CompactPainter

Realizat de Boroica Marius și Ganea Alexandru-Gabriel

Colegiul Național Pedagogic “Regele Ferdinand”

Sighetu Marmației

Profesor coordonator :Negrea Beatrix

**Cuprins:**

Prezentarea generală a programului……………………………………………..pg.3

Aplicația folosită pentru convertirea caracterelor in pixeli (convert.cpp)………pg.9

CompactPainter.cpp……………………………………………………………..pg.11

Bibliografie……………………………………………………………………...pg.65

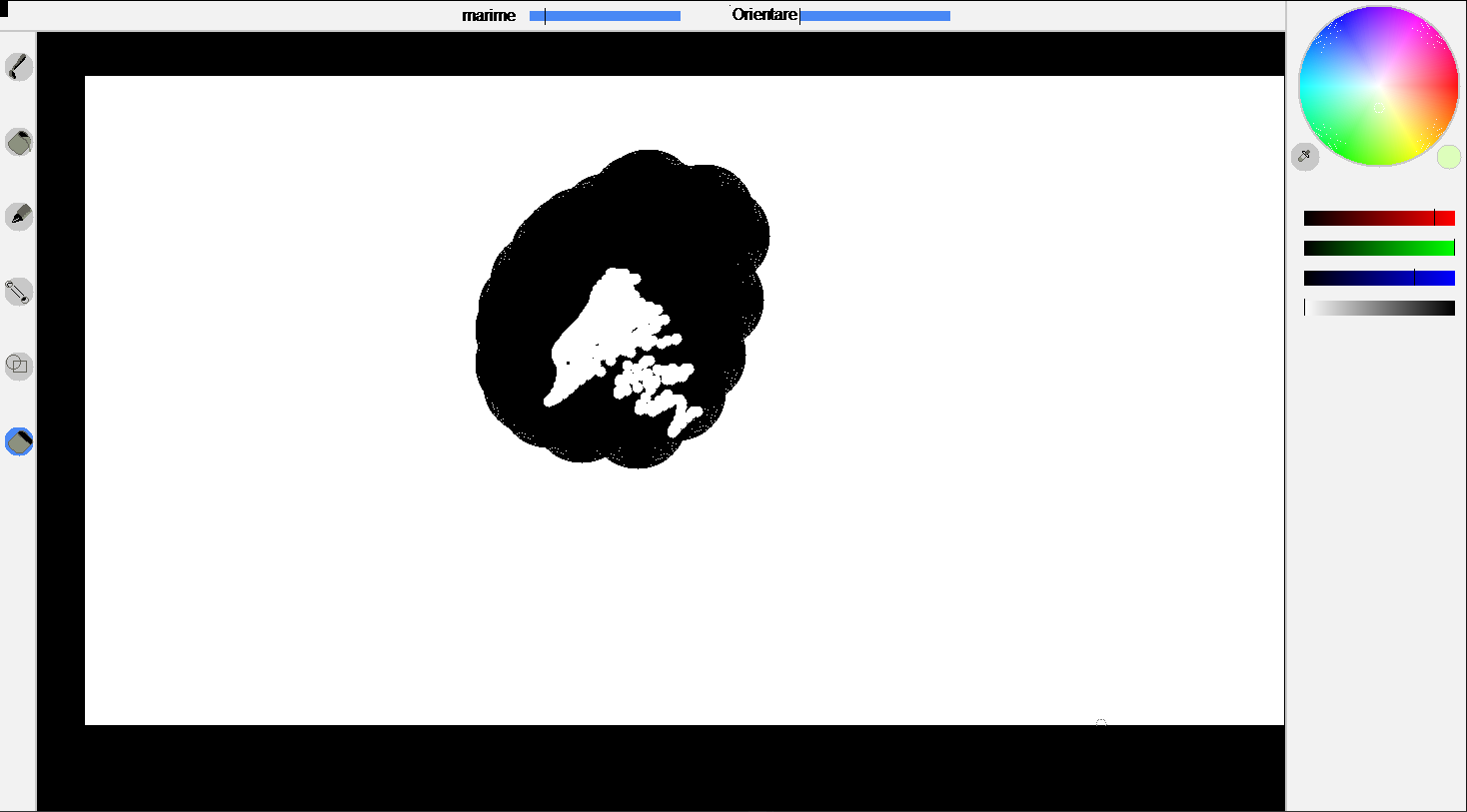
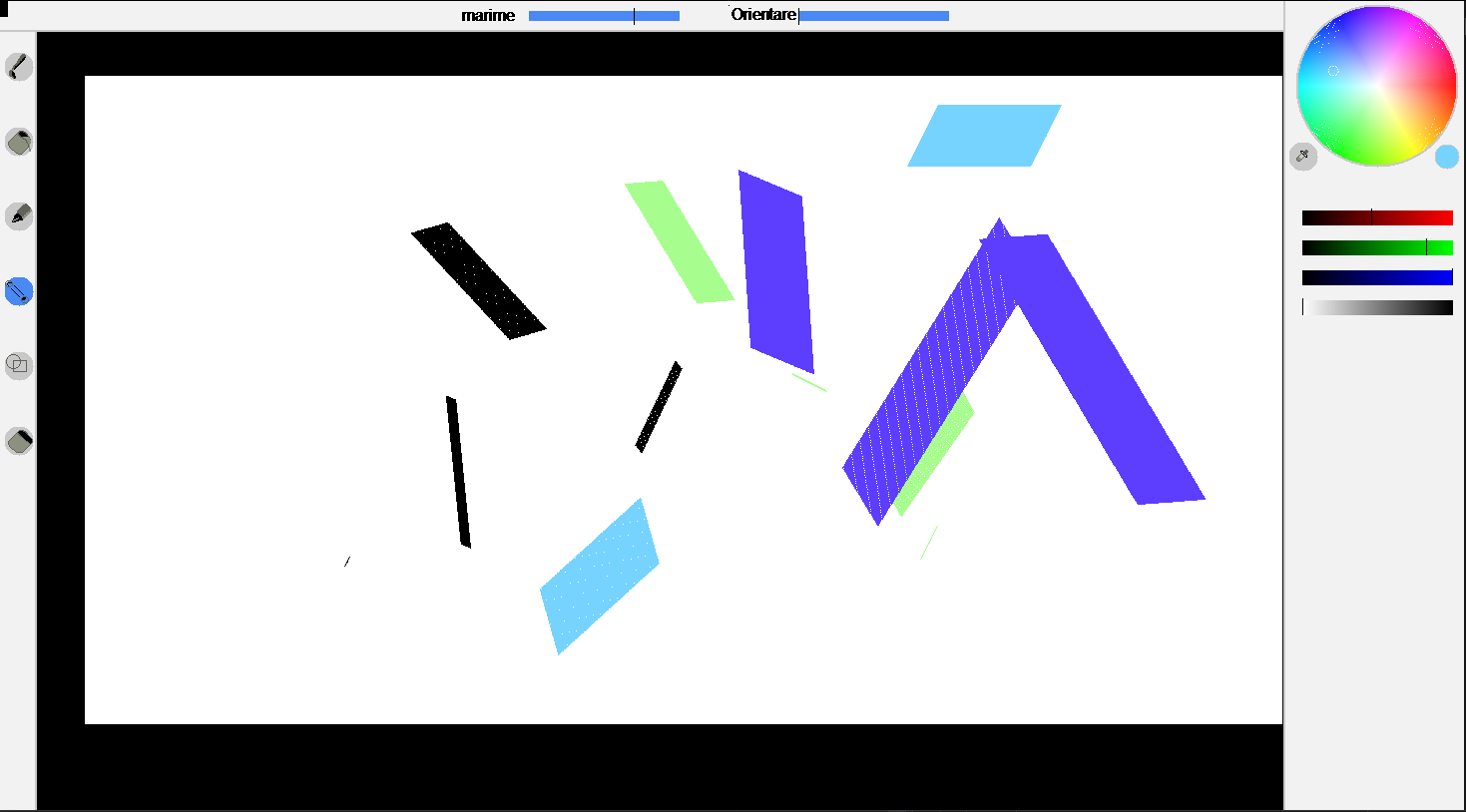
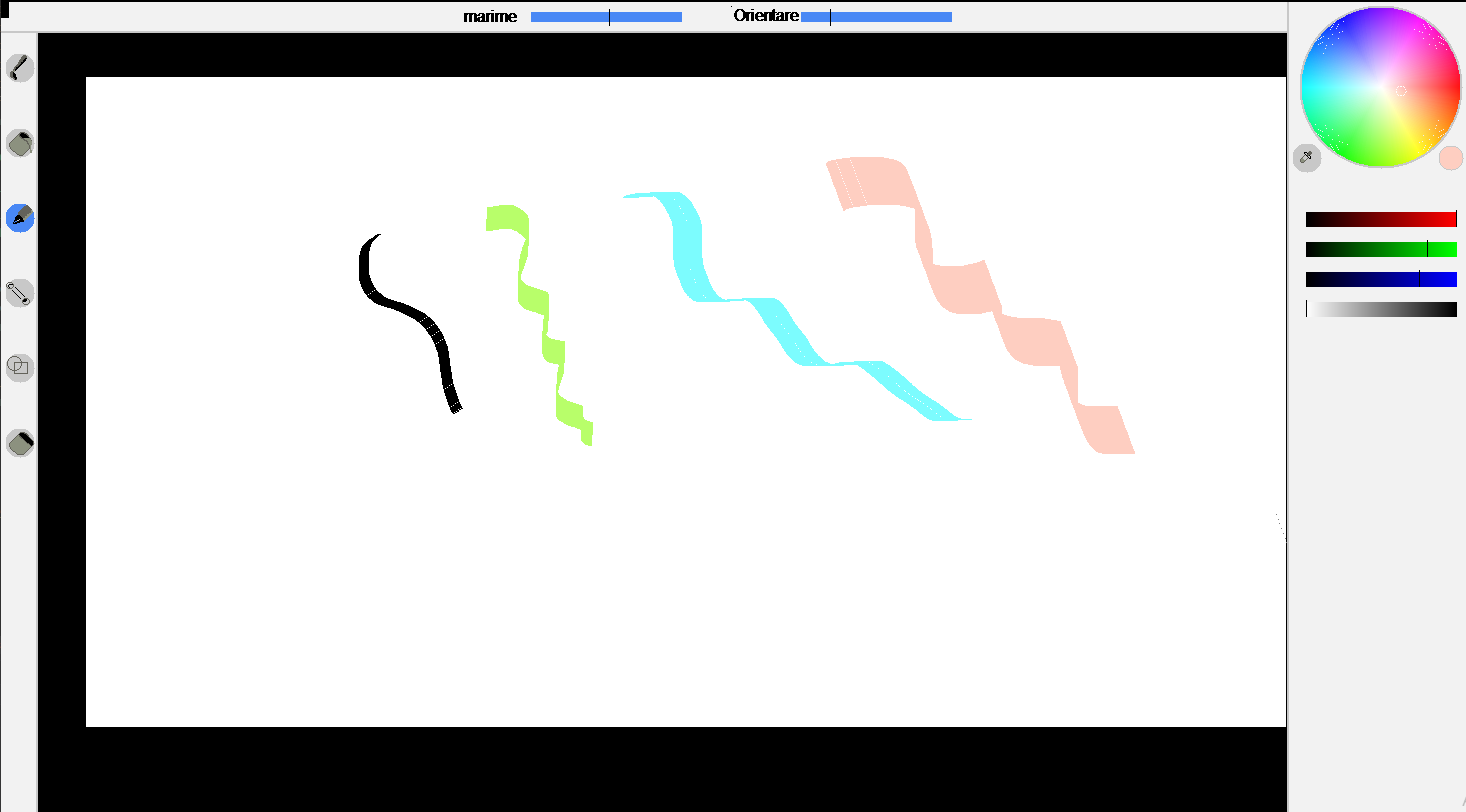
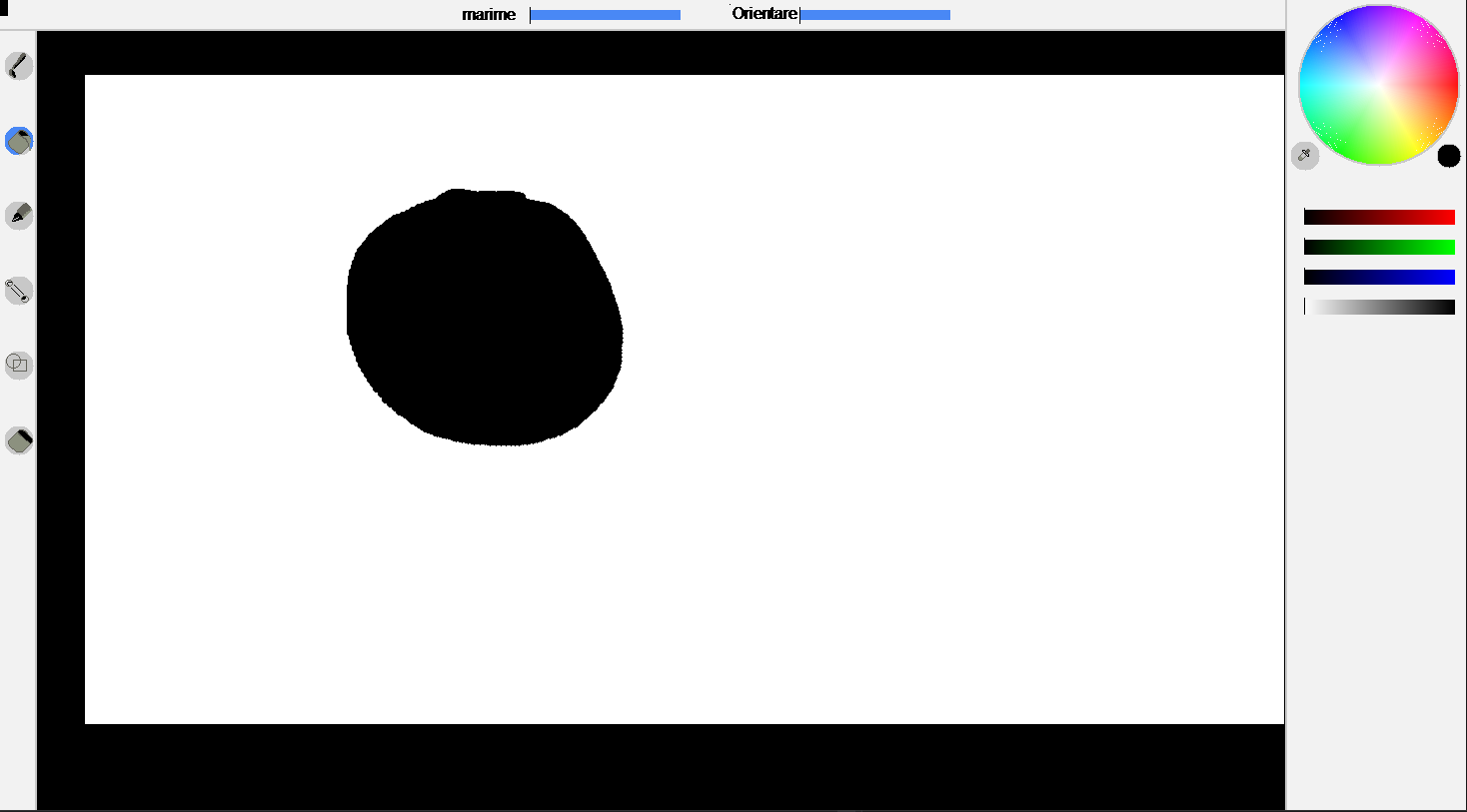
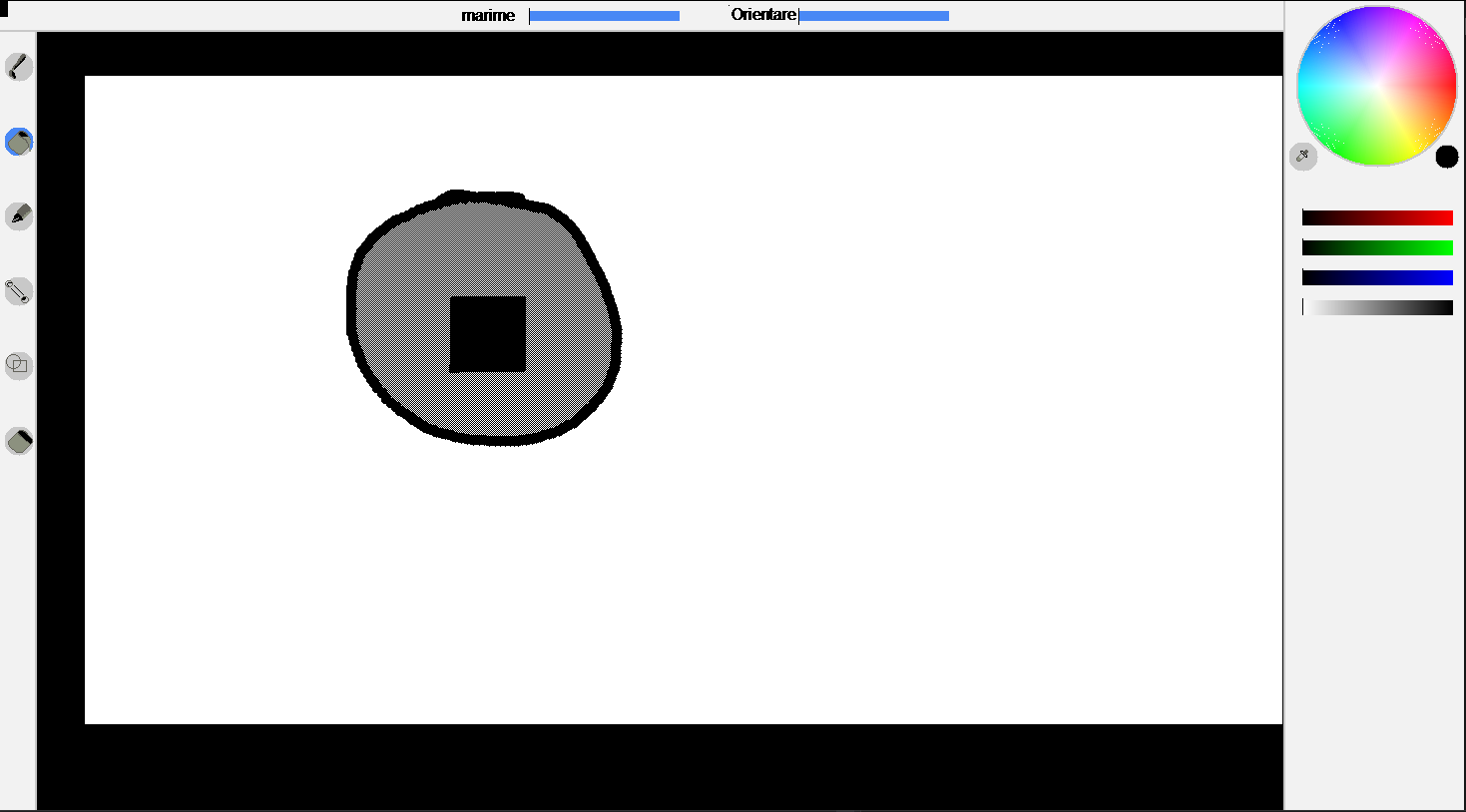
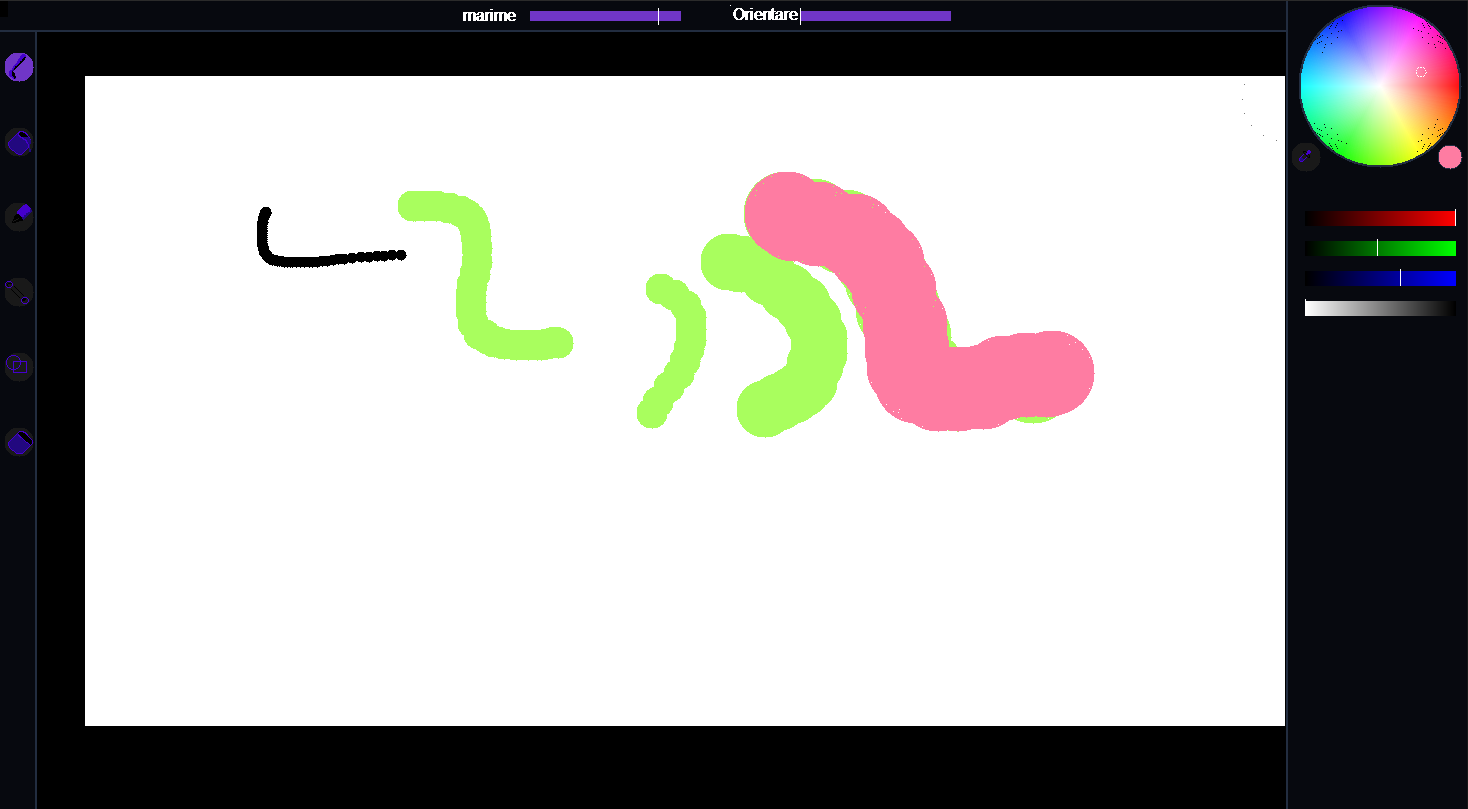
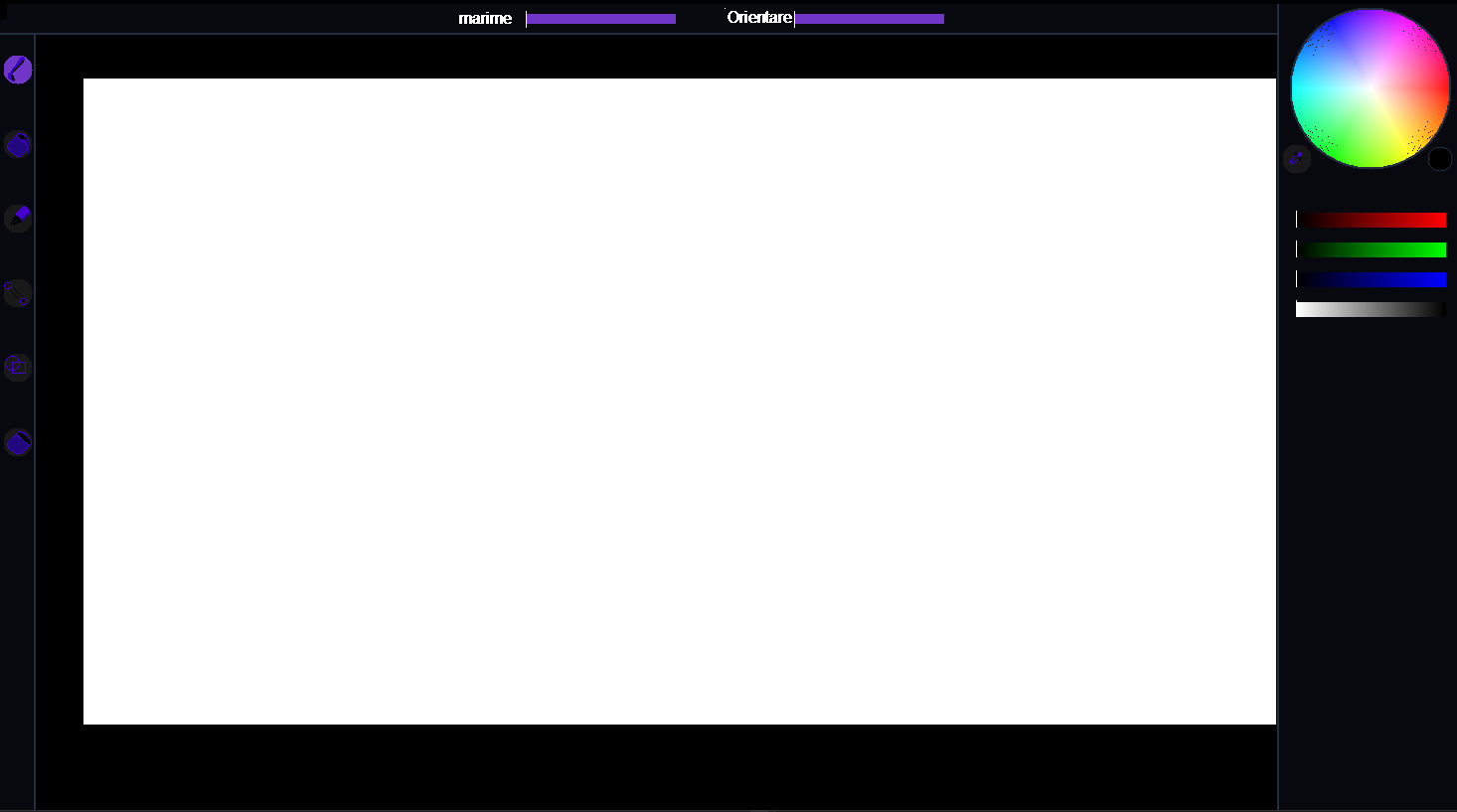
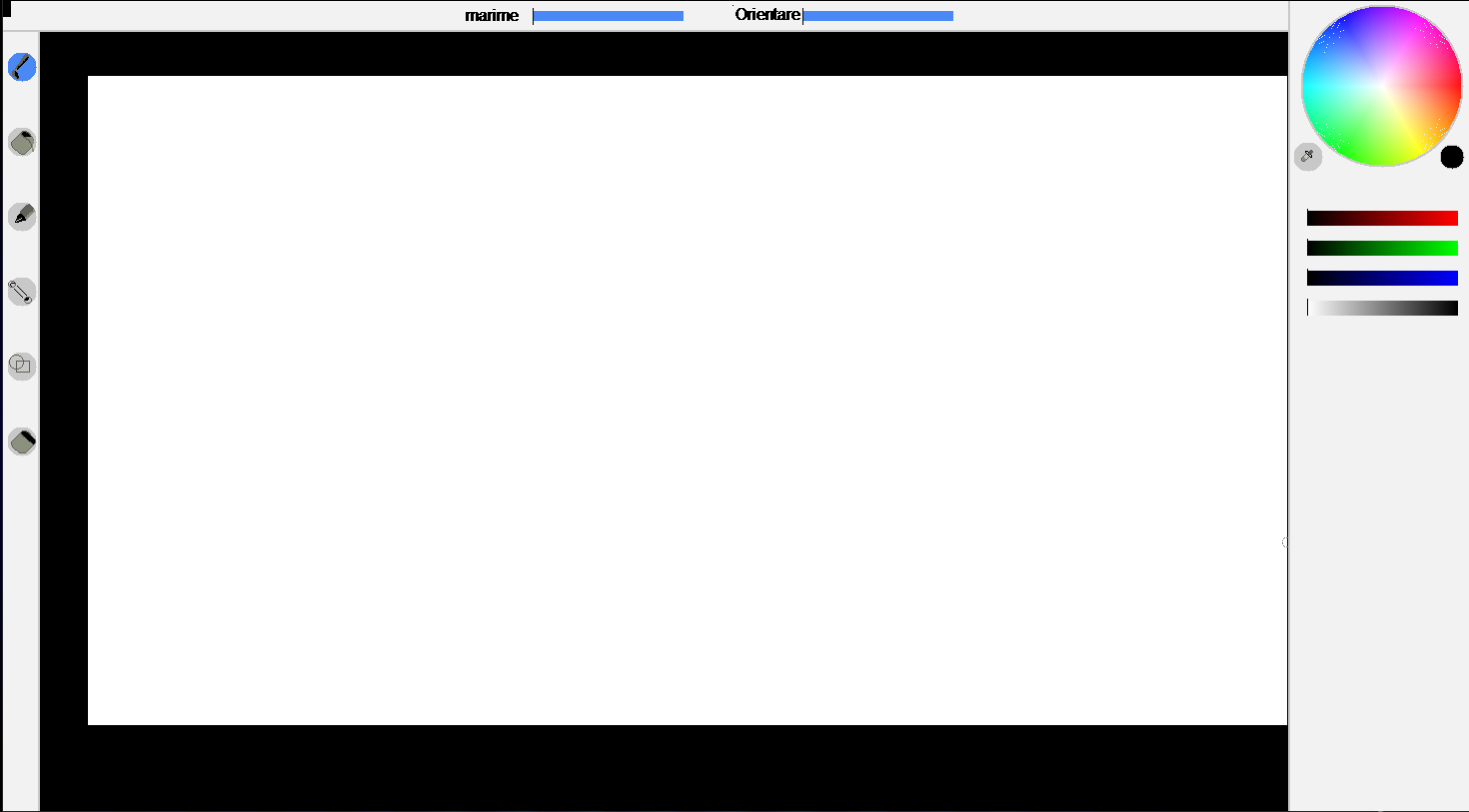
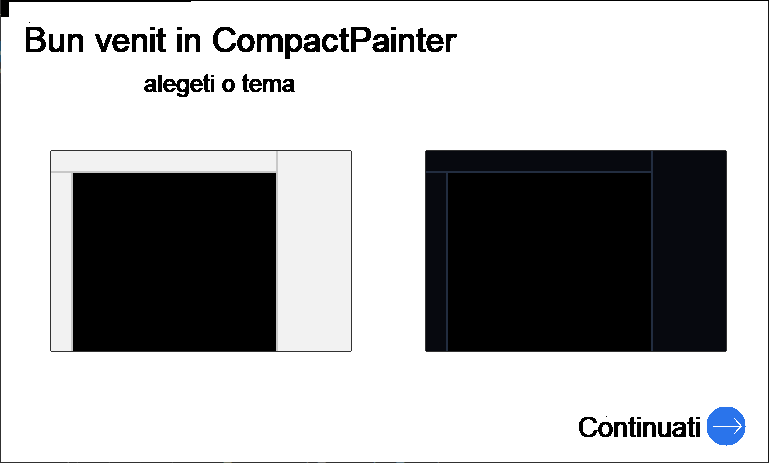
**CompactPainter** este o aplicație în care poți desena cu diferite pensule. Interfața Grafică este user friendly și cât mai simplificată. Pentru a facilita folosirea, există 2 teme pentru interfața grafică, cea nocturnă și cea diurnă.

