#include<iostream>

#include<conio.h>

#include<dos.h>

#include<stdlib.h>

#include<windows.h>

#include<graphics.h>

void y\_2021();

void next();

int main()

{

int gd=DETECT,gm;

initgraph(&gd,&gm,"c:\\tc\\bgi");

y\_2021();

next();

getch();

closegraph();

}

void y\_2021()

{

int a=300,b=-150;

int c=300,d=-150;

float x=0.8,y=0.6;

float m=0.8,n=0.6;

for(int r=0;r<500;r++)

{

//middle cloud

ellipse(200,30,60,240,10,15);

ellipse(225,20,345,165,20,15);

ellipse(262,27,340,160,20,15);

ellipse(268,42,230,50,20,10);

ellipse(226,46,163,340,31,15);

ellipse(300,21,20,182,21,19);

ellipse(325,31,290,100,21,15);

ellipse(299,45,228,20,33,11);

//left cloud

ellipse(40,30,60,240,10,15);

ellipse(65,20,345,165,20,15);

ellipse(105,27,340,160,20,15);

ellipse(109,42,230,50,20,10);

ellipse(66,46,163,340,31,17);

//right cloud

ellipse(450,30,60,240,10,15);

ellipse(475,20,345,165,20,15);

ellipse(512,27,340,160,20,15);

ellipse(518,42,230,50,20,10);

ellipse(476,46,163,340,31,15);

//tree1

//tree1 crown

line(240,420,275,420);

line(240,420,245,405);

line(275,420,270,405);

line(245,405,240,405);

line(270,405,275,405);

line(240,405,245,390);

line(275,405,270,390);

line(245,390,275,390);

line(270,390,275,390);

line(240,390,258,370);

line(275,390,257,370);

//tree1 trunk

line(250,440,250,420);

line(265,440,265,420);

line(250,440,265,440);

line(250,420,265,420);

//tree2

//tree1 crown

line(240+c,420+d,275+c,420+d);

line(240+c,420+d,245+c,405+d);

line(275+c,420+d,270+c,405+d);

line(245+c,405+d,240+c,405+d);

line(270+c,405+d,275+c,405+d);

line(240+c,405+d,245+c,390+d);

line(275+c,405+d,270+c,390+d);

line(245+c,390+d,275+c,390+d);

line(270+c,390+d,275+c,390+d);

line(240+c,390+d,258+c,370+d);

line(275+c,390+d,257+c,370+d);

//tree2 trunk

line(250+c,440+d,250+c,420+d);

line(265+c,440+d,265+c,420+d);

line(250+c,440+d,265+c,440+d);

line(250+c,420+d,265+c,420+d);

//tree3

//tree3 crown

line(240\*m,420\*n,275\*m,420\*n);

line(240\*m,420\*n,245\*m,405\*n);

line(275\*m,420\*n,270\*m,405\*n);

line(245\*m,405\*n,240\*m,405\*n);

line(270\*m,405\*n,275\*m,405\*n);

line(240\*m,405\*n,245\*m,390\*n);

line(275\*m,405\*n,270\*m,390\*n);

line(245\*m,390\*n,275\*m,390\*n);

line(270\*m,390\*n,275\*m,390\*n);

line(240\*m,390\*n,258\*m,370\*n);

line(275\*m,390\*n,257\*m,370\*n);

//tree1 trunk

line(250\*m,440\*n,250\*m,420\*n);

line(265\*m,440\*n,265\*m,420\*n);

line(250\*m,440\*n,265\*m,440\*n);

line(250\*m,420\*n,265\*m,420\*n);

//1st house

//triangle of first house

line(88,400,163,400);

line(100,370,88,400);

line(100,370,112,400);

line(100,370,150,370);

line(150,370,163,400);

line(88,400,163,400);

line(100,370,112,400);

line(100,370,150,370);

line(150,370,163,400);

line(112,400,112,440);

line(88,440,112,440);

line(88,400,88,440);

line(112,440,163,440);

line(163,400,163,440);

line(112,440,163,440);

line(112,400,112,440);

line(88,400,163,400);

//firsthousedoor

line(94,440,94,420);

line(106,440,106,420);

line(94,420,106,420);

//2nd house

//triangle of secound house

line(88+a,400+b,163+a,400+b);

line(100+a,370+b,88+a,400+b);

line(100+a,370+b,112+a,400+b);

line(100+a,370+b,150+a,370+b);

line(150+a,370+b,163+a,400+b);

line(88+a,400+b,163+a,400+b);

line(100+a,370+b,112+a,400+b);

line(100+a,370+b,150+a,370+b);

line(150+a,370+b,163+a,400+b);

line(112+a,400+b,112+a,440+b);

line(88+a,440+b,112+a,440+b);

line(88+a,400+b,88+a,440+b);

line(112+a,440+b,163+a,440+b);

line(163+a,400+b,163+a,440+b);

line(112+a,440+b,163+a,440+b);

line(112+a,400+b,112+a,440+b);

line(88+a,400+b,163+a,400+b);

//door of secound house

line(94+a,440+b,94+a,420+b);

line(106+a,440+b,106+a,420+b);

line(94+a,420+b,106+a,420+b);

//3rd house

//triangle of third house

line(88\*x,400\*y,163\*x,400\*y);

line(100\*x,370\*y,88\*x,400\*y);

line(100\*x,370\*y,112\*x,400\*y);

line(100\*x,370\*y,150\*x,370\*y);

line(150\*x,370\*y,163\*x,400\*y);

line(88\*x,400\*y,163\*x,400\*y);

line(100\*x,370\*y,112\*x,400\*y);

line(100\*x,370\*y,150\*x,370\*y);

line(150\*x,370\*y,163\*x,400\*y);

line(112\*x,400\*y,112\*x,440\*y);

line(88\*x,440\*y,112\*x,440\*y);

line(88\*x,400\*y,88\*x,440\*y);

line(112\*x,440\*y,163\*x,440\*y);

line(163\*x,400\*y,163\*x,440\*y);

line(112\*x,440\*y,163\*x,440\*y);

line(112\*x,400\*y,112\*x,440\*y);

line(88\*x,400\*y,163\*x,400\*y);

//door of third house

line(94\*x,440\*y,94\*x,420\*y);

line(106\*x,440\*y,106\*x,420\*y);

line(94\*x,420\*y,106\*x,420\*y);

