```
PImage sfondo;
PImage back;
PImage mirino;
PImage mirinoreverse;
PImage mappa;

int bgX;
int bgY;
int mirinoX;
int mirinoY;
int gap;

boolean upMove;
boolean downMove;
boolean leftMove;
boolean rightMove;
int[] arrayBack;
int indexBack;

int[][] area;
int pagina;
int colore = 0;

boolean viewMap;



void setup() {
  size(240,320);
```

```
pagina = 0;

gap = 5;
area = new int[9][2]; //[x][y]
arrayBack = new int[10];
indexBack = 0;
upMove = false;
downMove = false;
leftMove = false;
rightMove = false;

area[0][0] = -240;
area[0][1] = -320;
area[1][0] = -480;
area[1][1] = 0;
area[2][0] = -240;
area[2][1] = -640;
area[3][0] = 0;
area[3][1] = -320;
area[4][0] = 0;
area[4][1] = -640;
area[5][0] = 0;
area[5][1] = 0;
area[6][0] = -240;
area[6][1] = 0;
area[7][0] = -480;
area[7][1] = -320;
area[8][0] = -480;
area[8][1] = -640;
viewMap = false;

bgX = area[0][0];
bgY = area[0][1];
```

```
  sfondo = loadImage("image0K.jpg");
  back = loadImage("back.png");
  mirino = loadImage("00mirino.png");
  mirinoreverse = loadImage("01mirino.png");
  mappa = loadImage("mappabase.jpg");
}

void draw() {

  if(pagina==0) {
    image (sfondo, area[0][0], area[0][1]);
  }

  if(pagina > 0) {
    bgX = easeOut(bgX, area[pagina][0], 6);
    bgY = easeOut(bgY, area[pagina][1], 6);
    image (sfondo, bgX, bgY);
  }

  // show Map
  if(viewMap) {
    image(mappa, -240, -320);
  }

  if((pagina > 1 && pagina < 6) || (pagina > 6 && pagina == 8)) {
    image(back,190,253);
  }

  if(pagina > 0) {
    drawMirino();
```

```
  }
}

void addBack(int a) {
  arrayBack[indexBack] = a;
  indexBack++;
}

void resetBack() {
  for(int i=0; i< arrayBack.length; i++) {
    arrayBack[i] = 0;
  }
  indexBack = 0;
}



void mousePressed() {

  if(pagina > 2 && viewMap == false) {
    if(mirinoX > 190 && mirinoX <  240 && mirinoY > 253 && mirinoY < 320) {
      pagina = arrayBack[indexBack-2];
      indexBack--;
    }
  }

  if((pagina == 2 && viewMap==true) || (pagina == 6 && viewMap==true)) {
    if(mirinoX > 190 && mirinoX <  240 && mirinoY > 253 && mirinoY < 320) {
      viewMap = false;
    }
  }

  if(pagina == 0) {
```

```
  PAGINA = 1;
}

ELSE IF(PAGINA == 1) {

  IF(MIRINOX > 4 && MIRINOX < 234 && MIRINOY > 244 && MIRINOY < 311) {
    PAGINA = 5;
  }
  ELSE IF(MIRINOX > 85 && MIRINOX < 168 && MIRINOY > 130 && MIRINOY < 225) {
    PAGINA = 2;
    ADDBACK(2);
  }
}
ELSE IF(PAGINA == 2) {

  IF(MIRINOX > 40 && MIRINOX < 234 && MIRINOY > 180 && MIRINOY < 247) {
    PAGINA = 3;
    ADDBACK(3);
  }
  ELSE IF(MIRINOX > 40 && MIRINOX < 234 && MIRINOY > 93 && MIRINOY < 160) {
    PAGINA = 8;
    ADDBACK(8);
  }
  ELSE IF(MIRINOX > 40 && MIRINOX < 234 && MIRINOY > 4 && MIRINOY < 71) {
    VIEWMAP = TRUE;
  }
}

ELSE IF(PAGINA == 3) {

  IF(MIRINOX > 20 && MIRINOX <  180 && MIRINOY > 220 && MIRINOY < 255) {
    PAGINA = 4;
    ADDBACK(4);
```

```
  }
}

ELSE IF(PAGINA == 4) {

  IF(MIRINOX > 85 && MIRINOX < 168 && MIRINOY > 244 && MIRINOY < 311 ) {
    PAGINA = 2;
    RESETBACK();
    ADDBACK(2);
  }
}

ELSE IF(PAGINA == 5) {

  IF(MIRINOX > 85 && MIRINOX < 168 && MIRINOY > 220 && MIRINOY < 311 ) {
    PAGINA = 6;
    ADDBACK(6);
  }
}

ELSE IF(PAGINA == 6) {

  IF(MIRINOX > 40 && MIRINOX < 234 && MIRINOY > 4 && MIRINOY < 71) {
    VIEWMAP = TRUE;
  }
  IF(MIRINOX > 40 && MIRINOX <  234 && MIRINOY > 93 && MIRINOY < 160) {
    PAGINA = 7;
    ADDBACK(7);
  }
}

ELSE IF(PAGINA == 7) {
```

```
    PAGINA = 2;
    RESETBACK();
    ADDBACK(2);
  }
}


INT MIRINOXTEMP;
INT MIRINOYTEMP;


VOID DRAWMIRINO(){

  MIRINOXTEMP = MIRINOX -40;
  MIRINOYTEMP = MIRINOY -46;

  MIRINOX = MOUSEX;
  MIRINOY = MOUSEY;


  IF(PAGINA == 1){

    IF(MIRINOX > 4 && MIRINOX < 234 && MIRINOY > 244 && MIRINOY < 311){
      IMAGE(MIRINOREVERSE, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP,
-40, 270));
    }
    ELSE IF(MIRINOX > 85 && MIRINOX < 168 && MIRINOY > 130 && MIRINOY < 225){
      IMAGE(MIRINOREVERSE, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP,
-40, 270));
    }
    ELSE{
```

```
      IMAGE(MIRINO, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP, -40,
270));

    }