Ideea programului este de a putea fi creativi într-un mediu foarte portabil, nu este nevoie de instalarea acestuia, mărimea programului nu trece peste 3 MB și posibilitățile pe care programul le poate oferi sunt foarte multe.

Modul prin care utilizatorul interacționează cu programul este prin activarea click-ului drept. Acest program a fost realizat in Microsoft Visual Studio 2019, cu ajutorul următoarelor librării: Iostream, Fstream, Cmath, Thread, Queue,Windows.h.

Pentru convertirea caracterelor în pixeli s-a folosit Microsoft Paint și un program realizat de noi, deși nu este necesar pentru rularea programului principal.

Rapiditatea programului poate fi destul de scazută din cauza funcțiilor GetPixel și SetPixel, care ni s-au părut foarte ok pentru programul acesta și pot fi înțelese de foarte multe persoane fiind ușor de scris.



**Aplicația folosită pentru convertirea caracterelor în pixeli:**

#include <iostream>

#include <fstream>

#include <thread>

#include <windows.h>

using namespace std;

HDC window = GetDC(NULL);

ofstream out("Text.txt");

POINT m,n;

int main() {

GetCursorPos(&m);

while (true) {

if (GetAsyncKeyState((unsigned short)VK\_RBUTTON) & 0x8000) {

cout << 1 << endl;

GetCursorPos(&m);

break;

}

}

this\_thread::sleep\_for(chrono::seconds(1));

while (true) {

if (GetAsyncKeyState((unsigned short)VK\_RBUTTON) & 0x8000) {

cout << 2 << endl;

GetCursorPos(&n);

break;

}

}

long long el=0;

this\_thread::sleep\_for(chrono::seconds(5));

for (int y = m.y; y < n.y; y++) {

printf("%d\n", n.y - y);

for (int x = m.x; x < n.x; x++) {

if (GetPixel(window, x, y) != RGB(255, 255, 255)) {

out << '{' << x - m.x << ',' << y - m.y << "},";

el++;

}

}

}

out << '\n' << el;

out.close();

}

**Partea tehnică a aplicației:**

#include <iostream>

#include <fstream>

#include <cmath>

#include <thread>

#include <queue>

#include <Windows.h>

#define pi 3.14159265358

using namespace std;

HWND consoleWindow = GetConsoleWindow();

HDC consoleDC = GetDC(consoleWindow);

RECT rect;

string brush\_t = "paint\_brush";

POINT mouse\_pos;

int switch\_tool = 1;

int th = 1;

double r = 0, g = 0, b = 0;

double shade\_value = 100;

struct brush {

double size = 5;

double orientation = 330;

}paint\_brush, pen, line, eraser;

struct pixel {

double x;

double y;

COLORREF color;

};