//hills

setcolor(GREEN);

for(int j=200;j<=202; j++)

line(0,j,1500,j);

setfillstyle(1,GREEN);

line(0,200,150,100);

line(150,100,250,200);

floodfill(150,170,GREEN);

line(225,200,375,100);

line(375,100,475,200);

floodfill(375,130,GREEN);

line(425,200,550,100);

line(550,100,650,200);

floodfill(550,130,GREEN);

//sun

setcolor(YELLOW);

circle(240,100,40);

setfillstyle(1,YELLOW);

floodfill(240,100,YELLOW);

circle(20,300,10);

setcolor(12);

settextstyle(10,0,4);

outtextxy(10,290,"2");

setcolor(15);

line(20,350,18,400); // left

line(20,350,22,400); //right

line(0,400,700,400);

line(10,320,0,350);

line(33,320,40,350);

circle(70,300,10);

settextstyle(10,0,4);

outtextxy(57,290,"0");

line(70,350,68,400); // left

line(70,350,72,400); //right

line(0,400,700,400);

line(57,320,47,350);

line(82,320,90,350);

circle(120,300,10);

settextstyle(10,0,4);

setcolor(14);

outtextxy(110,290,"2");

setcolor(15);

line(120,350,118,400); // left

line(120,350,122,400); //right

line(0,400,700,400);

line(110,320,100,350);

line(133,320,142,350);

circle(170+r,300,10);

settextstyle(10,0,4);

setcolor(13);

outtextxy(156+r,290,"1");

setcolor(15);

if(r%10==0)

{

line(170+r,350,158+r,400); // left

line(170+r,350,182+r,400); //right

delay(30);

}

else

{

line(170+r,350,168+r,400); // left

line(170+r,350,172+r,400); //right

delay(20);

}

line(0,400,700,400);

line(157+r,320,147+r,350);

line(180+r,320,188+r,350);

delay(20);

cleardevice();

}

for(int s=0;s<560;s++) //second

{

//middle cloud

ellipse(200,30,60,240,10,15);

ellipse(225,20,345,165,20,15);

ellipse(262,27,340,160,20,15);

ellipse(268,42,230,50,20,10);

ellipse(226,46,163,340,31,15);

ellipse(300,21,20,182,21,19);

ellipse(325,31,290,100,21,15);

ellipse(299,45,228,20,33,11);

//left cloud

ellipse(40,30,60,240,10,15);

ellipse(65,20,345,165,20,15);

ellipse(105,27,340,160,20,15);

ellipse(109,42,230,50,20,10);

ellipse(66,46,163,340,31,17);

//right cloud

ellipse(450,30,60,240,10,15);

ellipse(475,20,345,165,20,15);

ellipse(512,27,340,160,20,15);

ellipse(518,42,230,50,20,10);

ellipse(476,46,163,340,31,15);

//tree1

//tree1 crown

line(240,420,275,420);

line(240,420,245,405);

line(275,420,270,405);

line(245,405,240,405);

line(270,405,275,405);

line(240,405,245,390);

line(275,405,270,390);

line(245,390,275,390);

line(270,390,275,390);

line(240,390,258,370);

line(275,390,257,370);

//tree1 trunk

line(250,440,250,420);

line(265,440,265,420);

line(250,440,265,440);

line(250,420,265,420);

//tree2

//tree1 crown

line(240+c,420+d,275+c,420+d);

line(240+c,420+d,245+c,405+d);

line(275+c,420+d,270+c,405+d);

line(245+c,405+d,240+c,405+d);

line(270+c,405+d,275+c,405+d);

line(240+c,405+d,245+c,390+d);

line(275+c,405+d,270+c,390+d);

line(245+c,390+d,275+c,390+d);

line(270+c,390+d,275+c,390+d);

line(240+c,390+d,258+c,370+d);

line(275+c,390+d,257+c,370+d);

//tree2 trunk

line(250+c,440+d,250+c,420+d);

line(265+c,440+d,265+c,420+d);

line(250+c,440+d,265+c,440+d);

line(250+c,420+d,265+c,420+d);

//tree3

//tree3 crown

line(240\*m,420\*n,275\*m,420\*n);

line(240\*m,420\*n,245\*m,405\*n);

line(275\*m,420\*n,270\*m,405\*n);

line(245\*m,405\*n,240\*m,405\*n);

line(270\*m,405\*n,275\*m,405\*n);

line(240\*m,405\*n,245\*m,390\*n);

line(275\*m,405\*n,270\*m,390\*n);

line(245\*m,390\*n,275\*m,390\*n);

line(270\*m,390\*n,275\*m,390\*n);

line(240\*m,390\*n,258\*m,370\*n);

line(275\*m,390\*n,257\*m,370\*n);

//tree1 trunk

line(250\*m,440\*n,250\*m,420\*n);

line(265\*m,440\*n,265\*m,420\*n);

line(250\*m,440\*n,265\*m,440\*n);

line(250\*m,420\*n,265\*m,420\*n);

//1st house

//triangle of first house

line(88,400,163,400);

line(100,370,88,400);

line(100,370,112,400);

line(100,370,150,370);

line(150,370,163,400);

line(88,400,163,400);

line(100,370,112,400);

line(100,370,150,370);

line(150,370,163,400);

line(112,400,112,440);

line(88,440,112,440);

line(88,400,88,440);

line(112,440,163,440);

line(163,400,163,440);

line(112,440,163,440);

line(112,400,112,440);

line(88,400,163,400);

//firsthousedoor

line(94,440,94,420);

line(106,440,106,420);

line(94,420,106,420);

//2nd house

//triangle of secound house

line(88+a,400+b,163+a,400+b);

line(100+a,370+b,88+a,400+b);

line(100+a,370+b,112+a,400+b);

line(100+a,370+b,150+a,370+b);

line(150+a,370+b,163+a,400+b);

line(88+a,400+b,163+a,400+b);

line(100+a,370+b,112+a,400+b);

line(100+a,370+b,150+a,370+b);

line(150+a,370+b,163+a,400+b);

line(112+a,400+b,112+a,440+b);

line(88+a,440+b,112+a,440+b);

line(88+a,400+b,88+a,440+b);

line(112+a,440+b,163+a,440+b);

line(163+a,400+b,163+a,440+b);

line(112+a,440+b,163+a,440+b);

line(112+a,400+b,112+a,440+b);

line(88+a,400+b,163+a,400+b);

//door of secound house

line(94+a,440+b,94+a,420+b);

line(106+a,440+b,106+a,420+b);

line(94+a,420+b,106+a,420+b);

