

**Ministerul Educației și Cercetării al Republicii Moldova**

**Universitatea Tehnică a Moldovei**

**Departamentul Ingineria Software și Automatica**



# Raport

Lucrarea de laborator nr. 1

**Grafică pe Calculator**

Varianta 3

**A efectuat:**

Student grupa TI-231 FR

Apareci Aurica

**A verificat:**

asistent universitar

Ursu Adriana

**Chișinău**

**2024**

## Cuprins

1.	<b>Cadrul teoretic .....</b>	<b>3</b>
2.	<b>Rezumat succint la temă.....</b>	<b>4</b>
3.	<b>Listingul programului .....</b>	<b>5</b>
4.	<b>Testarea aplicației.....</b>	<b>6</b>
5.	<b>Concluzii .....</b>	<b>8</b>

# 1. Cadrul teoretic

**Tema:** Studierea primitivelor grafice simple 2D

**Scopul lucrării:** Obținerea cunoștințelor practice în sinteza scenelor grafice 2D statice, utilizând primitivele grafice simple a bibliotecii **p5.js**

**Sarcina (conform variantei):** Elaborați un program pentru sinteza unei scene 2D statice utilizând cel puțin 6 primitive grafice de diferite cum ar fi - arc(), ellipse(), circle(), line(), point(), quad(), rect(), square(), triangle(), primitivele trebuie să fie cu diferite atribute, lucrarea trebuie semnată (numele prenumele grupa) în colțul dreapta jos a ecranului. Elaborați un program care creează personajul conform variantei indicate de profesor.



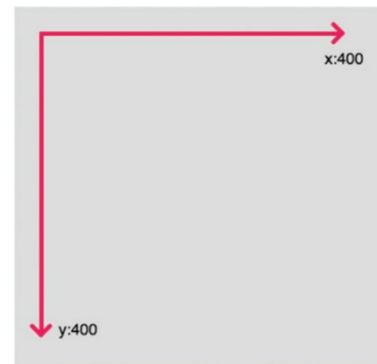
*Laborator nr. 1 Varianta 3*

## 2. Rezumat succint la temă

**P5.js** este o bibliotecă scrisă în limbajul de programare **JavaScript**, utilizată pentru crearea și vizualizarea imaginilor interactive cu ajutorul primitivelor grafice simple. P5.js permite crearea graficii pe calculator folosind un limbaj de programare. Aceasta permite integrarea simplă a codurilor scrise în pagini web prin adăugarea codului scris într-un document **HTML**.

**P5.js** este gratis, open-sources și independent de platformă, deci aplicațiile pot fi rulate pe orice sistem de operare, de asemenea p5.js are o familie mare de limbaje și medii programare înrudite.

Înainte de a începe crearea programelor proprii trebuie să știm că orice figură creată în mediul p5.js este legată de sistemul de coordonate, originea sistemului de coordonate în orice program este colțul din stânga sus al ecranului. Axa verticală se numește **axa Y**, iar cea orizontală **axa X**. Creșterea valorilor pentru coordonatele  $x$  și  $y$  sunt prezentate mai jos.



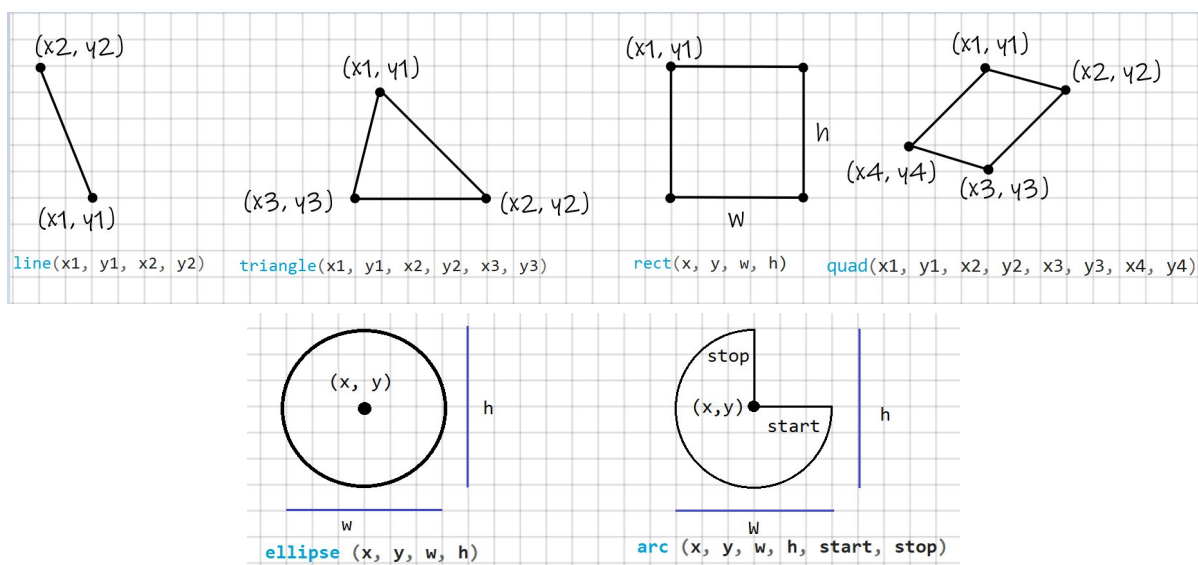
**Funcția setup()** – se apelează o singură dată la începutul programului. Este folosită pentru setarea proprietăților inițiale a mediului de lucru, cum ar fi dimensiunea și culoarea ecranului, precum și pentru încărcarea fișiere multimedia la lansarea programului, cum ar fi imagini și fonturi. Poate exista o singură funcție setup() per program și nu ar trebui apelată după execuția inițială.