int intro[7366][2]{

{152,0},{153,0},{154,0},{201,0},{202,0},{203,0},{480,0},{481,0},{482,0},{5,1},{6,1},{7,1},{8,1},{9,1},{10,1},{11,1},{12,1},{13,1},{14,1},{15,1},{16,1},{17,1},{18,1},{151,1},{152,1},{153,1},{154,1},{155,1},{200,1},{201,1},{202,1},{203,1},{204,1},{265,1},{266,1},{267,1},{268,1},{269,1},{270,1},{271,1},{272,1},{273,1},{274,1},{275,1},{432,1},{433,1},{434,1},{435,1},{436,1},{437,1},{438,1},{439,1},{440,1},{441,1},{442,1},{443,1},{444,1},{445,1},{479,1},{480,1},{481,1},{482,1},{483,1},{4,2},{5,2},{6,2},{7,2},{8,2},{9,2},{10,2},{11,2},{12,2},{13,2},{14,2},{15,2},{16,2},{17,2},{18,2},{19,2},{20,2},{150,2},{151,2},{152,2},{153,2},{154,2},{155,2},{156,2},{199,2},{200,2},{201,2},{202,2},{203,2},{204,2},{205,2},{263,2},{264,2},{265,2},{266,2},{267,2},{268,2},{269,2},{270,2},{271,2},{272,2},{273,2},{274,2},{275,2},{276,2},{277,2},{431,2},{432,2},{433,2},{434,2},{435,2},{436,2},{437,2},{438,2},{439,2},{440,2},{441,2},{442,2},{443,2},{444,2},{445,2},{446,2},{447,2},{478,2},{479,2},{480,2},{481,2},{482,2},{483,2},{484,2},{4,3},{5,3},{6,3},{7,3},{8,3},{9,3},{10,3},{11,3},{12,3},{13,3},{14,3},{15,3},{16,3},{17,3},{18,3},{19,3},{20,3},{21,3},{150,3},{151,3},{152,3},{153,3},{154,3},{155,3},{156,3},{163,3},{164,3},{165,3},{166,3},{167,3},{199,3},{200,3},{201,3},{202,3},{203,3},{204,3},{205,3},{262,3},{263,3},{264,3},{265,3},{266,3},{267,3},{268,3},{269,3},{270,3},{271,3},{272,3},{273,3},{274,3},{275,3},{276,3},{277,3},{278,3},{416,3},{417,3},{418,3},{419,3},{420,3},{431,3},{432,3},{433,3},{434,3},{435,3},{436,3},{437,3},{438,3},{439,3},{440,3},{441,3},{442,3},{443,3},{444,3},{445,3},{446,3},{447,3},{448,3},{478,3},{479,3},{480,3},{481,3},{482,3},{483,3},{484,3},{516,3},{517,3},{518,3},{519,3},{520,3},{4,4},{5,4},{6,4},{7,4},{8,4},{9,4},{10,4},{11,4},{12,4},{13,4},{14,4},{15,4},{16,4},{17,4},{18,4},{19,4},{20,4},{21,4},{22,4},{151,4},{152,4},{153,4},{154,4},{155,4},{163,4},{164,4},{165,4},{166,4},{167,4},{200,4},{201,4},{202,4},{203,4},{204,4},{261,4},{262,4},{263,4},{264,4},{265,4},{266,4},{267,4},{268,4},{269,4},{270,4},{271,4},{272,4},{273,4},{274,4},{275,4},{276,4},{277,4},{278,4},{279,4},{416,4},{417,4},{418,4},{419,4},{420,4},{431,4},{432,4},{433,4},{434,4},{435,4},{436,4},{437,4},{438,4},{439,4},{440,4},{441,4},{442,4},{443,4},{444,4},{445,4},{446,4},{447,4},{448,4},{449,4},{479,4},{480,4},{481,4},{482,4},{483,4},{516,4},{517,4},{518,4},{519,4},{520,4},{4,5},{5,5},{6,5},{7,5},{8,5},{9,5},{17,5},{18,5},{19,5},{20,5},{21,5},{22,5},{163,5},{164,5},{165,5},{166,5},{167,5},{260,5},{261,5},{262,5},{263,5},{264,5},{265,5},{266,5},{274,5},{275,5},{276,5},{277,5},{278,5},{279,5},{416,5},{417,5},{418,5},{419,5},{420,5},{431,5},{432,5},{433,5},{434,5},{435,5},{436,5},{443,5},{444,5},{445,5},{446,5},{447,5},{448,5},{449,5},{516,5},{517,5},{518,5},{519,5},{520,5},{4,6},{5,6},{6,6},{7,6},{8,6},{9,6},{18,6},{19,6},{20,6},{21,6},{22,6},{23,6},{163,6},{164,6},{165,6},{166,6},{167,6},{259,6},{260,6},{261,6},{262,6},{263,6},{264,6},{265,6},{276,6},{277,6},{278,6},{279,6},{416,6},{417,6},{418,6},{419,6},{420,6},{431,6},{432,6},{433,6},{434,6},{435,6},{436,6},{445,6},{446,6},{447,6},{448,6},{449,6},{450,6},{516,6},{517,6},{518,6},{519,6},{520,6},{4,7},{5,7},{6,7},{7,7},{8,7},{9,7},{18,7},{19,7},{20,7},{21,7},{22,7},{23,7},{163,7},{164,7},{165,7},{166,7},{167,7},{259,7},{260,7},{261,7},{262,7},{263,7},{264,7},{277,7},{278,7},{416,7},{417,7},{418,7},{419,7},{420,7},{431,7},{432,7},{433,7},{434,7},{435,7},{436,7},{445,7},{446,7},{447,7},{448,7},{449,7},{450,7},{516,7},{517,7},{518,7},{519,7},{520,7},{4,8},{5,8},{6,8},{7,8},{8,8},{9,8},{18,8},{19,8},{20,8},{21,8},{22,8},{23,8},{163,8},{164,8},{165,8},{166,8},{167,8},{258,8},{259,8},{260,8},{261,8},{262,8},{263,8},{416,8},{417,8},{418,8},{419,8},{420,8},{431,8},{432,8},{433,8},{434,8},{435,8},{436,8},{446,8},{447,8},{448,8},{449,8},{450,8},{516,8},{517,8},{518,8},{519,8},{520,8},{4,9},{5,9},{6,9},{7,9},{8,9},{9,9},{18,9},{19,9},{20,9},{21,9},{22,9},{23,9},{30,9},{31,9},{32,9},{33,9},{34,9},{44,9},{45,9},{46,9},{47,9},{48,9},{55,9},{56,9},{57,9},{58,9},{59,9},{62,9},{63,9},{64,9},{65,9},{66,9},{67,9},{68,9},{69,9},{70,9},{77,9},{78,9},{79,9},{80,9},{81,9},{82,9},{93,9},{94,9},{95,9},{96,9},{97,9},{98,9},{107,9},{108,9},{109,9},{110,9},{111,9},{112,9},{113,9},{114,9},{115,9},{126,9},{127,9},{128,9},{129,9},{130,9},{133,9},{134,9},{135,9},{136,9},{137,9},{138,9},{139,9},{140,9},{141,9},{151,9},{152,9},{153,9},{154,9},{155,9},{159,9},{160,9},{161,9},{162,9},{163,9},{164,9},{165,9},{166,9},{167,9},{168,9},{169,9},{170,9},{171,9},{172,9},{173,9},{200,9},{201,9},{202,9},{203,9},{204,9},{211,9},{212,9},{213,9},{214,9},{215,9},{218,9},{219,9},{220,9},{221,9},{222,9},{223,9},{224,9},{225,9},{226,9},{258,9},{259,9},{260,9},{261,9},{262,9},{263,9},{289,9},{290,9},{291,9},{292,9},{293,9},{294,9},{295,9},{296,9},{297,9},{298,9},{309,9},{310,9},{311,9},{312,9},{313,9},{316,9},{317,9},{318,9},{319,9},{320,9},{321,9},{322,9},{323,9},{330,9},{331,9},{332,9},{333,9},{334,9},{335,9},{336,9},{337,9},{347,9},{348,9},{349,9},{350,9},{351,9},{354,9},{355,9},{356,9},{357,9},{358,9},{359,9},{360,9},{361,9},{362,9},{375,9},{376,9},{377,9},{378,9},{379,9},{380,9},{381,9},{382,9},{383,9},{384,9},{399,9},{400,9},{401,9},{402,9},{403,9},{404,9},{405,9},{406,9},{407,9},{408,9},{412,9},{413,9},{414,9},{415,9},{416,9},{417,9},{418,9},{419,9},{420,9},{421,9},{422,9},{423,9},{424,9},{425,9},{426,9},{431,9},{432,9},{433,9},{434,9},{435,9},{436,9},{446,9},{447,9},{448,9},{449,9},{450,9},{459,9},{460,9},{461,9},{462,9},{463,9},{464,9},{465,9},{466,9},{467,9},{468,9},{479,9},{480,9},{481,9},{482,9},{483,9},{490,9},{491,9},{492,9},{493,9},{494,9},{497,9},{498,9},{499,9},{500,9},{501,9},{502,9},{503,9},{504,9},{505,9},{512,9},{513,9},{514,9},{515,9},{516,9},{517,9},{518,9},{519,9},{520,9},{521,9},{522,9},{523,9},{524,9},{525,9},{526,9},{536,9},{537,9},{538,9},{539,9},{540,9},{541,9},{542,9},{543,9},{544,9},{555,9},{556,9},{557,9},{558,9},{559,9},{561,9},{562,9},{563,9},{564,9},{565,9},{566,9},{567,9},{4,10},{5,10},{6,10},{7,10},{8,10},{9,10},{18,10},{19,10},{20,10},{21,10},{22,10},{23,10},{30,10},{31,10},{32,10},{33,10},{34,10},{44,10},{45,10},{46,10},{47,10},{48,10},{55,10},{56,10},{57,10},{58,10},{59,10},{60,10},{61,10},{62,10},{63,10},{64,10},{65,10},{66,10},{67,10},{68,10},{69,10},{70,10},{71,10},{77,10},{78,10},{79,10},{80,10},{81,10},{82,10},{93,10},{94,10},{95,10},{96,10},{97,10},{105,10},{106,10},{107,10},{108,10},{109,10},{110,10},{111,10},{112,10},{113,10},{114,10},{115,10},{116,10},{117,10},{126,10},{127,10},{128,10},{129,10},{130,10},{131,10},{132,10},{133,10},{134,10},{135,10},{136,10},{137,10},{138,10},{139,10},{140,10},{141,10},{142,10},{151,10},{152,10},{153,10},{154,10},{155,10},{159,10},{160,10},{161,10},{162,10},{163,10},{164,10},{165,10},{166,10},{167,10},{168,10},{169,10},{170,10},{171,10},{172,10},{173,10},{200,10},{201,10},{202,10},{203,10},{204,10},{211,10},{212,10},{213,10},{214,10},{215,10},{216,10},{217,10},{218,10},{219,10},{220,10},{221,10},{222,10},{223,10},{224,10},{225,10},{226,10},{227,10},{257,10},{258,10},{259,10},{260,10},{261,10},{262,10},{287,10},{288,10},{289,10},{290,10},{291,10},{292,10},{293,10},{294,10},{295,10},{296,10},{297,10},{298,10},{299,10},{300,10},{309,10},{310,10},{311,10},{312,10},{313,10},{314,10},{315,10},{316,10},{317,10},{318,10},{319,10},{320,10},{321,10},{322,10},{323,10},{324,10},{325,10},{328,10},{329,10},{330,10},{331,10},{332,10},{333,10},{334,10},{335,10},{336,10},{337,10},{338,10},{347,10},{348,10},{349,10},{350,10},{351,10},{353,10},{354,10},{355,10},{356,10},{357,10},{358,10},{359,10},{360,10},{361,10},{362,10},{363,10},{372,10},{373,10},{374,10},{375,10},{376,10},{377,10},{378,10},{379,10},{380,10},{381,10},{382,10},{383,10},{384,10},{385,10},{386,10},{397,10},{398,10},{399,10},{400,10},{401,10},{402,10},{403,10},{404,10},{405,10},{406,10},{407,10},{408,10},{409,10},{410,10},{412,10},{413,10},{414,10},{415,10},{416,10},{417,10},{418,10},{419,10},{420,10},{421,10},{422,10},{423,10},{424,10},{425,10},{426,10},{431,10},{432,10},{433,10},{434,10},{435,10},{436,10},{446,10},{447,10},{448,10},{449,10},{450,10},{456,10},{457,10},{458,10},{459,10},{460,10},{461,10},{462,10},{463,10},{464,10},{465,10},{466,10},{467,10},{468,10},{469,10},{470,10},{479,10},{480,10},{481,10},{482,10},{483,10},{490,10},{491,10},{492,10},{493,10},{494,10},{495,10},{496,10},{497,10},{498,10},{499,10},{500,10},{501,10},{502,10},{503,10},{504,10},{505,10},{506,10},{512,10},{513,10},{514,10},{515,10},{516,10},{517,10},{518,10},{519,10},{520,10},{521,10},{522,10},{523,10},{524,10},{525,10},{526,10},{534,10},{535,10},{536,10},{537,10},{538,10},{539,10},{540,10},{541,10},{542,10},{543,10},{544,10},{545,10},{546,10},{555,10},{556,10},{557,10},{558,10},{559,10},{560,10},{561,10},{562,10},{563,10},{564,10},{565,10},{566,10},{567,10},{568,10},{4,11},{5,11},{6,11},{7,11},{8,11},{9,11},{18,11},{19,11},{20,11},{21,11},{22,11},{30,11},{31,11},{32,11},{33,11},{34,11},{44,11},{45,11},{46,11},{47,11},{48,11},{55,11},{56,11},{57,11},{58,11},{59,11},{60,11},{61,11},{62,11},{63,11},{64,11},{65,11},{66,11},{67,11},{68,11},{69,11},{70,11},{71,11},{72,11},{78,11},{79,11},{80,11},{81,11},{82,11},{93,11},{94,11},{95,11},{96,11},{97,11},{104,11},{105,11},{106,11},{107,11},{108,11},{109,11},{110,11},{111,11},{112,11},{113,11},{114,11},{115,11},{116,11},{117,11},{118,11},{126,11},{127,11},{128,11},{129,11},{130,11},{131,11},{132,11},{133,11},{134,11},{135,11},{136,11},{137,11},{138,11},{139,11},{140,11},{141,11},{142,11},{143,11},{151,11},{152,11},{153,11},{154,11},{155,11},{159,11},{160,11},{161,11},{162,11},{163,11},{164,11},{165,11},{166,11},{167,11},{168,11},{169,11},{170,11},{171,11},{172,11},{173,11},{200,11},{201,11},{202,11},{203,11},{204,11},{211,11},{212,11},{213,11},{214,11},{215,11},{216,11},{217,11},{218,11},{219,11},{220,11},{221,11},{222,11},{223,11},{224,11},{225,11},{226,11},{227,11},{228,11},{257,11},{258,11},{259,11},{260,11},{261,11},{262,11},{286,11},{287,11},{288,11},{289,11},{290,11},{291,11},{292,11},{293,11},{294,11},{295,11},{296,11},{297,11},{298,11},{299,11},{300,11},{301,11},{309,11},{310,11},{311,11},{312,11},{313,11},{314,11},{315,11},{316,11},{317,11},{318,11},{319,11},{320,11},{321,11},{322,11},{323,11},{324,11},{325,11},{327,11},{328,11},{329,11},{330,11},{331,11},{332,11},{333,11},{334,11},{335,11},{336,11},{337,11},{338,11},{339,11},{347,11},{348,11},{349,11},{350,11},{351,11},{352,11},{353,11},{354,11},{355,11},{356,11},{357,11},{358,11},{359,11},{360,11},{361,11},{362,11},{363,11},{364,11},{372,11},{373,11},{374,11},{375,11},{376,11},{377,11},{378,11},{379,11},{380,11},{381,11},{382,11},{383,11},{384,11},{385,11},{386,11},{387,11},{396,11},{397,11},{398,11},{399,11},{400,11},{401,11},{402,11},{403,11},{404,11},{405,11},{406,11},{407,11},{408,11},{409,11},{410,11},{412,11},{413,11},{414,11},{415,11},{416,11},{417,11},{418,11},{419,11},{420,11},{421,11},{422,11},{423,11},{424,11},{425,11},{426,11},{431,11},{432,11},{433,11},{434,11},{435,11},{436,11},{445,11},{446,11},{447,11},{448,11},{449,11},{450,11},{456,11},{457,11},{458,11},{459,11},{460,11},{461,11},{462,11},{463,11},{464,11},{465,11},{466,11},{467,11},{468,11},{469,11},{470,11},{471,11},{479,11},{480,11},{481,11},{482,11},{483,11},{490,11},{491,11},{492,11},{493,11},{494,11},{495,11},{496,11},{497,11},{498,11},{499,11},{500,11},{501,11},{502,11},{503,11},{504,11},{505,11},{506,11},{507,11},{512,11},{513,11},{514,11},{515,11},{516,11},{517,11},{518,11},{519,11},{520,11},{521,11},{522,11},{523,11},{524,11},{525,11},{526,11},{533,11},{534,11},{535,11},{536,11},{537,11},{538,11},{539,11},{540,11},{541,11},{542,11},{543,11},{544,11},{545,11},{546,11},{547,11},{555,11},{556,11},{557,11},{558,11},{559,11},{560,11},{561,11},{562,11},{563,11},{564,11},{565,11},{566,11},{567,11},{568,11},{4,12},{5,12},{6,12},{7,12},{8,12},{9,12},{17,12},{18,12},{19,12},{20,12},{21,12},{22,12},{30,12},{31,12},{32,12},{33,12},{34,12},{44,12},{45,12},{46,12},{47,12},{48,12},{55,12},{56,12},{57,12},{58,12},{59,12},{60,12},{61,12},{62,12},{63,12},{64,12},{65,12},{66,12},{67,12},{68,12},{69,12},{70,12},{71,12},{72,12},{78,12},{79,12},{80,12},{81,12},{82,12},{83,12},{92,12},{93,12},{94,12},{95,12},{96,12},{97,12},{103,12},{104,12},{105,12},{106,12},{107,12},{108,12},{109,12},{112,12},{113,12},{114,12},{115,12},{116,12},{117,12},{118,12},{119,12},{126,12},{127,12},{128,12},{129,12},{130,12},{131,12},{132,12},{133,12},{134,12},{135,12},{136,12},{137,12},{138,12},{139,12},{140,12},{141,12},{142,12},{143,12},{151,12},{152,12},{153,12},{154,12},{155,12},{160,12},{161,12},{162,12},{163,12},{164,12},{165,12},{166,12},{167,12},{168,12},{169,12},{170,12},{171,12},{172,12},{200,12},{201,12},{202,12},{203,12},{204,12},{211,12},{212,12},{213,12},{214,12},{215,12},{216,12},{217,12},{218,12},{219,12},{220,12},{221,12},{222,12},{223,12},{224,12},