//3rd house

//triangle of third house

line(88\*x,400\*y,163\*x,400\*y);

line(100\*x,370\*y,88\*x,400\*y);

line(100\*x,370\*y,112\*x,400\*y);

line(100\*x,370\*y,150\*x,370\*y);

line(150\*x,370\*y,163\*x,400\*y);

line(88\*x,400\*y,163\*x,400\*y);

line(100\*x,370\*y,112\*x,400\*y);

line(100\*x,370\*y,150\*x,370\*y);

line(150\*x,370\*y,163\*x,400\*y);

line(112\*x,400\*y,112\*x,440\*y);

line(88\*x,440\*y,112\*x,440\*y);

line(88\*x,400\*y,88\*x,440\*y);

line(112\*x,440\*y,163\*x,440\*y);

line(163\*x,400\*y,163\*x,440\*y);

line(112\*x,440\*y,163\*x,440\*y);

line(112\*x,400\*y,112\*x,440\*y);

line(88\*x,400\*y,163\*x,400\*y);

//door of third house

line(94\*x,440\*y,94\*x,420\*y);

line(106\*x,440\*y,106\*x,420\*y);

line(94\*x,420\*y,106\*x,420\*y);

//hills

setcolor(GREEN);

for(int j=200;j<=202; j++)

line(0,j,1500,j);

setfillstyle(1,GREEN);

line(0,200,150,100);

line(150,100,250,200);

floodfill(150,170,GREEN);

line(225,200,375,100);

line(375,100,475,200);

floodfill(375,130,GREEN);

line(425,200,550,100);

line(550,100,650,200);

floodfill(550,130,GREEN);

//sun

setcolor(YELLOW);

circle(240,100,40);

setfillstyle(1,YELLOW);

floodfill(240,100,YELLOW);

//middle cloud

ellipse(200,30,60,240,10,15);

ellipse(225,20,345,165,20,15);

ellipse(262,27,340,160,20,15);

ellipse(268,42,230,50,20,10);

ellipse(226,46,163,340,31,15);

ellipse(300,21,20,182,21,19);

ellipse(325,31,290,100,21,15);

ellipse(299,45,228,20,33,11);

//left cloud

ellipse(40,30,60,240,10,15);

ellipse(65,20,345,165,20,15);

ellipse(105,27,340,160,20,15);

ellipse(109,42,230,50,20,10);

ellipse(66,46,163,340,31,17);

//right cloud

ellipse(450,30,60,240,10,15);

ellipse(475,20,345,165,20,15);

ellipse(512,27,340,160,20,15);

ellipse(518,42,230,50,20,10);

ellipse(476,46,163,340,31,15);

line(240,420,275,420);

line(240,420,245,405);

line(275,420,270,405);

line(245,405,240,405);

line(270,405,275,405);

line(240,405,245,390);

line(275,405,270,390);

line(245,390,275,390);

line(270,390,275,390);

line(240,390,258,370);

line(275,390,257,370);

//tree1 trunk

line(250,440,250,420);

line(265,440,265,420);

line(250,440,265,440);

line(250,420,265,420);

//tree1 crown

line(240+c,420+d,275+c,420+d);

line(240+c,420+d,245+c,405+d);

line(275+c,420+d,270+c,405+d);

line(245+c,405+d,240+c,405+d);

line(270+c,405+d,275+c,405+d);

line(240+c,405+d,245+c,390+d);

line(275+c,405+d,270+c,390+d);

line(245+c,390+d,275+c,390+d);

line(270+c,390+d,275+c,390+d);

line(240+c,390+d,258+c,370+d);

line(275+c,390+d,257+c,370+d);

//tree1 trunk

line(250+c,440+d,250+c,420+d);

line(265+c,440+d,265+c,420+d);

line(250+c,440+d,265+c,440+d);

line(250+c,420+d,265+c,420+d);

line(240+c,420+d,275+c,420+d);

line(240+c,420+d,245+c,405+d);

line(275+c,420+d,270+c,405+d);

line(245+c,405+d,240+c,405+d);

line(270+c,405+d,275+c,405+d);

line(240+c,405+d,245+c,390+d);

line(275+c,405+d,270+c,390+d);

line(245+c,390+d,275+c,390+d);

line(270+c,390+d,275+c,390+d);

line(240+c,390+d,258+c,370+d);

line(275+c,390+d,257+c,370+d);

//tree1 trunk

line(250+c,440+d,250+c,420+d);

line(265+c,440+d,265+c,420+d);

line(250+c,440+d,265+c,440+d);

line(250+c,420+d,265+c,420+d);

line(240\*m,420\*n,275\*m,420\*n);

line(240\*m,420\*n,245\*m,405\*n);

line(275\*m,420\*n,270\*m,405\*n);

line(245\*m,405\*n,240\*m,405\*n);

line(270\*m,405\*n,275\*m,405\*n);

line(240\*m,405\*n,245\*m,390\*n);

line(275\*m,405\*n,270\*m,390\*n);

line(245\*m,390\*n,275\*m,390\*n);

line(270\*m,390\*n,275\*m,390\*n);

line(240\*m,390\*n,258\*m,370\*n);

line(275\*m,390\*n,257\*m,370\*n);

//tree1 trunk

line(250\*m,440\*n,250\*m,420\*n);

line(265\*m,440\*n,265\*m,420\*n);

line(250\*m,440\*n,265\*m,440\*n);

line(250\*m,420\*n,265\*m,420\*n);

line(88,400,163,400);

line(100,370,88,400);

line(100,370,112,400);

line(100,370,150,370);

line(150,370,163,400);

line(88,400,163,400);

line(100,370,112,400);

line(100,370,150,370);

line(150,370,163,400);

line(112,400,112,440);

line(88,440,112,440);

line(88,400,88,440);

line(112,440,163,440);

line(163,400,163,440);

line(112,440,163,440);

line(112,400,112,440);

line(88,400,163,400);

line(94,440,94,420);

line(106,440,106,420);

line(94,420,106,420);

line(88+a,400+b,163+a,400+b);

line(100+a,370+b,88+a,400+b);

line(100+a,370+b,112+a,400+b);

line(100+a,370+b,150+a,370+b);

line(150+a,370+b,163+a,400+b);

line(88+a,400+b,163+a,400+b);

line(100+a,370+b,112+a,400+b);

line(100+a,370+b,150+a,370+b);

line(150+a,370+b,163+a,400+b);

line(112+a,400+b,112+a,440+b);

line(88+a,440+b,112+a,440+b);

line(88+a,400+b,88+a,440+b);

