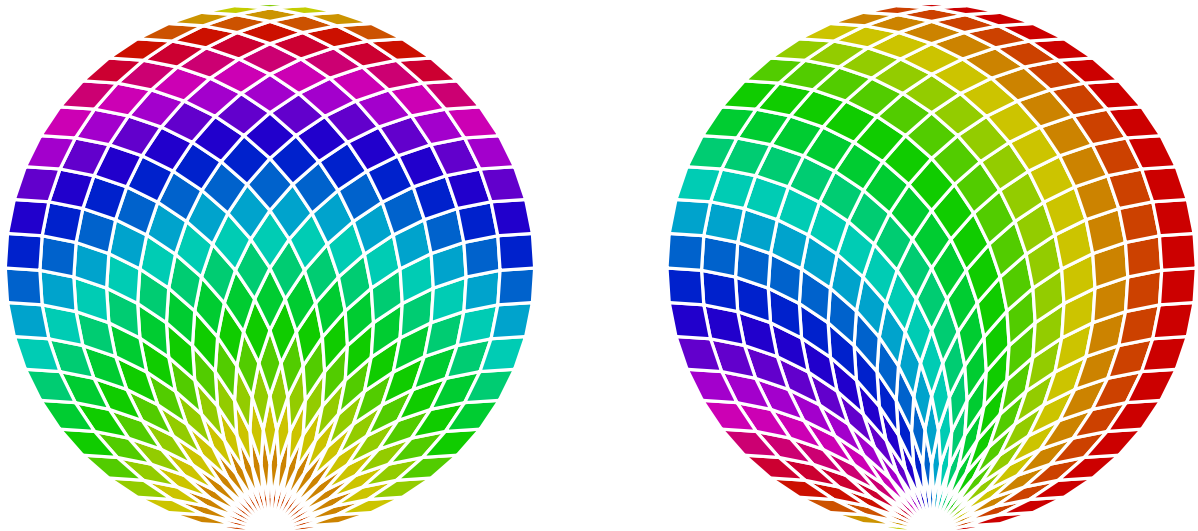
1 Solution

P1. Code hoàn chỉnh của bài kỳ trước:

[1]. Code của PT Sinh

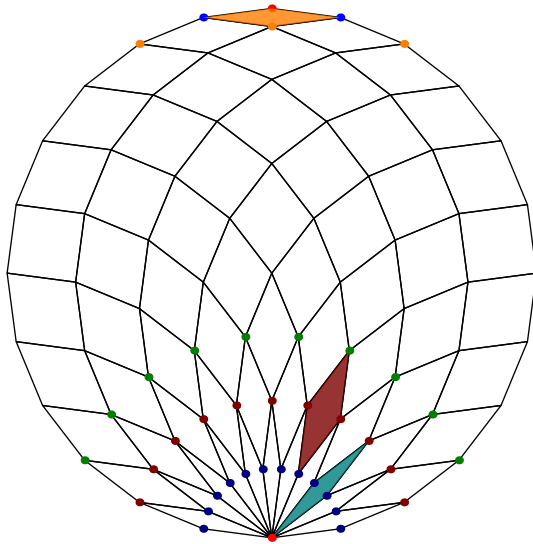
[2]. Code của Trần Quân

P2. Vẽ hình sau:



Trước khi vẽ hình này cần phân tích:

- Đa giác ngoài cùng là đa giác đều có $2n$ cạnh.
- Các đa giác cần tô màu là hình bình hành.
- Nếu lấy mốc từ đỉnh trên cùng của đa giác đều (đa giác vòng 1) thì từ 3 đỉnh đầu tiên sẽ xác định được đỉnh của đa giác vòng 2. Từ đó có thể dùng đệ qui để xác định các đỉnh còn lại.
- Nếu lấy mốc từ đỉnh dưới cùng của đa giác đều thì ta có thể xác định các đỉnh của hình bình hành theo vòng lặp. Chúng ta sẽ thử cách này.



```
pair [] vertex(pair O=(0,0), real r=1, int n, int k) {
    pair X=(O.x, O.y-r);
    pair [] V;
    for (int i=0; i<2n; ++i) V.push(rotate(i*180/n, O)*X);

    pair [] VA;
    real angle = (180-180/n)/(n-1);
    for (int i=0; i<=n-k; ++i) VA.push(rotate(angle*i, X)*V[k]);
    return VA;
}
```