{225,12},{226,12},{227,12},{228,12},{257,12},{258,12},{259,12},{260,12},{261,12},{262,12},{285,12},{286,12},{287,12},{288,12},{289,12},{290,12},{291,12},{292,12},{293,12},{294,12},{295,12},{296,12},{297,12},{298,12},{299,12},{300,12},{301,12},{302,12},{309,12},{310,12},{311,12},{312,12},{313,12},{314,12},{315,12},{316,12},{317,12},{318,12},{319,12},{320,12},{321,12},{322,12},{323,12},{324,12},{325,12},{326,12},{327,12},{328,12},{329,12},{330,12},{331,12},{332,12},{333,12},{334,12},{335,12},{336,12},{337,12},{338,12},{339,12},{340,12},{347,12},{348,12},{349,12},{350,12},{351,12},{352,12},{353,12},{354,12},{355,12},{356,12},{357,12},{358,12},{359,12},{360,12},{361,12},{362,12},{363,12},{364,12},{365,12},{371,12},{372,12},{373,12},{374,12},{375,12},{376,12},{377,12},{378,12},{379,12},{380,12},{381,12},{382,12},{383,12},{384,12},{385,12},{386,12},{387,12},{396,12},{397,12},{398,12},{399,12},{400,12},{401,12},{402,12},{403,12},{404,12},{405,12},{406,12},{407,12},{408,12},{409,12},{410,12},{413,12},{414,12},{415,12},{416,12},{417,12},{418,12},{419,12},{420,12},{421,12},{422,12},{423,12},{424,12},{425,12},{431,12},{432,12},{433,12},{434,12},{435,12},{436,12},{445,12},{446,12},{447,12},{448,12},{449,12},{450,12},{455,12},{456,12},{457,12},{458,12},{459,12},{460,12},{461,12},{462,12},{463,12},{464,12},{465,12},{466,12},{467,12},{468,12},{469,12},{470,12},{471,12},{479,12},{480,12},{481,12},{482,12},{483,12},{490,12},{491,12},{492,12},{493,12},{494,12},{495,12},{496,12},{497,12},{498,12},{499,12},{500,12},{501,12},{502,12},{503,12},{504,12},{505,12},{506,12},{507,12},{513,12},{514,12},{515,12},{516,12},{517,12},{518,12},{519,12},{520,12},{521,12},{522,12},{523,12},{524,12},{525,12},{532,12},{533,12},{534,12},{535,12},{536,12},{537,12},{538,12},{541,12},{542,12},{543,12},{544,12},{545,12},{546,12},{547,12},{548,12},{555,12},{556,12},{557,12},{558,12},{559,12},{560,12},{561,12},{562,12},{563,12},{564,12},{565,12},{566,12},{567,12},{568,12},{4,13},{5,13},{6,13},{7,13},{8,13},{9,13},{10,13},{11,13},{12,13},{13,13},{14,13},{15,13},{16,13},{17,13},{18,13},{19,13},{20,13},{21,13},{30,13},{31,13},{32,13},{33,13},{34,13},{44,13},{45,13},{46,13},{47,13},{48,13},{55,13},{56,13},{57,13},{58,13},{59,13},{60,13},{61,13},{62,13},{67,13},{68,13},{69,13},{70,13},{71,13},{72,13},{73,13},{78,13},{79,13},{80,13},{81,13},{82,13},{83,13},{92,13},{93,13},{94,13},{95,13},{96,13},{102,13},{103,13},{104,13},{105,13},{106,13},{107,13},{114,13},{115,13},{116,13},{117,13},{118,13},{119,13},{126,13},{127,13},{128,13},{129,13},{130,13},{131,13},{132,13},{133,13},{138,13},{139,13},{140,13},{141,13},{142,13},{143,13},{144,13},{151,13},{152,13},{153,13},{154,13},{155,13},{163,13},{164,13},{165,13},{166,13},{167,13},{200,13},{201,13},{202,13},{203,13},{204,13},{211,13},{212,13},{213,13},{214,13},{215,13},{216,13},{217,13},{218,13},{223,13},{224,13},{225,13},{226,13},{227,13},{228,13},{229,13},{257,13},{258,13},{259,13},{260,13},{261,13},{262,13},{284,13},{285,13},{286,13},{287,13},{288,13},{289,13},{290,13},{297,13},{298,13},{299,13},{300,13},{301,13},{302,13},{303,13},{309,13},{310,13},{311,13},{312,13},{313,13},{314,13},{315,13},{316,13},{321,13},{322,13},{323,13},{324,13},{325,13},{326,13},{327,13},{328,13},{329,13},{330,13},{335,13},{336,13},{337,13},{338,13},{339,13},{340,13},{347,13},{348,13},{349,13},{350,13},{351,13},{352,13},{353,13},{354,13},{360,13},{361,13},{362,13},{363,13},{364,13},{365,13},{366,13},{372,13},{373,13},{374,13},{375,13},{383,13},{384,13},{385,13},{386,13},{387,13},{388,13},{395,13},{396,13},{397,13},{398,13},{399,13},{400,13},{407,13},{408,13},{409,13},{410,13},{416,13},{417,13},{418,13},{419,13},{420,13},{431,13},{432,13},{433,13},{434,13},{435,13},{436,13},{444,13},{445,13},{446,13},{447,13},{448,13},{449,13},{450,13},{456,13},{457,13},{458,13},{459,13},{467,13},{468,13},{469,13},{470,13},{471,13},{472,13},{479,13},{480,13},{481,13},{482,13},{483,13},{490,13},{491,13},{492,13},{493,13},{494,13},{495,13},{496,13},{497,13},{502,13},{503,13},{504,13},{505,13},{506,13},{507,13},{508,13},{516,13},{517,13},{518,13},{519,13},{520,13},{531,13},{532,13},{533,13},{534,13},{535,13},{536,13},{543,13},{544,13},{545,13},{546,13},{547,13},{548,13},{555,13},{556,13},{557,13},{558,13},{559,13},{560,13},{561,13},{562,13},{4,14},{5,14},{6,14},{7,14},{8,14},{9,14},{10,14},{11,14},{12,14},{13,14},{14,14},{15,14},{16,14},{17,14},{18,14},{19,14},{20,14},{30,14},{31,14},{32,14},{33,14},{34,14},{44,14},{45,14},{46,14},{47,14},{48,14},{55,14},{56,14},{57,14},{58,14},{59,14},{60,14},{61,14},{68,14},{69,14},{70,14},{71,14},{72,14},{73,14},{79,14},{80,14},{81,14},{82,14},{83,14},{92,14},{93,14},{94,14},{95,14},{96,14},{102,14},{103,14},{104,14},{105,14},{106,14},{115,14},{116,14},{117,14},{118,14},{119,14},{120,14},{126,14},{127,14},{128,14},{129,14},{130,14},{131,14},{132,14},{139,14},{140,14},{141,14},{142,14},{143,14},{144,14},{151,14},{152,14},{153,14},{154,14},{155,14},{163,14},{164,14},{165,14},{166,14},{167,14},{200,14},{201,14},{202,14},{203,14},{204,14},{211,14},{212,14},{213,14},{214,14},{215,14},{216,14},{217,14},{224,14},{225,14},{226,14},{227,14},{228,14},{229,14},{257,14},{258,14},{259,14},{260,14},{261,14},{283,14},{284,14},{285,14},{286,14},{287,14},{288,14},{289,14},{298,14},{299,14},{300,14},{301,14},{302,14},{303,14},{309,14},{310,14},{311,14},{312,14},{313,14},{314,14},{315,14},{322,14},{323,14},{324,14},{325,14},{326,14},{327,14},{328,14},{329,14},{336,14},{337,14},{338,14},{339,14},{340,14},{347,14},{348,14},{349,14},{350,14},{351,14},{352,14},{353,14},{361,14},{362,14},{363,14},{364,14},{365,14},{366,14},{384,14},{385,14},{386,14},{387,14},{388,14},{394,14},{395,14},{396,14},{397,14},{398,14},{399,14},{409,14},{410,14},{416,14},{417,14},{418,14},{419,14},{420,14},{431,14},{432,14},{433,14},{434,14},{435,14},{436,14},{443,14},{444,14},{445,14},{446,14},{447,14},{448,14},{449,14},{468,14},{469,14},{470,14},{471,14},{472,14},{479,14},{480,14},{481,14},{482,14},{483,14},{490,14},{491,14},{492,14},{493,14},{494,14},{495,14},{496,14},{503,14},{504,14},{505,14},{506,14},{507,14},{508,14},{516,14},{517,14},{518,14},{519,14},{520,14},{531,14},{532,14},{533,14},{534,14},{535,14},{544,14},{545,14},{546,14},{547,14},{548,14},{549,14},{555,14},{556,14},{557,14},{558,14},{559,14},{560,14},{561,14},{4,15},{5,15},{6,15},{7,15},{8,15},{9,15},{10,15},{11,15},{12,15},{13,15},{14,15},{15,15},{16,15},{17,15},{18,15},{19,15},{20,15},{21,15},{30,15},{31,15},{32,15},{33,15},{34,15},{44,15},{45,15},{46,15},{47,15},{48,15},{55,15},{56,15},{57,15},{58,15},{59,15},{60,15},{69,15},{70,15},{71,15},{72,15},{73,15},{79,15},{80,15},{81,15},{82,15},{83,15},{84,15},{91,15},{92,15},{93,15},{94,15},{95,15},{96,15},{101,15},{102,15},{103,15},{104,15},{105,15},{106,15},{116,15},{117,15},{118,15},{119,15},{120,15},{126,15},{127,15},{128,15},{129,15},{130,15},{131,15},{140,15},{141,15},{142,15},{143,15},{144,15},{151,15},{152,15},{153,15},{154,15},{155,15},{163,15},{164,15},{165,15},{166,15},{167,15},{200,15},{201,15},{202,15},{203,15},{204,15},{211,15},{212,15},{213,15},{214,15},{215,15},{216,15},{225,15},{226,15},{227,15},{228,15},{229,15},{257,15},{258,15},{259,15},{260,15},{261,15},{283,15},{284,15},{285,15},{286,15},{287,15},{288,15},{299,15},{300,15},{301,15},{302,15},{303,15},{309,15},{310,15},{311,15},{312,15},{313,15},{314,15},{322,15},{323,15},{324,15},{325,15},{326,15},{327,15},{328,15},{336,15},{337,15},{338,15},{339,15},{340,15},{347,15},{348,15},{349,15},{350,15},{351,15},{352,15},{361,15},{362,15},{363,15},{364,15},{365,15},{366,15},{384,15},{385,15},{386,15},{387,15},{388,15},{394,15},{395,15},{396,15},{397,15},{398,15},{399,15},{416,15},{417,15},{418,15},{419,15},{420,15},{431,15},{432,15},{433,15},{434,15},{435,15},{436,15},{437,15},{438,15},{439,15},{440,15},{441,15},{442,15},{443,15},{444,15},{445,15},{446,15},{447,15},{448,15},{468,15},{469,15},{470,15},{471,15},{472,15},{479,15},{480,15},{481,15},{482,15},{483,15},{490,15},{491,15},{492,15},{493,15},{494,15},{495,15},{504,15},{505,15},{506,15},{507,15},{508,15},{516,15},{517,15},{518,15},{519,15},{520,15},{530,15},{531,15},{532,15},{533,15},{534,15},{535,15},{545,15},{546,15},{547,15},{548,15},{549,15},{555,15},{556,15},{557,15},{558,15},{559,15},{560,15},{4,16},{5,16},{6,16},{7,16},{8,16},{9,16},{10,16},{11,16},{12,16},{13,16},{14,16},{15,16},{16,16},{17,16},{18,16},{19,16},{20,16},{21,16},{22,16},{30,16},{31,16},{32,16},{33,16},{34,16},{44,16},{45,16},{46,16},{47,16},{48,16},{55,16},{56,16},{57,16},{58,16},{59,16},{69,16},{70,16},{71,16},{72,16},{73,16},{79,16},{80,16},{81,16},{82,16},{83,16},{84,16},{91,16},{92,16},{93,16},{94,16},{95,16},{101,16},{102,16},{103,16},{104,16},{105,16},{116,16},{117,16},{118,16},{119,16},{120,16},{126,16},{127,16},{128,16},{129,16},{130,16},{140,16},{141,16},{142,16},{143,16},{144,16},{151,16},{152,16},{153,16},{154,16},{155,16},{163,16},{164,16},{165,16},{166,16},{167,16},{200,16},{201,16},{202,16},{203,16},{204,16},{211,16},{212,16},{213,16},{214,16},{215,16},{225,16},{226,16},{227,16},{228,16},{229,16},{257,16},{258,16},{259,16},{260,16},{261,16},{283,16},{284,16},{285,16},{286,16},{287,16},{299,16},{300,16},{301,16},{302,16},{303,16},{304,16},{309,16},{310,16},{311,16},{312,16},{313,16},{322,16},{323,16},{324,16},{325,16},{326,16},{327,16},{336,16},{337,16},{338,16},{339,16},{340,16},{341,16},{347,16},{348,16},{349,16},{350,16},{351,16},{362,16},{363,16},{364,16},{365,16},{366,16},{384,16},{385,16},{386,16},{387,16},{388,16},{394,16},{395,16},{396,16},{397,16},{398,16},{416,16},{417,16},{418,16},{419,16},{420,16},{431,16},{432,16},{433,16},{434,16},{435,16},{436,16},{437,16},{438,16},{439,16},{440,16},{441,16},{442,16},{443,16},{444,16},{445,16},{446,16},{447,16},{448,16},{468,16},{469,16},{470,16},{471,16},{472,16},{479,16},{480,16},{481,16},{482,16},{483,16},{490,16},{491,16},{492,16},{493,16},{494,16},{504,16},{505,16},{506,16},{507,16},{508,16},{516,16},{517,16},{518,16},{519,16},{520,16},{530,16},{531,16},{532,16},{533,16},{534,16},{545,16},{546,16},{547,16},{548,16},{549,16},{555,16},{556,16},{557,16},{558,16},{559,16},{4,17},{5,17},{6,17},{7,17},{8,17},{9,17},{16,17},{17,17},{18,17},{19,17},{20,17},{21,17},{22,17},{23,17},{30,17},{31,17},{32,17},{33,17},{34,17},{44,17},{45,17},{46,17},{47,17},{48,17},{55,17},{56,17},{57,17},{58,17},{59,17},{69,17},{70,17},{71,17},{72,17},{73,17},{80,17},{81,17},{82,17},{83,17},{84,17},{91,17},{92,17},{93,17},{94,17},{95,17},{101,17},{102,17},{103,17},{104,17},{105,17},{116,17},{117,17},{118,17},{119,17},{120,17},{126,17},{127,17},{128,17},{129,17},{130,17},{140,17},{141,17},{142,17},{143,17},{144,17},{151,17},{152,17},{153,17},{154,17},{155,17},{163,17},{164,17},{165,17},{166,17},{167,17},{200,17},{201,17},{202,17},{203,17},{204,17},{211,17},{212,17},{213,17},{214,17},{215,17},{225,17},{226,17},{227,17},{228,17},{229,17},{257,17},{258,17},{259,17},{260,17},{261,17},{283,17},{284,17},{285,17},{286,17},{287,17},{299,17},{300,17},{301,17},{302,17},{303,17},{304,17},{309,17},{310,17},{311,17},{312,17},{313,17},{323,17},{324,17},{325,17},{326,17},{327,17},{336,17},{337,17},{338,17},{339,17},{340,17},{341,17},{347,17},{348,17},{349,17},{350,17},{351,17},{362,17},{363,17},{364,17},{365,17},{366,17},{367,17},{384,17},{385,17},{386,17},{387,17},{388,17},{394,17},{395,17},{396,17},{397,17},{398,17},{416,17},{417,17},{418,17},{419,17},{420,17},{431,17},{432,17},{433,17},{434,17},{435,17},{436,17},{437,17},{438,17},{439,17},{440,17},{441,17},{442,17},{443,17},{444,17},{445,17},{446,17},{468,17},{469,17},{470,17},{471,17},{472,17},{479,17},{480,17},{481,17},{482,17},{483,17},{490,17},{491,17},{492,17},{493,17},{494,17},{504,17},{505,17},{506,17},{507,17},{508,17},{516,17},{517,17},{518,17},{519,17},{520,17},{530,17},{531,17},{532,17},{533,17},{534,17},{545,17},{546,17},{547,17},{548,17},{549,17},{555,17},{556,17},{557,17},{558,17},{559,17},{4,18},{5,18},{6,18},{7,18},{8,18},{9,18},{18,18},{19,18},{20,18},{21,18},{22,18},{23,18},{24,18},{30,18},{31,18},{32,18},{33,18},{34,18},{44,18},{45,18},{46,18},{47,18},{48,18},{55,18},{56,18},{57,18},{58,18},{59,18},{69,18},{70,18},{71,18},{72,18},{73,18},{74,18},{80,18},{81,18},{82,18},{83,18},{84,18},{85,18},{90,18},{91,18},{92,18},{93,18},{94,18},{95,18},{101,18},{102,18},{103,18},{104,18},{105,18},{106,18},{107,18},{108,18},{109,18},{110,18},{111,18},{112,18},{113,18},{114,18},{115,18},{116,18},{117,18},{118,18},{119,18},{120,18},{126,18},{127,18},{128,18},{129,18},{130,18},{140,18},{141,18},{142,18},{143,18},{144,18},{145,18},{151,18},{152,18},{153,18},{154,18},{155,18},{163,18},{164,18},{165,18},{166,18},{167,18},{200,18},{201,18},{202,18},{203,18},{204,18},{211,18},{212,18},{213,18},{214,18},{215,18},{225,18},{226,18},{227,18},{228,18},{229,18},{230,18},{257,18},{258,18},{259,18},{260,18},{261,18},{262,18},{282,18},{283,18},{284,18},{285,18},{286,18},{287,18},{299,18},{