line(112+a,440+b,163+a,440+b);

line(163+a,400+b,163+a,440+b);

line(112+a,440+b,163+a,440+b);

line(112+a,400+b,112+a,440+b);

line(88+a,400+b,163+a,400+b);

line(94+a,440+b,94+a,420+b);

line(106+a,440+b,106+a,420+b);

line(94+a,420+b,106+a,420+b);

line(88\*x,400\*y,163\*x,400\*y);

line(100\*x,370\*y,88\*x,400\*y);

line(100\*x,370\*y,112\*x,400\*y);

line(100\*x,370\*y,150\*x,370\*y);

line(150\*x,370\*y,163\*x,400\*y);

line(88\*x,400\*y,163\*x,400\*y);

line(100\*x,370\*y,112\*x,400\*y);

line(100\*x,370\*y,150\*x,370\*y);

line(150\*x,370\*y,163\*x,400\*y);

line(112\*x,400\*y,112\*x,440\*y);

line(88\*x,440\*y,112\*x,440\*y);

line(88\*x,400\*y,88\*x,440\*y);

line(112\*x,440\*y,163\*x,440\*y);

line(163\*x,400\*y,163\*x,440\*y);

line(112\*x,440\*y,163\*x,440\*y);

line(112\*x,400\*y,112\*x,440\*y);

line(88\*x,400\*y,163\*x,400\*y);

line(94\*x,440\*y,94\*x,420\*y);

line(106\*x,440\*y,106\*x,420\*y);

line(94\*x,420\*y,106\*x,420\*y);

setcolor(GREEN);

for(int j=200;j<=202; j++)

line(0,j,1500,j);

setfillstyle(1,GREEN);

line(0,200,150,100);

line(150,100,250,200);

floodfill(150,170,GREEN);

line(225,200,375,100);

line(375,100,475,200);

floodfill(375,130,GREEN);

line(425,200,550,100);

line(550,100,650,200);

floodfill(550,130,GREEN);

setcolor(GREEN);

for(int j=200;j<=202;j++)line(0,j,1500,j);

setfillstyle(1,GREEN);

line(0,200,150,100);

line(150,100,250,200);

floodfill(150,170,GREEN);

line(225,200,375,100);

line(375,100,475,200);

floodfill(375,130,GREEN);

line(425,200,550,100);

line(550,100,650,200);

floodfill(550,130,GREEN);

setcolor(YELLOW);

circle(240,100,40);

setfillstyle(1,YELLOW);

floodfill(240,100,YELLOW);

settextstyle(10,0,4);

setcolor(12);

outtextxy(10,290,"2");

setcolor(15);

line(20,350,18,400); // left

line(20,350,22,400); //right

line(0,400,700,400);

line(10,320,0,350);

line(33,320,40,350);

circle(70,300,10);

settextstyle(10,0,4);

outtextxy(57,290,"0");

line(70,350,68,400); // left

line(70,350,72,400); //right

line(0,400,700,400);

line(57,320,47,350);

line(82,320,90,350);

circle(120,300,10);

settextstyle(10,0,4);

setcolor(14);

outtextxy(110,290,"2");

setcolor(15);

line(120,350,118,400); // left

line(120,350,122,400); //right

line(0,400,700,400);

line(110,320,100,350);

line(133,320,142,350);

circle(720-s,300,10);

settextstyle(10,0,4);

setcolor(13);

outtextxy(710-s,290,"2");

setcolor(15);

if(s%10==0)

{

line(720-s,350,704-s,400); // left

line(720-s,350,740-s,400);//right

delay(20);

}

else

{

line(720-s,350,718-s,400);

line(720-s,350,722-s,400);

delay(20);

}

line(0,400,700,400);

line(714-s,320,700-s,350);

line(730-s,320,742-s,350);

settextstyle(7,0,3);

setcolor(10);

outtextxy(750-s,350,"HAPPY");

setcolor(15);

settextstyle(7,0,3);

setcolor(11);

outtextxy(880-s,350,"NEW");

setcolor(15);

settextstyle(7,0,3);

setcolor(12);

outtextxy(960-s,350," YEAR");

setcolor(15);

setcolor(BLUE);

line(740-s,382,1070-s,382);

setcolor(4);

circle(760-s,391,9);

setfillstyle(HATCH\_FILL,RED);

circle(840-s,391,9);

setfillstyle(HATCH\_FILL,RED);

circle(939-s,391,9);

setfillstyle(HATCH\_FILL,RED);

circle(1030-s,391,9);

setfillstyle(HATCH\_FILL,RED);

setcolor(15);

delay(35);

cleardevice();

}

circle(20,300,10);

settextstyle(10,0,4);

setcolor(12);

outtextxy(10,290,"2");

setcolor(15);

line(20,350,18,400); // left

line(20,350,22,400); //right

line(0,400,700,400);

line(10,320,0,350);

line(33,320,40,350);

circle(70,300,10);

settextstyle(10,0,4);

outtextxy(57,290,"0");

line(70,350,68,400); // left

line(70,350,72,400); //right

line(0,400,700,400);

line(57,320,47,350);

line(82,320,90,350);

circle(120,300,10);

settextstyle(10,0,4);

setcolor(14);

outtextxy(110,290,"2");

setcolor(15);

line(120,350,118,400); // left

line(120,350,122,400); //right

line(0,400,700,400);

line(110,320,100,350);

line(133,320,142,350);

}

void next()

{

int a=300,b=-150;

int c=300,d=-150;

float x=0.8,y=0.6;

float m=0.8,n=0.6;

for(int j=0;j<10;j++)

{

//middle cloud

ellipse(200,30,60,240,10,15);

ellipse(225,20,345,165,20,15);

ellipse(262,27,340,160,20,15);

ellipse(268,42,230,50,20,10);

ellipse(226,46,163,340,31,15);

ellipse(300,21,20,182,21,19);

ellipse(325,31,290,100,21,15);

ellipse(299,45,228,20,33,11);

//left cloud

ellipse(40,30,60,240,10,15);

ellipse(65,20,345,165,20,15);

ellipse(105,27,340,160,20,15);

ellipse(109,42,230,50,20,10);

ellipse(66,46,163,340,31,17);

//right cloud

ellipse(450,30,60,240,10,15);

ellipse(475,20,345,165,20,15);

ellipse(512,27,340,160,20,15);

ellipse(518,42,230,50,20,10);

ellipse(476,46,163,340,31,15);