  }



  IF(PAGINA == 2){



    IF(MIRINOX > 40 && MIRINOX < 234 && MIRINOY > 180 && MIRINOY < 247){

      IMAGE(MIRINOREVERSE, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP,
-40, 270));

    }

    ELSE IF(MIRINOX > 40 && MIRINOX < 234 && MIRINOY > 93 && MIRINOY < 160){

      IMAGE(MIRINOREVERSE, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP,
-40, 270));

    }

    ELSE IF(MIRINOX > 40 && MIRINOX < 234 && MIRINOY > 4 && MIRINOY < 71){

      IMAGE(MIRINOREVERSE, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP,
-40, 270));

    }

    ELSE{

      IMAGE(MIRINO, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP, -40,
270));

    }

  }

  IF(PAGINA == 3){




    IF(MIRINOX > 20 && MIRINOX < 60 && MIRINOY > 50 && MIRINOY < 70 ){

      IMAGE(MIRINOREVERSE, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP,
-40, 270));

    }



    ELSE IF(MIRINOX > 67 && MIRINOX < 103 && MIRINOY > 50 && MIRINOY < 79){

      IMAGE(MIRINOREVERSE, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP,
-40, 270));
```

```
    }


    ELSE IF(MIRINOX > 120 && MIRINOX < 155 && MIRINOY > 46 && MIRINOY < 70){

      IMAGE(MIRINOREVERSE, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP,
-40, 270));

    }


    ELSE IF(MIRINOX > 165 && MIRINOX <  180 && MIRINOY > 46 && MIRINOY < 79){

      IMAGE(MIRINOREVERSE, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP,
-40, 270));

    }


    ELSE IF(MIRINOX > 200 && MIRINOX <  235 && MIRINOY > 46 && MIRINOY < 70){

      IMAGE(MIRINOREVERSE, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP,
-40, 270));

    }


    ELSE IF(MIRINOX > 20 && MIRINOX <  180 && MIRINOY > 220 && MIRINOY < 255){

      IMAGE(MIRINOREVERSE, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP,
-40, 270));

    }

    ELSE{

      IMAGE(MIRINO, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP, -40,
270));

    }

  }

  IF(PAGINA == 4){


    IF(MIRINOX > 85 && MIRINOX < 168 && MIRINOY > 244 && MIRINOY < 311 ){

      IMAGE(MIRINOREVERSE, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP,
-40, 270));

    }

    ELSE{

      IMAGE(MIRINO, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP, -40,
270));

    }
```

```
  }

  IF(PAGINA == 5){


    IF(MIRINOX > 85 && MIRINOX < 168 && MIRINOY > 220 && MIRINOY < 311 ){

      IMAGE(MIRINOREVERSE, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP,
-40, 270));

    }

    ELSE{

      IMAGE(MIRINO, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP, -40,
270));

    }

  }

  IF(PAGINA == 6){


    IF(MIRINOX > 40 && MIRINOX < 234 && MIRINOY > 4 && MIRINOY < 71){

      IMAGE(MIRINOREVERSE, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP,
-40, 270));

    }



    ELSE IF(MIRINOX > 40 && MIRINOX <  234 && MIRINOY > 93 && MIRINOY < 160){

      IMAGE(MIRINOREVERSE, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP,
-40, 270));

    }

    ELSE{

      IMAGE(MIRINO, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP, -40,
270));

    }

  }

  IF(PAGINA == 8){

      IMAGE(MIRINO, CONSTRAIN(MIRINOXTEMP, -35, 195) ,CONSTRAIN(MIRINOYTEMP, -40,
270));

  }

}

INT EASEOUT(INT CURRENTORIGINAL, INT TARGETORIGINAL, INT SPEED)

{
```

```
  // MAKE VALUES LARGE FOR MATH

  INT CURRENT = CURRENTORIGINAL * 1000; // MAKE CURRENT VALUE LARGE FOR MATH (E.G. 2 BECOMES 2000)

  INT TARGET = TARGETORIGINAL * 1000;   // MAKE TARGET VALUE LARGE FOR MATH  (E.G. 33 BECOMES 33000)



  // DO MATH TO CALCULATE OUR NEXT VALUE

  INT CHANGE = TARGET - CURRENT;     // FIND OUT HOW MUCH CHANGE THERE IS (E.G. 33000 - 2000 = 31000)

  INT CHANGELITTLE = CHANGE / SPEED; // MAKE THE CHANGE A LITTLE CHANGE   (E.G. 31000 / 4 = 7750)

  INT NEXT = CURRENT + CHANGELITTLE; // CHANGE THE CURRENT VALUE A LITTLE (E.G. 2000 + 7750 = 9750)



  // MAKE NEXT VALUE SMALL FOR SCREEN

  NEXT = NEXT / 1000;



  // IF OUR LITTLE CHANGE WAS SO LITTLE THAT WE DIDN'T MOVE...

  IF(NEXT == CURRENTORIGINAL){

    NEXT = TARGETORIGINAL; // OUR NEXT STEP IS OUR TARGET

  }



  RETURN NEXT; // RETURN OUR NEXT VALUE (E.G. 9750 / 1000 = 9, REMEMBER THAT WE STARTED WITH 2)

}
```