300,18},{301,18},{302,18},{303,18},{304,18},{309,18},{310,18},{311,18},{312,18},{313,18},{323,18},{324,18},{325,18},{326,18},{327,18},{336,18},{337,18},{338,18},{339,18},{340,18},{341,18},{347,18},{348,18},{349,18},{350,18},{351,18},{362,18},{363,18},{364,18},{365,18},{366,18},{367,18},{375,18},{376,18},{377,18},{378,18},{379,18},{380,18},{381,18},{382,18},{383,18},{384,18},{385,18},{386,18},{387,18},{388,18},{393,18},{394,18},{395,18},{396,18},{397,18},{398,18},{416,18},{417,18},{418,18},{419,18},{420,18},{431,18},{432,18},{433,18},{434,18},{435,18},{436,18},{437,18},{438,18},{439,18},{440,18},{441,18},{442,18},{443,18},{444,18},{459,18},{460,18},{461,18},{462,18},{463,18},{464,18},{465,18},{466,18},{467,18},{468,18},{469,18},{470,18},{471,18},{472,18},{479,18},{480,18},{481,18},{482,18},{483,18},{490,18},{491,18},{492,18},{493,18},{494,18},{504,18},{505,18},{506,18},{507,18},{508,18},{509,18},{516,18},{517,18},{518,18},{519,18},{520,18},{530,18},{531,18},{532,18},{533,18},{534,18},{535,18},{536,18},{537,18},{538,18},{539,18},{540,18},{541,18},{542,18},{543,18},{544,18},{545,18},{546,18},{547,18},{548,18},{549,18},{555,18},{556,18},{557,18},{558,18},{559,18},{4,19},{5,19},{6,19},{7,19},{8,19},{9,19},{19,19},{20,19},{21,19},{22,19},{23,19},{24,19},{30,19},{31,19},{32,19},{33,19},{34,19},{44,19},{45,19},{46,19},{47,19},{48,19},{55,19},{56,19},{57,19},{58,19},{59,19},{69,19},{70,19},{71,19},{72,19},{73,19},{74,19},{81,19},{82,19},{83,19},{84,19},{85,19},{90,19},{91,19},{92,19},{93,19},{94,19},{100,19},{101,19},{102,19},{103,19},{104,19},{105,19},{106,19},{107,19},{108,19},{109,19},{110,19},{111,19},{112,19},{113,19},{114,19},{115,19},{116,19},{117,19},{118,19},{119,19},{120,19},{126,19},{127,19},{128,19},{129,19},{130,19},{140,19},{141,19},{142,19},{143,19},{144,19},{145,19},{151,19},{152,19},{153,19},{154,19},{155,19},{163,19},{164,19},{165,19},{166,19},{167,19},{200,19},{201,19},{202,19},{203,19},{204,19},{211,19},{212,19},{213,19},{214,19},{215,19},{225,19},{226,19},{227,19},{228,19},{229,19},{230,19},{257,19},{258,19},{259,19},{260,19},{261,19},{262,19},{282,19},{283,19},{284,19},{285,19},{286,19},{287,19},{299,19},{300,19},{301,19},{302,19},{303,19},{304,19},{309,19},{310,19},{311,19},{312,19},{313,19},{323,19},{324,19},{325,19},{326,19},{327,19},{336,19},{337,19},{338,19},{339,19},{340,19},{341,19},{347,19},{348,19},{349,19},{350,19},{351,19},{362,19},{363,19},{364,19},{365,19},{366,19},{367,19},{373,19},{374,19},{375,19},{376,19},{377,19},{378,19},{379,19},{380,19},{381,19},{382,19},{383,19},{384,19},{385,19},{386,19},{387,19},{388,19},{393,19},{394,19},{395,19},{396,19},{397,19},{398,19},{416,19},{417,19},{418,19},{419,19},{420,19},{431,19},{432,19},{433,19},{434,19},{435,19},{436,19},{457,19},{458,19},{459,19},{460,19},{461,19},{462,19},{463,19},{464,19},{465,19},{466,19},{467,19},{468,19},{469,19},{470,19},{471,19},{472,19},{479,19},{480,19},{481,19},{482,19},{483,19},{490,19},{491,19},{492,19},{493,19},{494,19},{504,19},{505,19},{506,19},{507,19},{508,19},{509,19},{516,19},{517,19},{518,19},{519,19},{520,19},{529,19},{530,19},{531,19},{532,19},{533,19},{534,19},{535,19},{536,19},{537,19},{538,19},{539,19},{540,19},{541,19},{542,19},{543,19},{544,19},{545,19},{546,19},{547,19},{548,19},{549,19},{555,19},{556,19},{557,19},{558,19},{559,19},{4,20},{5,20},{6,20},{7,20},{8,20},{9,20},{20,20},{21,20},{22,20},{23,20},{24,20},{25,20},{30,20},{31,20},{32,20},{33,20},{34,20},{44,20},{45,20},{46,20},{47,20},{48,20},{55,20},{56,20},{57,20},{58,20},{59,20},{69,20},{70,20},{71,20},{72,20},{73,20},{74,20},{81,20},{82,20},{83,20},{84,20},{85,20},{89,20},{90,20},{91,20},{92,20},{93,20},{94,20},{100,20},{101,20},{102,20},{103,20},{104,20},{105,20},{106,20},{107,20},{108,20},{109,20},{110,20},{111,20},{112,20},{113,20},{114,20},{115,20},{116,20},{117,20},{118,20},{119,20},{120,20},{126,20},{127,20},{128,20},{129,20},{130,20},{140,20},{141,20},{142,20},{143,20},{144,20},{145,20},{151,20},{152,20},{153,20},{154,20},{155,20},{163,20},{164,20},{165,20},{166,20},{167,20},{200,20},{201,20},{202,20},{203,20},{204,20},{211,20},{212,20},{213,20},{214,20},{215,20},{225,20},{226,20},{227,20},{228,20},{229,20},{230,20},{257,20},{258,20},{259,20},{260,20},{261,20},{262,20},{282,20},{283,20},{284,20},{285,20},{286,20},{287,20},{299,20},{300,20},{301,20},{302,20},{303,20},{304,20},{309,20},{310,20},{311,20},{312,20},{313,20},{323,20},{324,20},{325,20},{326,20},{327,20},{336,20},{337,20},{338,20},{339,20},{340,20},{341,20},{347,20},{348,20},{349,20},{350,20},{351,20},{362,20},{363,20},{364,20},{365,20},{366,20},{367,20},{372,20},{373,20},{374,20},{375,20},{376,20},{377,20},{378,20},{379,20},{380,20},{381,20},{382,20},{383,20},{384,20},{385,20},{386,20},{387,20},{388,20},{393,20},{394,20},{395,20},{396,20},{397,20},{398,20},{416,20},{417,20},{418,20},{419,20},{420,20},{431,20},{432,20},{433,20},{434,20},{435,20},{436,20},{456,20},{457,20},{458,20},{459,20},{460,20},{461,20},{462,20},{463,20},{464,20},{465,20},{466,20},{467,20},{468,20},{469,20},{470,20},{471,20},{472,20},{479,20},{480,20},{481,20},{482,20},{483,20},{490,20},{491,20},{492,20},{493,20},{494,20},{504,20},{505,20},{506,20},{507,20},{508,20},{509,20},{516,20},{517,20},{518,20},{519,20},{520,20},{529,20},{530,20},{531,20},{532,20},{533,20},{534,20},{535,20},{536,20},{537,20},{538,20},{539,20},{540,20},{541,20},{542,20},{543,20},{544,20},{545,20},{546,20},{547,20},{548,20},{549,20},{555,20},{556,20},{557,20},{558,20},{559,20},{4,21},{5,21},{6,21},{7,21},{8,21},{9,21},{20,21},{21,21},{22,21},{23,21},{24,21},{25,21},{30,21},{31,21},{32,21},{33,21},{34,21},{44,21},{45,21},{46,21},{47,21},{48,21},{55,21},{56,21},{57,21},{58,21},{59,21},{69,21},{70,21},{71,21},{72,21},{73,21},{74,21},{81,21},{82,21},{83,21},{84,21},{85,21},{86,21},{89,21},{90,21},{91,21},{92,21},{93,21},{100,21},{101,21},{102,21},{103,21},{104,21},{105,21},{126,21},{127,21},{128,21},{129,21},{130,21},{140,21},{141,21},{142,21},{143,21},{144,21},{145,21},{151,21},{152,21},{153,21},{154,21},{155,21},{163,21},{164,21},{165,21},{166,21},{167,21},{200,21},{201,21},{202,21},{203,21},{204,21},{211,21},{212,21},{213,21},{214,21},{215,21},{225,21},{226,21},{227,21},{228,21},{229,21},{230,21},{257,21},{258,21},{259,21},{260,21},{261,21},{262,21},{282,21},{283,21},{284,21},{285,21},{286,21},{287,21},{299,21},{300,21},{301,21},{302,21},{303,21},{304,21},{309,21},{310,21},{311,21},{312,21},{313,21},{323,21},{324,21},{325,21},{326,21},{327,21},{336,21},{337,21},{338,21},{339,21},{340,21},{341,21},{347,21},{348,21},{349,21},{350,21},{351,21},{362,21},{363,21},{364,21},{365,21},{366,21},{367,21},{371,21},{372,21},{373,21},{374,21},{375,21},{376,21},{377,21},{378,21},{379,21},{384,21},{385,21},{386,21},{387,21},{388,21},{393,21},{394,21},{395,21},{396,21},{397,21},{398,21},{416,21},{417,21},{418,21},{419,21},{420,21},{431,21},{432,21},{433,21},{434,21},{435,21},{436,21},{455,21},{456,21},{457,21},{458,21},{459,21},{460,21},{461,21},{462,21},{463,21},{468,21},{469,21},{470,21},{471,21},{472,21},{479,21},{480,21},{481,21},{482,21},{483,21},{490,21},{491,21},{492,21},{493,21},{494,21},{504,21},{505,21},{506,21},{507,21},{508,21},{509,21},{516,21},{517,21},{518,21},{519,21},{520,21},{529,21},{530,21},{531,21},{532,21},{533,21},{534,21},{555,21},{556,21},{557,21},{558,21},{559,21},{4,22},{5,22},{6,22},{7,22},{8,22},{9,22},{20,22},{21,22},{22,22},{23,22},{24,22},{25,22},{30,22},{31,22},{32,22},{33,22},{34,22},{44,22},{45,22},{46,22},{47,22},{48,22},{55,22},{56,22},{57,22},{58,22},{59,22},{69,22},{70,22},{71,22},{72,22},{73,22},{74,22},{82,22},{83,22},{84,22},{85,22},{86,22},{89,22},{90,22},{91,22},{92,22},{93,22},{101,22},{102,22},{103,22},{104,22},{105,22},{126,22},{127,22},{128,22},{129,22},{130,22},{140,22},{141,22},{142,22},{143,22},{144,22},{145,22},{151,22},{152,22},{153,22},{154,22},{155,22},{163,22},{164,22},{165,22},{166,22},{167,22},{200,22},{201,22},{202,22},{203,22},{204,22},{211,22},{212,22},{213,22},{214,22},{215,22},{225,22},{226,22},{227,22},{228,22},{229,22},{230,22},{257,22},{258,22},{259,22},{260,22},{261,22},{262,22},{283,22},{284,22},{285,22},{286,22},{287,22},{299,22},{300,22},{301,22},{302,22},{303,22},{304,22},{309,22},{310,22},{311,22},{312,22},{313,22},{323,22},{324,22},{325,22},{326,22},{327,22},{336,22},{337,22},{338,22},{339,22},{340,22},{341,22},{347,22},{348,22},{349,22},{350,22},{351,22},{362,22},{363,22},{364,22},{365,22},{366,22},{367,22},{371,22},{372,22},{373,22},{374,22},{375,22},{376,22},{384,22},{385,22},{386,22},{387,22},{388,22},{393,22},{394,22},{395,22},{396,22},{397,22},{398,22},{416,22},{417,22},{418,22},{419,22},{420,22},{431,22},{432,22},{433,22},{434,22},{435,22},{436,22},{455,22},{456,22},{457,22},{458,22},{459,22},{460,22},{468,22},{469,22},{470,22},{471,22},{472,22},{479,22},{480,22},{481,22},{482,22},{483,22},{490,22},{491,22},{492,22},{493,22},{494,22},{504,22},{505,22},{506,22},{507,22},{508,22},{509,22},{516,22},{517,22},{518,22},{519,22},{520,22},{530,22},{531,22},{532,22},{533,22},{534,22},{555,22},{556,22},{557,22},{558,22},{559,22},{4,23},{5,23},{6,23},{7,23},{8,23},{9,23},{20,23},{21,23},{22,23},{23,23},{24,23},{25,23},{30,23},{31,23},{32,23},{33,23},{34,23},{44,23},{45,23},{46,23},{47,23},{48,23},{55,23},{56,23},{57,23},{58,23},{59,23},{69,23},{70,23},{71,23},{72,23},{73,23},{74,23},{82,23},{83,23},{84,23},{85,23},{86,23},{87,23},{88,23},{89,23},{90,23},{91,23},{92,23},{93,23},{101,23},{102,23},{103,23},{104,23},{105,23},{126,23},{127,23},{128,23},{129,23},{130,23},{140,23},{141,23},{142,23},{143,23},{144,23},{145,23},{151,23},{152,23},{153,23},{154,23},{155,23},{163,23},{164,23},{165,23},{166,23},{167,23},{200,23},{201,23},{202,23},{203,23},{204,23},{211,23},{212,23},{213,23},{214,23},{215,23},{225,23},{226,23},{227,23},{228,23},{229,23},{230,23},{258,23},{259,23},{260,23},{261,23},{262,23},{263,23},{283,23},{284,23},{285,23},{286,23},{287,23},{299,23},{300,23},{301,23},{302,23},{303,23},{304,23},{309,23},{310,23},{311,23},{312,23},{313,23},{323,23},{324,23},{325,23},{326,23},{327,23},{336,23},{337,23},{338,23},{339,23},{340,23},{341,23},{347,23},{348,23},{349,23},{350,23},{351,23},{362,23},{363,23},{364,23},{365,23},{366,23},{371,23},{372,23},{373,23},{374,23},{375,23},{384,23},{385,23},{386,23},{387,23},{388,23},{394,23},{395,23},{396,23},{397,23},{398,23},{416,23},{417,23},{418,23},{419,23},{420,23},{431,23},{432,23},{433,23},{434,23},{435,23},{436,23},{455,23},{456,23},{457,23},{458,23},{459,23},{468,23},{469,23},{470,23},{471,23},{472,23},{479,23},{480,23},{481,23},{482,23},{483,23},{490,23},{491,23},{492,23},{493,23},{494,23},{504,23},{505,23},{506,23},{507,23},{508,23},{509,23},{516,23},{517,23},{518,23},{519,23},{520,23},{530,23},{531,23},{532,23},{533,23},{534,23},{555,23},{556,23},{557,23},{558,23},{559,23},{4,24},{5,24},{6,24},{7,24},{8,24},{9,24},{20,24},{21,24},{22,24},{23,24},{24,24},{25,24},{30,24},{31,24},{32,24},{33,24},{34,24},{44,24},{45,24},{46,24},{47,24},{48,24},{55,24},{56,24},{57,24},{58,24},{59,24},{69,24},{70,24},{71,24},{72,24},{73,24},{74,24},{82,24},{83,24},{84,24},{85,24},{86,24},{87,24},{88,24},{89,24},{90,24},{91,24},{92,24},{101,24},{102,24},{103,24},{104,24},{105,24},{106,24},{126,24},{127,24},{128,24},{129,24},{130,24},{140,24},{141,24},{142,24},{143,24},{144,24},{145,24},{151,24},{152,24},{153,24},{154,24},{155,24},{163,24},{164,24},{165,24},{166,24},{167,24},{200,24},{201,24},{202,24},{203,24},{204,24},{211,24},{212,24},{213,24},{214,24},{215,24},{225,24},{226,24},{227,24},{228,24},{229,24},{230,24},{258,24},{259,24},{260,24},{261,24},{262,24},{263,24},{264,24},{278,24},{279,24},{283,24},{284,24},{285,24},{286,24},{287,24},{288,24},{298,24},{299,24},{300,24},{301,24},{302,24},{303,24},{309,24},{310,24},{311,24},{312,24},{313,24},{323,24},{324,24},{325,24},{326,24},{327,24},{336,24},{337,24},{338,24},{339,24},{340,24},{341,24},{347,24},{348,24},{349,24},{350,24},{351,24},{352,24},{361,24},{362,24},{363,24},{364,24},{365,24},{366,24},{370,24},{371,24},{372,24},{373,24},{374,24},{375,24},{384,24},{385,24},{386,24},{387,24},{388,24},{394,24},{395,24},{396,24},{397,24},{398,24},{399,24},{409,24},{410,24},{416,24},{417,24},{418,24},{419,24},{420,24},{431,24},{432,24},{433,24},{434,24},{435,24},{436,24},{454,24},{455,24},{456,24},{457,24},{458,24},{459,24},{468,24},{469,24},{470,24},{471,24},{472,24},{479,24},{480,24},{481,24},{482,24},{483,24},{490,24},{491,24},{492,24},{493,24},{494,24},{504,24},{505,24},{506,24},{507,24},{508,24},{509,24},{516,24},{517,24},{518,24},{519,24},{520,24},{530,24},{531,24},{532,24},{533,24},{534,24},{535,24},{555,24},{556,24},{557,24},{558,24},{559,24},{4,25},{5,25},{6,25},{7,25},{8,25},{9,25},{19,25},{20,25},{21,25},{22,25},{23,25},{24,25},{30,25},{31,25},{32,25},{33,25},{34,25},{35,25},{42,25},{43,25},{44,25},{45,25},{46,25},{47,25},{48,25},{55,25},{56,25},{57,25},{58,25},{59,25},{69,25},{70,25},