//tree1

//tree1 crown

line(240,420,275,420);

line(240,420,245,405);

line(275,420,270,405);

line(245,405,240,405);

line(270,405,275,405);

line(240,405,245,390);

line(275,405,270,390);

line(245,390,275,390);

line(270,390,275,390);

line(240,390,258,370);

line(275,390,257,370);

//tree1 trunk

line(250,440,250,420);

line(265,440,265,420);

line(250,440,265,440);

line(250,420,265,420);

//tree2

//tree1 crown

line(240+c,420+d,275+c,420+d);

line(240+c,420+d,245+c,405+d);

line(275+c,420+d,270+c,405+d);

line(245+c,405+d,240+c,405+d);

line(270+c,405+d,275+c,405+d);

line(240+c,405+d,245+c,390+d);

line(275+c,405+d,270+c,390+d);

line(245+c,390+d,275+c,390+d);

line(270+c,390+d,275+c,390+d);

line(240+c,390+d,258+c,370+d);

line(275+c,390+d,257+c,370+d);

//tree2 trunk

line(250+c,440+d,250+c,420+d);

line(265+c,440+d,265+c,420+d);

line(250+c,440+d,265+c,440+d);

line(250+c,420+d,265+c,420+d);

//tree3

//tree3 crown

line(240\*m,420\*n,275\*m,420\*n);

line(240\*m,420\*n,245\*m,405\*n);

line(275\*m,420\*n,270\*m,405\*n);

line(245\*m,405\*n,240\*m,405\*n);

line(270\*m,405\*n,275\*m,405\*n);

line(240\*m,405\*n,245\*m,390\*n);

line(275\*m,405\*n,270\*m,390\*n);

line(245\*m,390\*n,275\*m,390\*n);

line(270\*m,390\*n,275\*m,390\*n);

line(240\*m,390\*n,258\*m,370\*n);

line(275\*m,390\*n,257\*m,370\*n);

//tree1 trunk

line(250\*m,440\*n,250\*m,420\*n);

line(265\*m,440\*n,265\*m,420\*n);

line(250\*m,440\*n,265\*m,440\*n);

line(250\*m,420\*n,265\*m,420\*n);

//1st house

//triangle of first house

line(88,400,163,400);

line(100,370,88,400);

line(100,370,112,400);

line(100,370,150,370);

line(150,370,163,400);

line(88,400,163,400);

line(100,370,112,400);

line(100,370,150,370);

line(150,370,163,400);

line(112,400,112,440);

line(88,440,112,440);

line(88,400,88,440);

line(112,440,163,440);

line(163,400,163,440);

line(112,440,163,440);

line(112,400,112,440);

line(88,400,163,400);

//firsthousedoor

line(94,440,94,420);

line(106,440,106,420);

line(94,420,106,420);

//2nd house

//triangle of secound house

line(88+a,400+b,163+a,400+b);

line(100+a,370+b,88+a,400+b);

line(100+a,370+b,112+a,400+b);

line(100+a,370+b,150+a,370+b);

line(150+a,370+b,163+a,400+b);

line(88+a,400+b,163+a,400+b);

line(100+a,370+b,112+a,400+b);

line(100+a,370+b,150+a,370+b);

line(150+a,370+b,163+a,400+b);

line(112+a,400+b,112+a,440+b);

line(88+a,440+b,112+a,440+b);

line(88+a,400+b,88+a,440+b);

line(112+a,440+b,163+a,440+b);

line(163+a,400+b,163+a,440+b);

line(112+a,440+b,163+a,440+b);

line(112+a,400+b,112+a,440+b);

line(88+a,400+b,163+a,400+b);

//door of secound house

line(94+a,440+b,94+a,420+b);

line(106+a,440+b,106+a,420+b);

line(94+a,420+b,106+a,420+b);

//3rd house

//triangle of third house

line(88\*x,400\*y,163\*x,400\*y);

line(100\*x,370\*y,88\*x,400\*y);

line(100\*x,370\*y,112\*x,400\*y);

line(100\*x,370\*y,150\*x,370\*y);

line(150\*x,370\*y,163\*x,400\*y);

line(88\*x,400\*y,163\*x,400\*y);

line(100\*x,370\*y,112\*x,400\*y);

line(100\*x,370\*y,150\*x,370\*y);

line(150\*x,370\*y,163\*x,400\*y);

line(112\*x,400\*y,112\*x,440\*y);

line(88\*x,440\*y,112\*x,440\*y);

line(88\*x,400\*y,88\*x,440\*y);

line(112\*x,440\*y,163\*x,440\*y);

line(163\*x,400\*y,163\*x,440\*y);

line(112\*x,440\*y,163\*x,440\*y);

line(112\*x,400\*y,112\*x,440\*y);

line(88\*x,400\*y,163\*x,400\*y);

//door of third house

line(94\*x,440\*y,94\*x,420\*y);

line(106\*x,440\*y,106\*x,420\*y);

line(94\*x,420\*y,106\*x,420\*y);

//hills

setcolor(GREEN);

for(int j=200;j<=202; j++)

line(0,j,1500,j);

setfillstyle(1,GREEN);

line(0,200,150,100);

line(150,100,250,200);

floodfill(150,170,GREEN);

line(225,200,375,100);

line(375,100,475,200);

floodfill(375,130,GREEN);

line(425,200,550,100);

line(550,100,650,200);

floodfill(550,130,GREEN);

//sun

setcolor(YELLOW);

circle(240,100,40);

setfillstyle(1,YELLOW);

floodfill(240,100,YELLOW);

setcolor(15);

circle(20,300,10);

settextstyle(10,0,4);

setcolor(12);

outtextxy(10,290,"2");

setcolor(15);

line(20,350,18,400); // left

line(20,350,22,400); //right

line(0,400,700,400);

line(10,320,0,350);

line(33,320,40,350);

circle(70,300,10);

settextstyle(10,0,4);

outtextxy(57,290,"0");

line(70,350,68,400); // left

line(70,350,72,400); //right

line(0,400,700,400);

line(57,320,47,350);

line(82,320,90,350);

circle(120,300,10);

settextstyle(10,0,4);

setcolor(14);

outtextxy(110,290,"2");

setcolor(15);

line(120,350,118,400); // left

line(120,350,122,400); //right

line(0,400,700,400);

line(110,320,100,350);

line(133,320,142,350);

circle(170,300,10);

settextstyle(10,0,4);

setcolor(13);

outtextxy(156,290,"2");

setcolor(15);

line(170,350,168,400); // left

line(170,350,172,400);//right

line(0,400,700,400);

line(159,320,147,350);

line(175,317,188,350);

settextstyle(7,0,4);

setcolor(10);

outtextxy(200,350,"3rd");

setcolor(15);

settextstyle(7,0,4);

setcolor(11);

outtextxy(320,350,"wave");

setcolor(15);

settextstyle(7,0,4);

setcolor(12);

outtextxy(400,350,"STRIKES");

setcolor(15);

setcolor(RED);

line(200,382,500,382);

setcolor(8);

circle(230,391,9);

circle(300,391,9);

circle(370,391,9);

circle(450,391,9);

if(j%2==0)