Hàm `vertex(pair O=(0,0), real r=1, int n, int k)` sẽ trả về các đỉnh của hình bình hành với các tham số là tâm của đa giác đều, số cạnh $2n$ của đa giác đều. Để dễ hình dung có thể vẽ các đỉnh từ hàm này, ví dụ:

```
unitsize(3.5 cm);

pair [] vertex(pair O=(0,0), real r=1, int n, int k) {
    pair X=(O.x, O.y-r);
    pair [] V;
    for (int i=0; i<2n; ++i) V.push(rotate(i*180/n, O)*X);

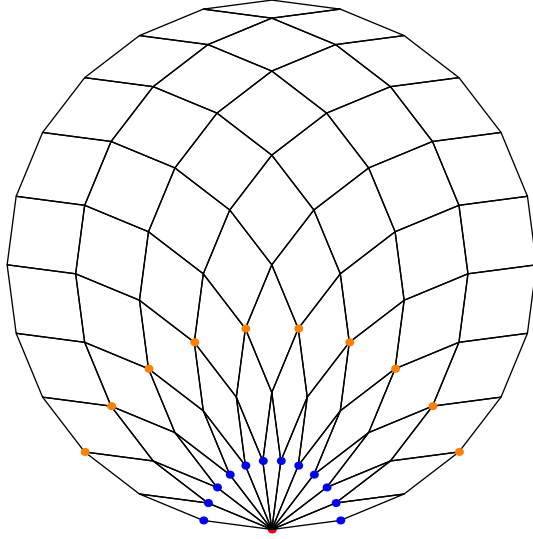
    pair [] VA;
    real angle = (180-180/n)/(n-1);
    for (int i=0; i<=n-k; ++i) VA.push(rotate(angle*i, X)*V[k]);
    return VA;
}

real r=1;
pair O=(0, 0);
int n=12;
pair X=(O.x, O.y-r);
pair [] V;
for (int i=0; i<2n; ++i) V.push(rotate(i*180/n, O)*X);
V.cyclic=true;

pair [] V1 = vertex(O, r=r, n=n, k=1);
for (int i=0; i<V1.length; ++i) dot(V1[i], blue);

pair [] V3 = vertex(O, r=r, n=n, k=3);
for (int i=0; i<V3.length; ++i) dot(V3[i], orange);
```

Trong đoạn code trên, ta khai báo các tham số cơ bản để sử dụng hàm `vertex`. Kết quả như sau:



Từ hàm **vertex**, ta viết tiếp hàm thứ 2 để fill các hình bình hành.

```
void draw_poly_rhombus(pair O=(0,0), real r=1, int n, bool direction=true) {
    pair X=(O.x, O.y-r);
    pair [] V;
    for (int i=0; i<2n; ++i) V.push(rotate(i*180/n, O)*X);
    V.cyclic=true;

    pair [][] VX;
    for (int i=0; i<=n; ++i) VX[i]=vertex(O, r, n, i);

    for (int i=1; i<n; ++i)
    for (int j=0; j<VX[i+1].length; ++j) {
        path pa=VX[i-1][j+1]--VX[i][j]--VX[i+1][j]--VX[i][j+1]--cycle;
        int k=i;
        if (direction==false) k=j;
        filldraw(pa, hsv(0.8*k*n,1,0.8), white+1.2bp);
    }
}
```

Hàm *draw_poly_rhombus* vẫn có các tham số chính là tâm đa giác đều $2n$ cạnh, bán kính đa giác đều, và thêm tham số **direction** để tô màu hình bình hành theo hướng nào. Việc tô màu dùng hàm **hsv**, giá trị màu sẽ phụ thuộc vào biến k .