{71,25},{72,25},{73,25},{74,25},{83,25},{84,25},{85,25},{86,25},{87,25},{88,25},{89,25},{90,25},{91,25},{92,25},{101,25},{102,25},{103,25},{104,25},{105,25},{106,25},{126,25},{127,25},{128,25},{129,25},{130,25},{140,25},{141,25},{142,25},{143,25},{144,25},{145,25},{151,25},{152,25},{153,25},{154,25},{155,25},{163,25},{164,25},{165,25},{166,25},{167,25},{200,25},{201,25},{202,25},{203,25},{204,25},{211,25},{212,25},{213,25},{214,25},{215,25},{225,25},{226,25},{227,25},{228,25},{229,25},{230,25},{259,25},{260,25},{261,25},{262,25},{263,25},{264,25},{265,25},{276,25},{277,25},{278,25},{279,25},{283,25},{284,25},{285,25},{286,25},{287,25},{288,25},{298,25},{299,25},{300,25},{301,25},{302,25},{303,25},{309,25},{310,25},{311,25},{312,25},{313,25},{323,25},{324,25},{325,25},{326,25},{327,25},{336,25},{337,25},{338,25},{339,25},{340,25},{341,25},{347,25},{348,25},{349,25},{350,25},{351,25},{352,25},{353,25},{361,25},{362,25},{363,25},{364,25},{365,25},{366,25},{370,25},{371,25},{372,25},{373,25},{374,25},{375,25},{383,25},{384,25},{385,25},{386,25},{387,25},{388,25},{394,25},{395,25},{396,25},{397,25},{398,25},{399,25},{408,25},{409,25},{410,25},{416,25},{417,25},{418,25},{419,25},{420,25},{431,25},{432,25},{433,25},{434,25},{435,25},{436,25},{454,25},{455,25},{456,25},{457,25},{458,25},{459,25},{467,25},{468,25},{469,25},{470,25},{471,25},{472,25},{479,25},{480,25},{481,25},{482,25},{483,25},{490,25},{491,25},{492,25},{493,25},{494,25},{504,25},{505,25},{506,25},{507,25},{508,25},{509,25},{516,25},{517,25},{518,25},{519,25},{520,25},{530,25},{531,25},{532,25},{533,25},{534,25},{535,25},{555,25},{556,25},{557,25},{558,25},{559,25},{4,26},{5,26},{6,26},{7,26},{8,26},{9,26},{18,26},{19,26},{20,26},{21,26},{22,26},{23,26},{24,26},{30,26},{31,26},{32,26},{33,26},{34,26},{35,26},{36,26},{41,26},{42,26},{43,26},{44,26},{45,26},{46,26},{47,26},{48,26},{55,26},{56,26},{57,26},{58,26},{59,26},{69,26},{70,26},{71,26},{72,26},{73,26},{74,26},{83,26},{84,26},{85,26},{86,26},{87,26},{88,26},{89,26},{90,26},{91,26},{92,26},{102,26},{103,26},{104,26},{105,26},{106,26},{107,26},{116,26},{117,26},{118,26},{119,26},{126,26},{127,26},{128,26},{129,26},{130,26},{140,26},{141,26},{142,26},{143,26},{144,26},{145,26},{151,26},{152,26},{153,26},{154,26},{155,26},{163,26},{164,26},{165,26},{166,26},{167,26},{168,26},{200,26},{201,26},{202,26},{203,26},{204,26},{211,26},{212,26},{213,26},{214,26},{215,26},{225,26},{226,26},{227,26},{228,26},{229,26},{230,26},{259,26},{260,26},{261,26},{262,26},{263,26},{264,26},{265,26},{266,26},{274,26},{275,26},{276,26},{277,26},{278,26},{279,26},{284,26},{285,26},{286,26},{287,26},{288,26},{289,26},{290,26},{296,26},{297,26},{298,26},{299,26},{300,26},{301,26},{302,26},{309,26},{310,26},{311,26},{312,26},{313,26},{323,26},{324,26},{325,26},{326,26},{327,26},{336,26},{337,26},{338,26},{339,26},{340,26},{341,26},{347,26},{348,26},{349,26},{350,26},{351,26},{352,26},{353,26},{354,26},{360,26},{361,26},{362,26},{363,26},{364,26},{365,26},{371,26},{372,26},{373,26},{374,26},{375,26},{382,26},{383,26},{384,26},{385,26},{386,26},{387,26},{388,26},{395,26},{396,26},{397,26},{398,26},{399,26},{400,26},{406,26},{407,26},{408,26},{409,26},{410,26},{416,26},{417,26},{418,26},{419,26},{420,26},{421,26},{431,26},{432,26},{433,26},{434,26},{435,26},{436,26},{455,26},{456,26},{457,26},{458,26},{459,26},{466,26},{467,26},{468,26},{469,26},{470,26},{471,26},{472,26},{479,26},{480,26},{481,26},{482,26},{483,26},{490,26},{491,26},{492,26},{493,26},{494,26},{504,26},{505,26},{506,26},{507,26},{508,26},{509,26},{516,26},{517,26},{518,26},{519,26},{520,26},{521,26},{531,26},{532,26},{533,26},{534,26},{535,26},{536,26},{545,26},{546,26},{547,26},{548,26},{555,26},{556,26},{557,26},{558,26},{559,26},{4,27},{5,27},{6,27},{7,27},{8,27},{9,27},{10,27},{11,27},{12,27},{13,27},{14,27},{15,27},{16,27},{17,27},{18,27},{19,27},{20,27},{21,27},{22,27},{23,27},{31,27},{32,27},{33,27},{34,27},{35,27},{36,27},{37,27},{38,27},{39,27},{40,27},{41,27},{42,27},{43,27},{44,27},{45,27},{46,27},{47,27},{48,27},{55,27},{56,27},{57,27},{58,27},{59,27},{69,27},{70,27},{71,27},{72,27},{73,27},{74,27},{83,27},{84,27},{85,27},{86,27},{87,27},{88,27},{89,27},{90,27},{91,27},{102,27},{103,27},{104,27},{105,27},{106,27},{107,27},{108,27},{109,27},{110,27},{111,27},{112,27},{113,27},{114,27},{115,27},{116,27},{117,27},{118,27},{119,27},{126,27},{127,27},{128,27},{129,27},{130,27},{140,27},{141,27},{142,27},{143,27},{144,27},{145,27},{151,27},{152,27},{153,27},{154,27},{155,27},{163,27},{164,27},{165,27},{166,27},{167,27},{168,27},{169,27},{170,27},{171,27},{172,27},{173,27},{200,27},{201,27},{202,27},{203,27},{204,27},{211,27},{212,27},{213,27},{214,27},{215,27},{225,27},{226,27},{227,27},{228,27},{229,27},{230,27},{260,27},{261,27},{262,27},{263,27},{264,27},{265,27},{266,27},{267,27},{268,27},{269,27},{270,27},{271,27},{272,27},{273,27},{274,27},{275,27},{276,27},{277,27},{278,27},{279,27},{284,27},{285,27},{286,27},{287,27},{288,27},{289,27},{290,27},{291,27},{292,27},{293,27},{294,27},{295,27},{296,27},{297,27},{298,27},{299,27},{300,27},{301,27},{302,27},{309,27},{310,27},{311,27},{312,27},{313,27},{323,27},{324,27},{325,27},{326,27},{327,27},{336,27},{337,27},{338,27},{339,27},{340,27},{341,27},{347,27},{348,27},{349,27},{350,27},{351,27},{352,27},{353,27},{354,27},{355,27},{356,27},{357,27},{358,27},{359,27},{360,27},{361,27},{362,27},{363,27},{364,27},{365,27},{371,27},{372,27},{373,27},{374,27},{375,27},{376,27},{377,27},{380,27},{381,27},{382,27},{383,27},{384,27},{385,27},{386,27},{387,27},{388,27},{395,27},{396,27},{397,27},{398,27},{399,27},{400,27},{401,27},{402,27},{403,27},{404,27},{405,27},{406,27},{407,27},{408,27},{409,27},{410,27},{416,27},{417,27},{418,27},{419,27},{420,27},{421,27},{422,27},{423,27},{424,27},{425,27},{426,27},{431,27},{432,27},{433,27},{434,27},{435,27},{436,27},{455,27},{456,27},{457,27},{458,27},{459,27},{460,27},{461,27},{464,27},{465,27},{466,27},{467,27},{468,27},{469,27},{470,27},{471,27},{472,27},{479,27},{480,27},{481,27},{482,27},{483,27},{490,27},{491,27},{492,27},{493,27},{494,27},{504,27},{505,27},{506,27},{507,27},{508,27},{509,27},{516,27},{517,27},{518,27},{519,27},{520,27},{521,27},{522,27},{523,27},{524,27},{525,27},{526,27},{531,27},{532,27},{533,27},{534,27},{535,27},{536,27},{537,27},{538,27},{539,27},{540,27},{541,27},{542,27},{543,27},{544,27},{545,27},{546,27},{547,27},{548,27},{555,27},{556,27},{557,27},{558,27},{559,27},{4,28},{5,28},{6,28},{7,28},{8,28},{9,28},{10,28},{11,28},{12,28},{13,28},{14,28},{15,28},{16,28},{17,28},{18,28},{19,28},{20,28},{21,28},{22,28},{31,28},{32,28},{33,28},{34,28},{35,28},{36,28},{37,28},{38,28},{39,28},{40,28},{41,28},{42,28},{43,28},{44,28},{45,28},{46,28},{47,28},{48,28},{55,28},{56,28},{57,28},{58,28},{59,28},{69,28},{70,28},{71,28},{72,28},{73,28},{74,28},{84,28},{85,28},{86,28},{87,28},{88,28},{89,28},{90,28},{91,28},{103,28},{104,28},{105,28},{106,28},{107,28},{108,28},{109,28},{110,28},{111,28},{112,28},{113,28},{114,28},{115,28},{116,28},{117,28},{118,28},{119,28},{126,28},{127,28},{128,28},{129,28},{130,28},{140,28},{141,28},{142,28},{143,28},{144,28},{145,28},{151,28},{152,28},{153,28},{154,28},{155,28},{164,28},{165,28},{166,28},{167,28},{168,28},{169,28},{170,28},{171,28},{172,28},{173,28},{200,28},{201,28},{202,28},{203,28},{204,28},{211,28},{212,28},{213,28},{214,28},{215,28},{225,28},{226,28},{227,28},{228,28},{229,28},{230,28},{261,28},{262,28},{263,28},{264,28},{265,28},{266,28},{267,28},{268,28},{269,28},{270,28},{271,28},{272,28},{273,28},{274,28},{275,28},{276,28},{277,28},{278,28},{285,28},{286,28},{287,28},{288,28},{289,28},{290,28},{291,28},{292,28},{293,28},{294,28},{295,28},{296,28},{297,28},{298,28},{299,28},{300,28},{301,28},{309,28},{310,28},{311,28},{312,28},{313,28},{323,28},{324,28},{325,28},{326,28},{327,28},{336,28},{337,28},{338,28},{339,28},{340,28},{341,28},{347,28},{348,28},{349,28},{350,28},{351,28},{352,28},{353,28},{354,28},{355,28},{356,28},{357,28},{358,28},{359,28},{360,28},{361,28},{362,28},{363,28},{364,28},{372,28},{373,28},{374,28},{375,28},{376,28},{377,28},{378,28},{379,28},{380,28},{381,28},{382,28},{383,28},{384,28},{385,28},{386,28},{387,28},{388,28},{396,28},{397,28},{398,28},{399,28},{400,28},{401,28},{402,28},{403,28},{404,28},{405,28},{406,28},{407,28},{408,28},{409,28},{410,28},{417,28},{418,28},{419,28},{420,28},{421,28},{422,28},{423,28},{424,28},{425,28},{426,28},{431,28},{432,28},{433,28},{434,28},{435,28},{436,28},{456,28},{457,28},{458,28},{459,28},{460,28},{461,28},{462,28},{463,28},{464,28},{465,28},{466,28},{467,28},{468,28},{469,28},{470,28},{471,28},{472,28},{479,28},{480,28},{481,28},{482,28},{483,28},{490,28},{491,28},{492,28},{493,28},{494,28},{504,28},{505,28},{506,28},{507,28},{508,28},{509,28},{517,28},{518,28},{519,28},{520,28},{521,28},{522,28},{523,28},{524,28},{525,28},{526,28},{532,28},{533,28},{534,28},{535,28},{536,28},{537,28},{538,28},{539,28},{540,28},{541,28},{542,28},{543,28},{544,28},{545,28},{546,28},{547,28},{548,28},{555,28},{556,28},{557,28},{558,28},{559,28},{4,29},{5,29},{6,29},{7,29},{8,29},{9,29},{10,29},{11,29},{12,29},{13,29},{14,29},{15,29},{16,29},{17,29},{18,29},{19,29},{20,29},{21,29},{32,29},{33,29},{34,29},{35,29},{36,29},{37,29},{38,29},{39,29},{40,29},{41,29},{42,29},{43,29},{44,29},{45,29},{46,29},{47,29},{48,29},{55,29},{56,29},{57,29},{58,29},{59,29},{69,29},{70,29},{71,29},{72,29},{73,29},{74,29},{84,29},{85,29},{86,29},{87,29},{88,29},{89,29},{90,29},{104,29},{105,29},{106,29},{107,29},{108,29},{109,29},{110,29},{111,29},{112,29},{113,29},{114,29},{115,29},{116,29},{117,29},{118,29},{126,29},{127,29},{128,29},{129,29},{130,29},{140,29},{141,29},{142,29},{143,29},{144,29},{145,29},{151,29},{152,29},{153,29},{154,29},{155,29},{164,29},{165,29},{166,29},{167,29},{168,29},{169,29},{170,29},{171,29},{172,29},{173,29},{200,29},{201,29},{202,29},{203,29},{204,29},{211,29},{212,29},{213,29},{214,29},{215,29},{225,29},{226,29},{227,29},{228,29},{229,29},{230,29},{263,29},{264,29},{265,29},{266,29},{267,29},{268,29},{269,29},{270,29},{271,29},{272,29},{273,29},{274,29},{275,29},{276,29},{277,29},{286,29},{287,29},{288,29},{289,29},{290,29},{291,29},{292,29},{293,29},{294,29},{295,29},{296,29},{297,29},{298,29},{299,29},{309,29},{310,29},{311,29},{312,29},{313,29},{323,29},{324,29},{325,29},{326,29},{327,29},{336,29},{337,29},{338,29},{339,29},{340,29},{341,29},{347,29},{348,29},{349,29},{350,29},{351,29},{352,29},{353,29},{354,29},{355,29},{356,29},{357,29},{358,29},{359,29},{360,29},{361,29},{362,29},{363,29},{372,29},{373,29},{374,29},{375,29},{376,29},{377,29},{378,29},{379,29},{380,29},{381,29},{382,29},{383,29},{384,29},{385,29},{386,29},{387,29},{388,29},{397,29},{398,29},{399,29},{400,29},{401,29},{402,29},{403,29},{404,29},{405,29},{406,29},{407,29},{408,29},{409,29},{417,29},{418,29},{419,29},{420,29},{421,29},{422,29},{423,29},{424,29},{425,29},{426,29},{431,29},{432,29},{433,29},{434,29},{435,29},{436,29},{456,29},{457,29},{458,29},{459,29},{460,29},{461,29},{462,29},{463,29},{464,29},{465,29},{466,29},{467,29},{468,29},{469,29},{470,29},{471,29},{472,29},{479,29},{480,29},{481,29},{482,29},{483,29},{490,29},{491,29},{492,29},{493,29},{494,29},{504,29},{505,29},{506,29},{507,29},{508,29},{509,29},{517,29},{518,29},{519,29},{520,29},{521,29},{522,29},{523,29},{524,29},{525,29},{526,29},{533,29},{534,29},{535,29},{536,29},{537,29},{538,29},{539,29},{540,29},{541,29},{542,29},{543,29},{544,29},{545,29},{546,29},{547,29},{555,29},{556,29},{557,29},{558,29},{559,29},{5,30},{6,30},{7,30},{8,30},{9,30},{10,30},{11,30},{12,30},{13,30},{14,30},{15,30},{16,30},{17,30},{18,30},{19,30},{34,30},{35,30},{36,30},{37,30},{38,30},{39,30},{40,30},{41,30},{44,30},{45,30},{46,30},{47,30},{48,30},{55,30},{56,30},{57,30},{58,30},{59,30},{69,30},{70,30},{71,30},{72,30},{73,30},{85,30},{86,30},{87,30},{88,30},{89,30},{90,30},{106,30},{107,30},{108,30},{109,30},{110,30},{111,30},{112,30},{113,30},{114,30},{115,30},{116,30},{126,30},{127,30},{128,30},{129,30},{130,30},{140,30},{141,30},{142,30},{143,30},{144,30},{151,30},{152,30},{153,30},{154,30},{155,30},{165,30},{166,30},{167,30},{168,30},{169,30},{170,30},{171,30},{172,30},{173,30},{200,30},{201,30},{202,30},{203,30},{204,30},{211,30},{212,30},{213,30},{214,30},{215,30},{225,30},{226,30},{227,30},{228,30},{229,30},{265,30},{266,30},{267,30},{268,30},{269,30},{270,30},{271,30},{272,30},{273,30},{274,30},{288,30},{289,30},{290,30},{291,30},{292,30},{293,30},{294,30},{295,30},{296,30},{297,30},{298,30},{309,30},{310,30},{311,30},{312,30},{313,30},{323,30},{324,30},{325,30},{326,30},{327,30},{336,30},{337,30},{338,30},{339,30},{340,30},{341,30},{347,30},{348,30},{349,30},{350,30},{351,30},{354,30},{355,30},{356,30},{357,30},{358,30},{359,30},{360,30},{361,30},{374,30},{375,30},{376,30},{377,30},{378,30},{379,30},