{

for(int i=0;i<60;i++)

{

setcolor(15);

circle(20,300,10);

settextstyle(10,0,4);

setcolor(12);

outtextxy(10,290,"2");

setcolor(15);

line(20,350,18,400); // left

line(20,350,22,400);//right

line(0,400,700,400);

line(10,320,0,350);

line(33,320,40,350);

circle(70,300,10);

settextstyle(10,0,4);

outtextxy(57,290,"0");

line(70,350,68,400); // left

line(70,350,72,400); //right

line(0,400,700,400);

line(57,320,47,350);

line(82,320,90,350);

circle(120,300,10);

settextstyle(10,0,4);

setcolor(14);

outtextxy(110,290,"2");

setcolor(15);

line(120,350,118,400); // left

line(120,350,122,400); //right

line(0,400,700,400);

line(110,320,100,350);

line(133,320,142,350);

circle(170,300,10);

settextstyle(10,0,4);

setcolor(13);

outtextxy(156,290,"2");

setcolor(15);

line(170,350,168,400); // left

line(170,350,172,400); //right

line(0,400,700,400);

line(159,320,147,350);

line(175,317,188,350);

settextstyle(7,0,4);

setcolor(10);

outtextxy(200,350,"3rd");

setcolor(15);

settextstyle(7,0,4);

setcolor(11);

outtextxy(320,350,"wave");

setcolor(15);

settextstyle(7,0,4);

setcolor(12);

outtextxy(400,350,"STRIKES");

setcolor(15);

setcolor(BLUE);

line(200,382,500,382);

setcolor(8);

circle(230,391,9);

circle(300,391,9);

circle(370,391,9);

circle(450,391,9);

setcolor(2);

outtextxy(200-i,70-i,"'");

outtextxy(204,70-i,"'");

outtextxy(208+i,70+i,"'");

outtextxy(210-i,74+i,"'");

outtextxy(210-i,78-i,"'");

outtextxy(210+i,82+i,"'");

outtextxy(199+i,74-i,"'");

outtextxy(203+i,74-i,"'");

outtextxy(206-i,74-i,"'");

outtextxy(193-i,70-i,"'");

outtextxy(195,70-i,"'");

outtextxy(188+i,70+i,"'");

outtextxy(184-i,44+i,"'");

outtextxy(222-i,88-i,"'");

outtextxy(226+i,62+i,"'");

outtextxy(229+i,54-i,"'");

outtextxy(201+i,64-i,"'");

outtextxy(205-i,70-i,"'");

outtextxy(199+i+2,78-i,"'");

outtextxy(203-i,78-i+2,"'");

outtextxy(206+i+1,78-i,"'");

outtextxy(199-i,82+i,"'");

outtextxy(203+i,82+6-i,"'");

outtextxy(206+i,82-i,"'");

setcolor(13);

outtextxy(300-i,70-i,"'");

outtextxy(304,70-i,"'");

outtextxy(308+i,70+i,"'");

outtextxy(310-i,74+i,"'");

outtextxy(310-i,78-i,"'");

outtextxy(310+i,82+i,"'");

outtextxy(299+i,74-i,"'");

outtextxy(303+i,74-i,"'");

outtextxy(306-i,74-i,"'");

outtextxy(293-i,70-i,"'");

outtextxy(295,70-i,"'");

outtextxy(288+i,70+i,"'");

outtextxy(284-i,44+i,"'");

outtextxy(322-i,88-i,"'");

outtextxy(326+i,62+i,"'");

outtextxy(329+i,54-i,"'");

outtextxy(301+i,64-i,"'");

outtextxy(205-i,70-i,"'");

outtextxy(299+i+2,78-i,"'");

outtextxy(303-i,78-i+2,"'");

outtextxy(306+i+1,78-i,"'");

outtextxy(299-i,82+i,"'");

outtextxy(303+i,82+6-i,"'");

outtextxy(306+i,82-i,"'");

setcolor(YELLOW);

outtextxy(400-i,70-i,"'");

outtextxy(404,70-i,"'");

outtextxy(408+i,70+i,"'");

outtextxy(410-i,74+i,"'");

outtextxy(410-i,78-i,"'");

outtextxy(410+i,82+i,"'");

outtextxy(399+i,74-i,"'");

outtextxy(403+i,74-i,"'");

outtextxy(406-i,74-i,"'");

outtextxy(393-i,70-i,"'");

outtextxy(395,70-i,"'");

outtextxy(388+i,70+i,"'");

outtextxy(384-i,44+i,"'");

outtextxy(422-i,88-i,"'");

outtextxy(426+i,62+i,"'");

outtextxy(429+i,54-i,"'");

outtextxy(401+i,64-i,"'");

outtextxy(205-i,70-i,"'");

outtextxy(399+i+2,78-i,"'");

outtextxy(403-i,78-i+2,"'");

outtextxy(406+i+1,78-i,"'");

outtextxy(399-i,82+i,"'");

outtextxy(403+i,82+6-i,"'");

outtextxy(406+i,82-i,"'");

setcolor(6);

outtextxy(500-i,80-i,"'");

outtextxy(504,80-i,"'");

outtextxy(508+i,80+i,"'");

outtextxy(510-i,84+i,"'");

outtextxy(510-i,88-i,"'");

outtextxy(510+i,92+i,"'");

outtextxy(499+i,84-i,"'");

outtextxy(503+i,84-i,"'");

outtextxy(506-i,84-i,"'");

outtextxy(493-i,80-i,"'");

outtextxy(495,80-i,"'");

outtextxy(488+i,80+i,"'");

outtextxy(484-i,54+i,"'");

outtextxy(522-i,98-i,"'");

outtextxy(526+i,72+i,"'");

outtextxy(529+i,64-i,"'");

outtextxy(501+i,74-i,"'");

outtextxy(405-i,80-i,"'");

outtextxy(499+i+2,88-i,"'");