Như vậy đoạn code cuối cùng như sau:

```
unitsize(3.5cm);

pair [] vertex(pair O=(0,0), real r=1, int n, int k) {
    pair X=(O.x, O.y-r);
    pair [] V;
    for (int i=0; i<2n; ++i) V.push(rotate(i*180/n, O)*X);

    pair [] VA;
    real angle = (180-180/n)/(n-1);
    for (int i=0; i<=n-k; ++i) VA.push(rotate(angle*i, X)*V[k]);
    return VA;
}

void draw_poly_rhombus(pair O=(0,0), real r=1, int n, bool direction=true) {
    pair X=(O.x, O.y-r);
    pair [] V;
    for (int i=0; i<2n; ++i) V.push(rotate(i*180/n, O)*X);
    V.cyclic=true;

    pair [][] VX;
```

```

for (int i=0; i<=n; ++i) VX[i]=vertex(0, r, n, i);

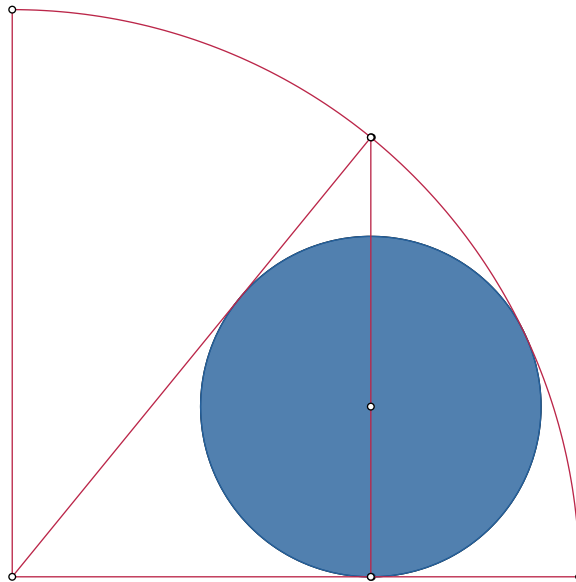
for (int i=1; i<n; ++i)
for (int j=0; j<VX[i+1].length; ++j) {
    path pa=VX[i-1][j+1]--VX[i][j]--VX[i+1][j]--VX[i][j+1]--cycle;
    int k=i;
    if (direction==false) k=j;
    filldraw(pa, hsv(0.8*k*n,1,0.8), white+1.2bp);
}

draw_poly_rhombus((0,0), r=1, n=24, direction=true);
draw_poly_rhombus((2.5,0), r=1, n=24, direction=false);

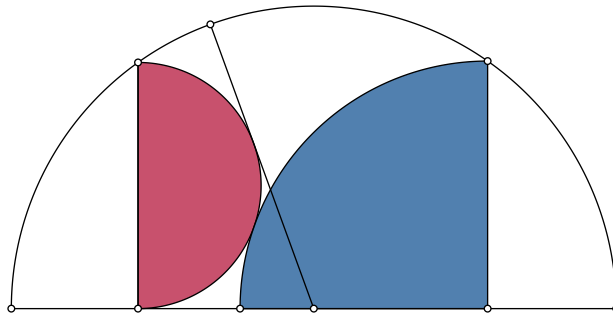
```

2 Problem

P1. Vẽ hình sau:



P2. Vẽ hình sau:



3 Tài liệu tham khảo

- [1]. Tài liệu Asymptote: the Vector Graphics Language
- [2]. Các ví dụ của Asymptote

- [3]. Tài liệu Asymptote Démarrage rapide
- [4]. Gói geometry.asy và olympiad.asy
- [5]. Tài liệu hướng dẫn gói **geometry.asy**: Euclidean geometry with asymptote
- [6]. Tài liệu hướng dẫn Asymptote 3D: Asymptote 3D, Bruno M. Colombel
- [7]. Tài liệu hướng dẫn ASY (tiếng Việt): ASY for Beginners, Trần Quân