{380,30},{381,30},{382,30},{384,30},{385,30},{386,30},{387,30},{388,30},{399,30},{400,30},{401,30},{402,30},{403,30},{404,30},{405,30},{406,30},{407,30},{418,30},{419,30},{420,30},{421,30},{422,30},{423,30},{424,30},{425,30},{426,30},{431,30},{432,30},{433,30},{434,30},{435,30},{436,30},{458,30},{459,30},{460,30},{461,30},{462,30},{463,30},{464,30},{465,30},{466,30},{468,30},{469,30},{470,30},{471,30},{472,30},{479,30},{480,30},{481,30},{482,30},{483,30},{490,30},{491,30},{492,30},{493,30},{494,30},{504,30},{505,30},{506,30},{507,30},{508,30},{518,30},{519,30},{520,30},{521,30},{522,30},{523,30},{524,30},{525,30},{526,30},{535,30},{536,30},{537,30},{538,30},{539,30},{540,30},{541,30},{542,30},{543,30},{544,30},{545,30},{555,30},{556,30},{557,30},{558,30},{559,30},{347,31},{348,31},{349,31},{350,31},{351,31},{347,32},{348,32},{349,32},{350,32},{351,32},{347,33},{348,33},{349,33},{350,33},{351,33},{347,34},{348,34},{349,34},{350,34},{351,34},{347,35},{348,35},{349,35},{350,35},{351,35},{347,36},{348,36},{349,36},{350,36},{351,36},{347,37},{348,37},{349,37},{350,37},{351,37},{347,38},{348,38},{349,38},{350,38},{351,38},{225,70},{226,70},{227,70},{228,70},{225,71},{226,71},{227,71},{228,71},{225,72},{226,72},{227,72},{228,72},{284,72},{285,72},{286,72},{287,72},{225,73},{226,73},{227,73},{228,73},{284,73},{285,73},{286,73},{287,73},{288,73},{225,74},{226,74},{227,74},{228,74},{275,74},{276,74},{277,74},{278,74},{284,74},{285,74},{286,74},{287,74},{321,74},{322,74},{323,74},{324,74},{225,75},{226,75},{227,75},{228,75},{275,75},{276,75},{277,75},{278,75},{321,75},{322,75},{323,75},{324,75},{225,76},{226,76},{227,76},{228,76},{275,76},{276,76},{277,76},{278,76},{321,76},{322,76},{323,76},{324,76},{212,77},{213,77},{214,77},{215,77},{216,77},{217,77},{218,77},{219,77},{220,77},{225,77},{226,77},{227,77},{228,77},{235,77},{236,77},{237,77},{238,77},{239,77},{240,77},{241,77},{248,77},{249,77},{250,77},{251,77},{252,77},{253,77},{254,77},{255,77},{256,77},{257,77},{258,77},{263,77},{264,77},{265,77},{266,77},{267,77},{268,77},{269,77},{273,77},{274,77},{275,77},{276,77},{277,77},{278,77},{279,77},{280,77},{281,77},{282,77},{284,77},{285,77},{286,77},{287,77},{301,77},{302,77},{303,77},{304,77},{305,77},{306,77},{307,77},{308,77},{319,77},{320,77},{321,77},{322,77},{323,77},{324,77},{325,77},{326,77},{327,77},{328,77},{333,77},{334,77},{335,77},{336,77},{337,77},{338,77},{339,77},{344,77},{345,77},{346,77},{347,77},{348,77},{349,77},{350,77},{351,77},{352,77},{353,77},{354,77},{357,77},{358,77},{359,77},{360,77},{361,77},{362,77},{368,77},{369,77},{370,77},{371,77},{372,77},{373,77},{374,77},{375,77},{376,77},{211,78},{212,78},{213,78},{214,78},{215,78},{216,78},{217,78},{218,78},{219,78},{220,78},{221,78},{225,78},{226,78},{227,78},{228,78},{233,78},{234,78},{235,78},{236,78},{237,78},{238,78},{239,78},{240,78},{241,78},{242,78},{247,78},{248,78},{249,78},{250,78},{251,78},{252,78},{253,78},{254,78},{255,78},{256,78},{257,78},{261,78},{262,78},{263,78},{264,78},{265,78},{266,78},{267,78},{268,78},{269,78},{270,78},{273,78},{274,78},{275,78},{276,78},{277,78},{278,78},{279,78},{280,78},{281,78},{284,78},{285,78},{286,78},{287,78},{299,78},{300,78},{301,78},{302,78},{303,78},{304,78},{305,78},{306,78},{307,78},{308,78},{309,78},{319,78},{320,78},{321,78},{322,78},{323,78},{324,78},{325,78},{326,78},{327,78},{331,78},{332,78},{333,78},{334,78},{335,78},{336,78},{337,78},{338,78},{339,78},{340,78},{344,78},{345,78},{346,78},{347,78},{348,78},{349,78},{350,78},{351,78},{352,78},{353,78},{354,78},{355,78},{356,78},{357,78},{358,78},{359,78},{360,78},{361,78},{362,78},{363,78},{367,78},{368,78},{369,78},{370,78},{371,78},{372,78},{373,78},{374,78},{375,78},{376,78},{377,78},{211,79},{212,79},{213,79},{214,79},{218,79},{219,79},{220,79},{221,79},{225,79},{226,79},{227,79},{228,79},{233,79},{234,79},{235,79},{236,79},{240,79},{241,79},{242,79},{243,79},{246,79},{247,79},{248,79},{249,79},{253,79},{254,79},{255,79},{256,79},{261,79},{262,79},{263,79},{264,79},{268,79},{269,79},{270,79},{271,79},{275,79},{276,79},{277,79},{278,79},{284,79},{285,79},{286,79},{287,79},{299,79},{300,79},{301,79},{302,79},{303,79},{306,79},{307,79},{308,79},{309,79},{310,79},{321,79},{322,79},{323,79},{324,79},{331,79},{332,79},{333,79},{334,79},{338,79},{339,79},{340,79},{341,79},{344,79},{345,79},{346,79},{347,79},{348,79},{349,79},{351,79},{352,79},{353,79},{354,79},{355,79},{356,79},{357,79},{358,79},{360,79},{361,79},{362,79},{363,79},{367,79},{368,79},{369,79},{370,79},{374,79},{375,79},{376,79},{377,79},{219,80},{220,80},{221,80},{225,80},{226,80},{227,80},{228,80},{232,80},{233,80},{234,80},{235,80},{241,80},{242,80},{243,80},{246,80},{247,80},{248,80},{249,80},{253,80},{254,80},{255,80},{256,80},{260,80},{261,80},{262,80},{263,80},{269,80},{270,80},{271,80},{275,80},{276,80},{277,80},{278,80},{284,80},{285,80},{286,80},{287,80},{298,80},{299,80},{300,80},{301,80},{307,80},{308,80},{309,80},{310,80},{321,80},{322,80},{323,80},{324,80},{330,80},{331,80},{332,80},{333,80},{339,80},{340,80},{341,80},{344,80},{345,80},{346,80},{347,80},{348,80},{352,80},{353,80},{354,80},{355,80},{356,80},{361,80},{362,80},{363,80},{364,80},{375,80},{376,80},{377,80},{219,81},{220,81},{221,81},{222,81},{225,81},{226,81},{227,81},{228,81},{232,81},{233,81},{234,81},{235,81},{241,81},{242,81},{243,81},{244,81},{246,81},{247,81},{248,81},{249,81},{253,81},{254,81},{255,81},{256,81},{260,81},{261,81},{262,81},{263,81},{269,81},{270,81},{271,81},{272,81},{275,81},{276,81},{277,81},{278,81},{284,81},{285,81},{286,81},{287,81},{298,81},{299,81},{300,81},{301,81},{308,81},{309,81},{310,81},{311,81},{321,81},{322,81},{323,81},{324,81},{330,81},{331,81},{332,81},{333,81},{339,81},{340,81},{341,81},{342,81},{344,81},{345,81},{346,81},{347,81},{353,81},{354,81},{355,81},{356,81},{361,81},{362,81},{363,81},{364,81},{375,81},{376,81},{377,81},{378,81},{213,82},{214,82},{215,82},{216,82},{217,82},{218,82},{219,82},{220,82},{221,82},{222,82},{225,82},{226,82},{227,82},{228,82},{232,82},{233,82},{234,82},{235,82},{236,82},{237,82},{238,82},{239,82},{240,82},{241,82},{242,82},{243,82},{244,82},{246,82},{247,82},{248,82},{249,82},{253,82},{254,82},{255,82},{256,82},{260,82},{261,82},{262,82},{263,82},{264,82},{265,82},{266,82},{267,82},{268,82},{269,82},{270,82},{271,82},{272,82},{275,82},{276,82},{277,82},{278,82},{284,82},{285,82},{286,82},{287,82},{298,82},{299,82},{300,82},{301,82},{308,82},{309,82},{310,82},{311,82},{321,82},{322,82},{323,82},{324,82},{330,82},{331,82},{332,82},{333,82},{334,82},{335,82},{336,82},{337,82},{338,82},{339,82},{340,82},{341,82},{342,82},{344,82},{345,82},{346,82},{347,82},{353,82},{354,82},{355,82},{356,82},{361,82},{362,82},{363,82},{364,82},{369,82},{370,82},{371,82},{372,82},{373,82},{374,82},{375,82},{376,82},{377,82},{378,82},{212,83},{213,83},{214,83},{215,83},{216,83},{217,83},{218,83},{219,83},{220,83},{221,83},{222,83},{225,83},{226,83},{227,83},{228,83},{232,83},{233,83},{234,83},{235,83},{236,83},{237,83},{238,83},{239,83},{240,83},{241,83},{242,83},{243,83},{244,83},{247,83},{248,83},{249,83},{250,83},{251,83},{252,83},{253,83},{254,83},{255,83},{260,83},{261,83},{262,83},{263,83},{264,83},{265,83},{266,83},{267,83},{268,83},{269,83},{270,83},{271,83},{272,83},{275,83},{276,83},{277,83},{278,83},{284,83},{285,83},{286,83},{287,83},{298,83},{299,83},{300,83},{308,83},{309,83},{310,83},{311,83},{321,83},{322,83},{323,83},{324,83},{330,83},{331,83},{332,83},{333,83},{334,83},{335,83},{336,83},{337,83},{338,83},{339,83},{340,83},{341,83},{342,83},{344,83},{345,83},{346,83},{347,83},{353,83},{354,83},{355,83},{356,83},{361,83},{362,83},{363,83},{364,83},{368,83},{369,83},{370,83},{371,83},{372,83},{373,83},{374,83},{375,83},{376,83},{377,83},{378,83},{211,84},{212,84},{213,84},{214,84},{215,84},{219,84},{220,84},{221,84},{222,84},{225,84},{226,84},{227,84},{228,84},{232,84},{233,84},{234,84},{235,84},{246,84},{247,84},{248,84},{249,84},{250,84},{251,84},{252,84},{253,84},{254,84},{260,84},{261,84},{262,84},{263,84},{275,84},{276,84},{277,84},{278,84},{284,84},{285,84},{286,84},{287,84},{298,84},{299,84},{300,84},{301,84},{308,84},{309,84},{310,84},{311,84},{321,84},{322,84},{323,84},{324,84},{330,84},{331,84},{332,84},{333,84},{344,84},{345,84},{346,84},{347,84},{353,84},{354,84},{355,84},{356,84},{361,84},{362,84},{363,84},{364,84},{367,84},{368,84},{369,84},{370,84},{371,84},{375,84},{376,84},{377,84},{378,84},{211,85},{212,85},{213,85},{214,85},{219,85},{220,85},{221,85},{222,85},{225,85},{226,85},{227,85},{228,85},{232,85},{233,85},{234,85},{235,85},{246,85},{247,85},{248,85},{260,85},{261,85},{262,85},{263,85},{275,85},{276,85},{277,85},{278,85},{284,85},{285,85},{286,85},{287,85},{298,85},{299,85},{300,85},{301,85},{308,85},{309,85},{310,85},{311,85},{321,85},{322,85},{323,85},{324,85},{330,85},{331,85},{332,85},{333,85},{344,85},{345,85},{346,85},{347,85},{353,85},{354,85},{355,85},{356,85},{361,85},{362,85},{363,85},{364,85},{367,85},{368,85},{369,85},{370,85},{375,85},{376,85},{377,85},{378,85},{211,86},{212,86},{213,86},{219,86},{220,86},{221,86},{222,86},{225,86},{226,86},{227,86},{228,86},{232,86},{233,86},{234,86},{235,86},{245,86},{246,86},{247,86},{248,86},{249,86},{260,86},{261,86},{262,86},{263,86},{275,86},{276,86},{277,86},{278,86},{284,86},{285,86},{286,86},{287,86},{298,86},{299,86},{300,86},{301,86},{307,86},{308,86},{309,86},{310,86},{321,86},{322,86},{323,86},{324,86},{330,86},{331,86},{332,86},{333,86},{344,86},{345,86},{346,86},{347,86},{353,86},{354,86},{355,86},{356,86},{361,86},{362,86},{363,86},{364,86},{367,86},{368,86},{369,86},{375,86},{376,86},{377,86},{378,86},{211,87},{212,87},{213,87},{214,87},{217,87},{218,87},{219,87},{220,87},{221,87},{222,87},{225,87},{226,87},{227,87},{228,87},{233,87},{234,87},{235,87},{236,87},{241,87},{242,87},{243,87},{246,87},{247,87},{248,87},{249,87},{250,87},{251,87},{252,87},{253,87},{254,87},{255,87},{256,87},{261,87},{262,87},{263,87},{264,87},{269,87},{270,87},{271,87},{275,87},{276,87},{277,87},{278,87},{280,87},{281,87},{284,87},{285,87},{286,87},{287,87},{298,87},{299,87},{300,87},{301,87},{302,87},{303,87},{306,87},{307,87},{308,87},{309,87},{310,87},{321,87},{322,87},{323,87},{324,87},{326,87},{327,87},{331,87},{332,87},{333,87},{334,87},{339,87},{340,87},{341,87},{344,87},{345,87},{346,87},{347,87},{353,87},{354,87},{355,87},{356,87},{361,87},{362,87},{363,87},{364,87},{367,87},{368,87},{369,87},{370,87},{373,87},{374,87},{375,87},{376,87},{377,87},{378,87},{211,88},{212,88},{213,88},{214,88},{215,88},{216,88},{217,88},{218,88},{219,88},{220,88},{221,88},{222,88},{225,88},{226,88},{227,88},{228,88},{233,88},{234,88},{235,88},{236,88},{237,88},{238,88},{239,88},{240,88},{241,88},{242,88},{243,88},{246,88},{247,88},{248,88},{249,88},{250,88},{251,88},{252,88},{253,88},{254,88},{255,88},{256,88},{257,88},{261,88},{262,88},{263,88},{264,88},{265,88},{266,88},{267,88},{268,88},{269,88},{270,88},{271,88},{275,88},{276,88},{277,88},{278,88},{279,88},{280,88},{281,88},{282,88},{284,88},{285,88},{286,88},{287,88},{299,88},{300,88},{301,88},{302,88},{303,88},{304,88},{305,88},{306,88},{307,88},{308,88},{309,88},{321,88},{322,88},{323,88},{324,88},{325,88},{326,88},{327,88},{328,88},{331,88},{332,88},{333,88},{334,88},{335,88},{336,88},{337,88},{338,88},{339,88},{340,88},{341,88},{344,88},{345,88},{346,88},{347,88},{353,88},{354,88},{355,88},{356,88},{361,88},{362,88},{363,88},{364,88},{367,88},{368,88},{369,88},{370,88},{371,88},{372,88},{373,88},{374,88},{375,88},{376,88},{377,88},{378,88},{212,89},{213,89},{214,89},{215,89},{216,89},{217,89},{218,89},{219,89},{220,89},{221,89},{222,89},{225,89},{226,89},{227,89},{228,89},{234,89},{235,89},{236,89},{237,89},{238,89},{239,89},{240,89},{241,89},{242,89},{246,89},{247,89},{248,89},{249,89},{254,89},{255,89},{256,89},{257,89},{262,89},{263,89},{264,89},{265,89},{266,89},{267,89},{268,89},{269,89},{270,89},{276,89},{277,89},{278,89},{279,89},{280,89},{281,89},{284,89},{285,89},{286,89},{287,89},{300,89},{301,89},{302,89},{303,89},{304,89},{305,89},{306,89},{307,89},{308,89},{322,89},{323,89},{324,89},{325,89},{326,89},{327,89},{332,89},{333,89},{334,89},{335,89},{336,89},{337,89},{338,89},{339,89},{340,89},{344,89},{345,89},{346,89},{347,89},{353,89},{354,89},{355,89},{356,89},{361,89},{362,89},{363,89},{364,89},{368,89},{369,89},{370,89},{371,89},{372,89},{373,89},{374,89},{375,89},{376,89},{377,89},{378,89},{245,90},{246,90},{247,90},{248,90},{255,90},{256,90},{257,90},{245,91},{246,91},{247,91},{248,91},{254,91},{255,91},{256,91},{257,91},{245,92},{246,92},{247,92},{248,92},{249,92},{253,92},{254,92},{255,92},{256,92},{257,92},{246,93},{247,93},{248,93},{249,93},{250,93},{251,93},{252,93},{253,93},{254,93},{255,93},{256,93},{247,94},{248,94},{249,94},{250,94},{251,94},{252,94},{253,94},{254,94},{255,94},