outtextxy(503-i,88-i+2,"'");

outtextxy(506+i+1,88-i,"'");

outtextxy(499-i,82+i,"'");

outtextxy(503+i,82+6-i,"'");

outtextxy(506+i,82-i,"'");

setcolor(9);

outtextxy(100-i,90-i,"'");

outtextxy(104,90-i,"'");

outtextxy(108+i,90+i,"'");

outtextxy(110-i,94+i,"'");

outtextxy(110-i,98-i,"'");

outtextxy(110+i,92+i,"'");

outtextxy(199+i,94-i,"'");

outtextxy(103+i,94-i,"'");

outtextxy(106-i,94-i,"'");

outtextxy(193-i,90-i,"'");

outtextxy(195,90-i,"'");

outtextxy(188+i,90+i,"'");

outtextxy(184-i,64+i,"'");

outtextxy(122-i,98-i,"'");

outtextxy(126+i,82+i,"'");

outtextxy(129+i,74-i,"'");

outtextxy(101+i,84-i,"'");

outtextxy(105-i,90-i,"'");

outtextxy(199+i+2,98-i,"'");

outtextxy(103-i,98-i+2,"'");

outtextxy(106+i+1,98-i,"'");

outtextxy(199-i,92+i,"'");

outtextxy(103+i,92+6-i,"'");

outtextxy(106+i,92-i,"'");

setcolor(5);

outtextxy(100-i,190-i,"'");

outtextxy(104,190-i,"'");

outtextxy(108+i,190+i,"'");

outtextxy(110-i,194+i,"'");

outtextxy(110-i,198-i,"'");

outtextxy(110+i,192+i,"'");

outtextxy(199+i,194-i,"'");

outtextxy(103+i,194-i,"'");

outtextxy(106-i,194-i,"'");

outtextxy(193-i,190-i,"'");

outtextxy(195,190-i,"'");

outtextxy(188+i,190+i,"'");

outtextxy(184-i,164+i,"'");

outtextxy(122-i,198-i,"'");

outtextxy(126+i,182+i,"'");

outtextxy(129+i,174-i,"'");

outtextxy(101+i,184-i,"'");

outtextxy(105-i,190-i,"'");

outtextxy(199+i+2,198-i,"'");

outtextxy(103-i,198-i+2,"'");

outtextxy(106+i+1,198-i,"'");

outtextxy(199-i,192+i,"'");

outtextxy(103+i,192+6-i,"'");

outtextxy(106+i,192-i,"'");

setcolor(3);

outtextxy(500-i,130-i,"'");

outtextxy(504,130-i,"'");

outtextxy(508+i,130+i,"'");

outtextxy(510-i,134+i,"'");

outtextxy(510-i,138-i,"'");

outtextxy(510+i,132+i,"'");

outtextxy(499+i,134-i,"'");

outtextxy(503+i,134-i,"'");

outtextxy(506-i,134-i,"'");

outtextxy(493-i,130-i,"'");

outtextxy(495,130-i,"'");

outtextxy(488+i,130+i,"'");

outtextxy(484-i,114+i,"'");

outtextxy(522-i,138-i,"'");

outtextxy(526+i,112+i,"'");

outtextxy(529+i,124-i,"'");

outtextxy(501+i,124-i,"'");

outtextxy(405-i,130-i,"'");

outtextxy(499+i+2,138-i,"'");

outtextxy(503-i,138-i+2,"'");

outtextxy(506+i+1,138-i,"'");

outtextxy(499-i,132+i,"'");

outtextxy(503+i,132+6-i,"'");

outtextxy(506+i,132-i,"'");

setcolor(10);

outtextxy(400-i,190-i,"'");

outtextxy(404,190-i,"'");

outtextxy(408+i,190+i,"'");

outtextxy(410-i,194+i,"'");

outtextxy(410-i,198-i,"'");

outtextxy(410+i,192+i,"'");

outtextxy(399+i,194-i,"'");

outtextxy(403+i,194-i,"'");

outtextxy(406-i,194-i,"'");

outtextxy(493-i,190-i,"'");

outtextxy(495,190-i,"'");

outtextxy(488+i,190+i,"'");

outtextxy(484-i,164+i,"'");

outtextxy(422-i,198-i,"'");

outtextxy(426+i,162+i,"'");

outtextxy(429+i,174-i,"'");

outtextxy(401+i,174-i,"'");

outtextxy(305-i,190-i,"'");

outtextxy(399+i+2,198-i,"'");

outtextxy(403-i,198-i+2,"'");

outtextxy(406+i+1,198-i,"'");

outtextxy(399-i,192+i,"'");

outtextxy(403+i,192+6-i,"'");

outtextxy(406+i,192-i,"'");

delay(6);

cleardevice();

}

}

//middle cloud

ellipse(200,30,60,240,10,15);

ellipse(225,20,345,165,20,15);

ellipse(262,27,340,160,20,15);

ellipse(268,42,230,50,20,10);

ellipse(226,46,163,340,31,15);

ellipse(300,21,20,182,21,19);

ellipse(325,31,290,100,21,15);

ellipse(299,45,228,20,33,11);

//left cloud

ellipse(40,30,60,240,10,15);

ellipse(65,20,345,165,20,15);

ellipse(105,27,340,160,20,15);

ellipse(109,42,230,50,20,10);

ellipse(66,46,163,340,31,17);

//right cloud

ellipse(450,30,60,240,10,15);

ellipse(475,20,345,165,20,15);

ellipse(512,27,340,160,20,15);

ellipse(518,42,230,50,20,10);

ellipse(476,46,163,340,31,15);

//tree1

//tree1 crown

line(240,420,275,420);

line(240,420,245,405);

line(275,420,270,405);

line(245,405,240,405);

line(270,405,275,405);

line(240,405,245,390);

line(275,405,270,390);

line(245,390,275,390);

line(270,390,275,390);

line(240,390,258,370);

line(275,390,257,370);

//tree1 trunk

line(250,440,250,420);

line(265,440,265,420);

line(250,440,265,440);

line(250,420,265,420);

//tree2

//tree1 crown

line(240+c,420+d,275+c,420+d);

line(240+c,420+d,245+c,405+d);

line(275+c,420+d,270+c,405+d);

line(245+c,405+d,240+c,405+d);

line(270+c,405+d,275+c,405+d);

line(240+c,405+d,245+c,390+d);

line(275+c,405+d,270+c,390+d);

line(245+c,390+d,275+c,390+d);

line(270+c,390+d,275+c,390+d);

line(240+c,390+d,258+c,370+d);

line(275+c,390+d,257+c,370+d);