**Notă:** Variabilele declarate în setup() nu sunt disponibile în alte funcții, inclusiv draw ().

**Funcția draw ()** se apelează imediat după setup(), execută în continuu rândurile de cod care sunt incluse în corpul său, până la săvârșitul programului sau până la apelarea noLoop().

**Notă:** dacă în setup () este apelată funcția noLoop(), funcția draw() se va executa o singură dată.

**Primitivele grafice simple** reprezintă figurile geometrice care pot fi create cu ajutorul funcțiilor din biblioteca grafică P5.js. Cele mai simple primitive grafice sunt primitivele grafice 2D, în figura 1.1 arătată corespunderea punctelor figurilor geometrice și parametrilor care trebuie indicate ca argumentele funcțiilor în codul programului.



### 3. Listingul programului

```
function setup() {
  createCanvas(400, 400);
  background(233);

  let brown = color(125, 0, 0);
  let yellow = color(255, 190, 0);
  let black = color(0);

  drawFlower(200, 210, 100, brown);

  fill(brown);
  ellipse(200, 210, 220, 220);

  // Urechi mari
  fill(brown);
  stroke(black)
  ellipse(170, 170, 50, 50);
  ellipse(230, 170, 50, 50);

  // Urechi mici
  fill(200, 100, 100);
  ellipse(170, 170, 25, 25);
  ellipse(230, 170, 25, 25);

  // Fata
  fill(yellow);
  stroke(black);
  strokeWeight(3);
  triangle(125, 290, 200, 120, 275, 290);

  // Ochi
  fill(black);
  ellipse(185, 210, 25, 25);
  ellipse(215, 210, 25, 25);

  // Reflexii ochi
  fill(255);
  ellipse(180, 205, 8, 8);
  ellipse(210, 205, 8, 8);

  // Nas
  fill(black);
  triangle(195, 240, 205, 240, 200, 250);

  // Gura
  line(200, 250, 200, 260);
  line(200, 260, 190, 265);
  line(200, 260, 210, 265);

  // Mustăți
  line(190, 260, 170, 255);
  line(190, 265, 170, 265);
  line(190, 270, 170, 275);
```

```

    line(210, 260, 230, 255);
    line(210, 265, 230, 265);
    line(210, 270, 230, 275);
  }

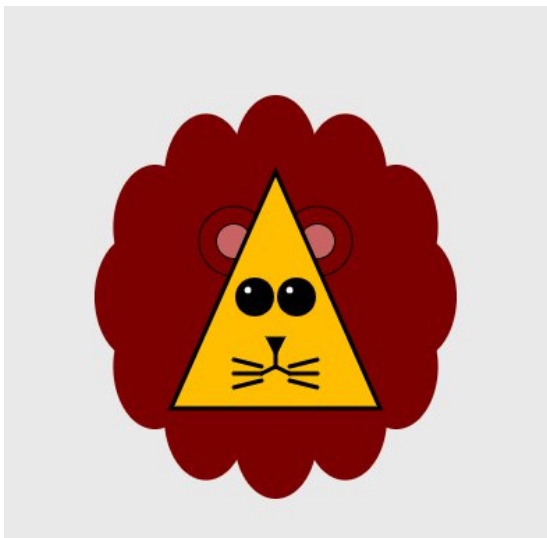
function drawFlower(x, y, radius, petalColor) {
  let numPetals = 12;
  let angleStep = TWO_PI / numPetals;

  fill(petalColor);
  noStroke();

  for (let i = 0; i < numPetals; i++) {
    let angle = i * angleStep;
    let petalX = x + cos(angle) * radius;
    let petalY = y + sin(angle) * radius;
    ellipse(petalX, petalY, 60, 90);
  }
}

```

#### 4. Testarea aplicației



## 5. Concluzii

JavaScript (JS) este un limbaj de scripting, folosit în principal pe Web. Este folosit pentru a îmbunătăți paginile HTML și este de obicei găsit încorporat în codul HTML. JavaScript este un limbaj interpretat. Astfel, nu trebuie să fie compilat. JavaScript redă paginile web într-un mod interactiv și dinamic. Acest lucru permite paginilor să reacționeze la evenimente, să prezinte efecte speciale, să accepte text variabil, să valideze date, etc.

**P5.js** este o bibliotecă JavaScript pentru un cod creativ. O colecție de metode predefinite, ce ne oferă instrumente care simplifică procesul de creare a imaginilor interactive prin intermediul codului, în browserul web.

La elaborarea am folosit drept sursa de informație suportul de curs, cat si documentația publică a bibliotecii P5.js. Din motivul ca dețin deja o anumita experiență de lucru în JS, nu am întâlnit dificultăți la elaborarea acestei lucrări. Desenul atribuit variantei primite a fost realizat cu ajutorul primitivelor grafice.