//tree2 trunk

line(250+c,440+d,250+c,420+d);

line(265+c,440+d,265+c,420+d);

line(250+c,440+d,265+c,440+d);

line(250+c,420+d,265+c,420+d);

//tree3

//tree3 crown

line(240\*m,420\*n,275\*m,420\*n);

line(240\*m,420\*n,245\*m,405\*n);

line(275\*m,420\*n,270\*m,405\*n);

line(245\*m,405\*n,240\*m,405\*n);

line(270\*m,405\*n,275\*m,405\*n);

line(240\*m,405\*n,245\*m,390\*n);

line(275\*m,405\*n,270\*m,390\*n);

line(245\*m,390\*n,275\*m,390\*n);

line(270\*m,390\*n,275\*m,390\*n);

line(240\*m,390\*n,258\*m,370\*n);

line(275\*m,390\*n,257\*m,370\*n);

//tree1 trunk

line(250\*m,440\*n,250\*m,420\*n);

line(265\*m,440\*n,265\*m,420\*n);

line(250\*m,440\*n,265\*m,440\*n);

line(250\*m,420\*n,265\*m,420\*n);

//1st house

//triangle of first house

line(88,400,163,400);

line(100,370,88,400);

line(100,370,112,400);

line(100,370,150,370);

line(150,370,163,400);

line(88,400,163,400);

line(100,370,112,400);

line(100,370,150,370);

line(150,370,163,400);

line(112,400,112,440);

line(88,440,112,440);

line(88,400,88,440);

line(112,440,163,440);

line(163,400,163,440);

line(112,440,163,440);

line(112,400,112,440);

line(88,400,163,400);

//firsthousedoor

line(94,440,94,420);

line(106,440,106,420);

line(94,420,106,420);

//2nd house

//triangle of secound house

line(88+a,400+b,163+a,400+b);

line(100+a,370+b,88+a,400+b);

line(100+a,370+b,112+a,400+b);

line(100+a,370+b,150+a,370+b);

line(150+a,370+b,163+a,400+b);

line(88+a,400+b,163+a,400+b);

line(100+a,370+b,112+a,400+b);

line(100+a,370+b,150+a,370+b);

line(150+a,370+b,163+a,400+b);

line(112+a,400+b,112+a,440+b);

line(88+a,440+b,112+a,440+b);

line(88+a,400+b,88+a,440+b);

line(112+a,440+b,163+a,440+b);

line(163+a,400+b,163+a,440+b);

line(112+a,440+b,163+a,440+b);

line(112+a,400+b,112+a,440+b);

line(88+a,400+b,163+a,400+b);

//door of secound house

line(94+a,440+b,94+a,420+b);

line(106+a,440+b,106+a,420+b);

line(94+a,420+b,106+a,420+b);

//3rd house

//triangle of third house

line(88\*x,400\*y,163\*x,400\*y);

line(100\*x,370\*y,88\*x,400\*y);

line(100\*x,370\*y,112\*x,400\*y);

line(100\*x,370\*y,150\*x,370\*y);

line(150\*x,370\*y,163\*x,400\*y);

line(88\*x,400\*y,163\*x,400\*y);

line(100\*x,370\*y,112\*x,400\*y);

line(100\*x,370\*y,150\*x,370\*y);

line(150\*x,370\*y,163\*x,400\*y);

line(112\*x,400\*y,112\*x,440\*y);

line(88\*x,440\*y,112\*x,440\*y);

line(88\*x,400\*y,88\*x,440\*y);

line(112\*x,440\*y,163\*x,440\*y);

line(163\*x,400\*y,163\*x,440\*y);

line(112\*x,440\*y,163\*x,440\*y);

line(112\*x,400\*y,112\*x,440\*y);

line(88\*x,400\*y,163\*x,400\*y);

//door of third house

line(94\*x,440\*y,94\*x,420\*y);

line(106\*x,440\*y,106\*x,420\*y);

line(94\*x,420\*y,106\*x,420\*y);

//hills

setcolor(GREEN);

for(int j=200;j<=202; j++)

line(0,j,1500,j);

setfillstyle(1,GREEN);

line(0,200,150,100);

line(150,100,250,200);

floodfill(150,170,GREEN);

line(225,200,375,100);

line(375,100,475,200);

floodfill(375,130,GREEN);

line(425,200,550,100);

line(550,100,650,200);

floodfill(550,130,GREEN);

//sun

setcolor(YELLOW);

circle(240,100,40);

setfillstyle(1,YELLOW);

floodfill(240,100,YELLOW);

setcolor(15);

circle(20,300,10);

settextstyle(10,0,4);

setcolor(12);

outtextxy(10,290,"2");

setcolor(15);

line(20,350,18,400); // left

line(20,350,22,400); //right

line(0,400,700,400);

line(10,320,0,350);

line(33,320,40,350);

circle(70,300,10);

settextstyle(10,0,4);

outtextxy(57,290,"0");

line(70,350,68,400); // left

line(70,350,72,400); //right

line(0,400,700,400);

line(57,320,47,350);

line(82,320,90,350);

circle(120,300,10);

settextstyle(10,0,4);

setcolor(14);

outtextxy(110,290,"2");

setcolor(15);

line(120,350,118,400); // left

line(120,350,122,400);//right

line(0,400,700,400);

line(110,320,100,350);

line(133,320,142,350);

circle(170,300,10);

settextstyle(10,0,4);

setcolor(13);

outtextxy(156,290,"2");

setcolor(15);

line(170,350,168,400); // left

line(170,350,172,400); //right

line(0,400,700,400);

line(159,320,147,350);

line(175,317,188,350);

settextstyle(10,0,3);

setcolor(10);

outtextxy(200,350,"FROM");

setcolor(15);

settextstyle(10,0,3);

setcolor(11);

outtextxy(320,350,"GOA");

setcolor(15);

settextstyle(10,0,3);

setcolor(12);

outtextxy(400,350,"UNIVERSITY");

setcolor(15);

}

setcolor(BLUE);

line(200,382,500,382);

setcolor(8);

circle(230,391,9);

circle(300,391,9);

circle(370,391,9);

circle(450,391,9);

settextstyle(10,0,1);

setcolor(11);

outtextxy(200,250,"SEMESTER");

setcolor(13);

outtextxy(350,250,"EXAMS");

setcolor(14);

outtextxy(480,250,"ONLINE");

}