};

int continuati[1134][2]{

{8,1},{9,1},{10,1},{11,1},{12,1},{13,1},{14,1},{15,1},{16,1},{55,1},{56,1},{57,1},{61,1},{62,1},{63,1},{64,1},{114,1},{115,1},{116,1},{120,1},{121,1},{122,1},{123,1},{6,2},{7,2},{8,2},{9,2},{10,2},{11,2},{12,2},{13,2},{14,2},{15,2},{16,2},{17,2},{18,2},{54,2},{55,2},{56,2},{57,2},{61,2},{62,2},{63,2},{64,2},{113,2},{114,2},{115,2},{116,2},{120,2},{121,2},{122,2},{123,2},{5,3},{6,3},{7,3},{8,3},{9,3},{10,3},{14,3},{15,3},{16,3},{17,3},{18,3},{19,3},{54,3},{55,3},{56,3},{57,3},{113,3},{114,3},{115,3},{116,3},{5,4},{6,4},{7,4},{8,4},{16,4},{17,4},{18,4},{19,4},{54,4},{55,4},{56,4},{57,4},{113,4},{114,4},{115,4},{116,4},{4,5},{5,5},{6,5},{7,5},{17,5},{18,5},{19,5},{20,5},{54,5},{55,5},{56,5},{57,5},{113,5},{114,5},{115,5},{116,5},{3,6},{4,6},{5,6},{6,6},{17,6},{18,6},{19,6},{20,6},{26,6},{27,6},{28,6},{29,6},{30,6},{31,6},{32,6},{38,6},{39,6},{40,6},{41,6},{42,6},{43,6},{44,6},{45,6},{46,6},{47,6},{48,6},{52,6},{53,6},{54,6},{55,6},{56,6},{57,6},{58,6},{59,6},{61,6},{62,6},{63,6},{64,6},{67,6},{68,6},{69,6},{70,6},{71,6},{72,6},{73,6},{74,6},{75,6},{76,6},{77,6},{82,6},{83,6},{84,6},{85,6},{91,6},{92,6},{93,6},{94,6},{99,6},{100,6},{101,6},{102,6},{103,6},{104,6},{105,6},{106,6},{107,6},{111,6},{112,6},{113,6},{114,6},{115,6},{116,6},{117,6},{118,6},{120,6},{121,6},{122,6},{123,6},{3,7},{4,7},{5,7},{6,7},{24,7},{25,7},{26,7},{27,7},{28,7},{29,7},{30,7},{31,7},{32,7},{33,7},{34,7},{38,7},{39,7},{40,7},{41,7},{42,7},{43,7},{44,7},{45,7},{46,7},{47,7},{48,7},{49,7},{52,7},{53,7},{54,7},{55,7},{56,7},{57,7},{58,7},{59,7},{61,7},{62,7},{63,7},{64,7},{67,7},{68,7},{69,7},{70,7},{71,7},{72,7},{73,7},{74,7},{75,7},{76,7},{77,7},{78,7},{82,7},{83,7},{84,7},{85,7},{91,7},{92,7},{93,7},{94,7},{98,7},{99,7},{100,7},{101,7},{102,7},{103,7},{104,7},{105,7},{106,7},{107,7},{108,7},{111,7},{112,7},{113,7},{114,7},{115,7},{116,7},{117,7},{118,7},{120,7},{121,7},{122,7},{123,7},{3,8},{4,8},{5,8},{6,8},{24,8},{25,8},{26,8},{27,8},{28,8},{30,8},{31,8},{32,8},{33,8},{34,8},{38,8},{39,8},{40,8},{41,8},{42,8},{43,8},{45,8},{46,8},{47,8},{48,8},{49,8},{54,8},{55,8},{56,8},{57,8},{61,8},{62,8},{63,8},{64,8},{67,8},{68,8},{69,8},{70,8},{71,8},{72,8},{74,8},{75,8},{76,8},{77,8},{78,8},{82,8},{83,8},{84,8},{85,8},{91,8},{92,8},{93,8},{94,8},{97,8},{98,8},{99,8},{100,8},{101,8},{104,8},{105,8},{106,8},{107,8},{108,8},{109,8},{113,8},{114,8},{115,8},{116,8},{120,8},{121,8},{122,8},{123,8},{3,9},{4,9},{5,9},{6,9},{23,9},{24,9},{25,9},{26,9},{32,9},{33,9},{34,9},{35,9},{38,9},{39,9},{40,9},{41,9},{42,9},{47,9},{48,9},{49,9},{50,9},{54,9},{55,9},{56,9},{57,9},{61,9},{62,9},{63,9},{64,9},{67,9},{68,9},{69,9},{70,9},{71,9},{76,9},{77,9},{78,9},{79,9},{82,9},{83,9},{84,9},{85,9},{91,9},{92,9},{93,9},{94,9},{97,9},{98,9},{99,9},{100,9},{106,9},{107,9},{108,9},{109,9},{113,9},{114,9},{115,9},{116,9},{120,9},{121,9},{122,9},{123,9},{3,10},{4,10},{5,10},{6,10},{23,10},{24,10},{25,10},{26,10},{32,10},{33,10},{34,10},{35,10},{38,10},{39,10},{40,10},{41,10},{47,10},{48,10},{49,10},{50,10},{54,10},{55,10},{56,10},{57,10},{61,10},{62,10},{63,10},{64,10},{67,10},{68,10},{69,10},{70,10},{76,10},{77,10},{78,10},{79,10},{82,10},{83,10},{84,10},{85,10},{91,10},{92,10},{93,10},{94,10},{106,10},{107,10},{108,10},{109,10},{113,10},{114,10},{115,10},{116,10},{120,10},{121,10},{122,10},{123,10},{3,11},{4,11},{5,11},{6,11},{22,11},{23,11},{24,11},{25,11},{33,11},{34,11},{35,11},{36,11},{38,11},{39,11},{40,11},{41,11},{47,11},{48,11},{49,11},{50,11},{54,11},{55,11},{56,11},{57,11},{61,11},{62,11},{63,11},{64,11},{67,11},{68,11},{69,11},{70,11},{76,11},{77,11},{78,11},{79,11},{82,11},{83,11},{84,11},{85,11},{91,11},{92,11},{93,11},{94,11},{102,11},{103,11},{104,11},{105,11},{106,11},{107,11},{108,11},{109,11},{113,11},{114,11},{115,11},{116,11},{120,11},{121,11},{122,11},{123,11},{3,12},{4,12},{5,12},{6,12},{22,12},{23,12},{24,12},{25,12},{33,12},{34,12},{35,12},{36,12},{38,12},{39,12},{40,12},{41,12},{47,12},{48,12},{49,12},{50,12},{54,12},{55,12},{56,12},{57,12},{61,12},{62,12},{63,12},{64,12},{67,12},{68,12},{69,12},{70,12},{76,12},{77,12},{78,12},{79,12},{82,12},{83,12},{84,12},{85,12},{91,12},{92,12},{93,12},{94,12},{99,12},{100,12},{101,12},{102,12},{103,12},{104,12},{105,12},{106,12},{107,12},{108,12},{109,12},{113,12},{114,12},{115,12},{116,12},{120,12},{121,12},{122,12},{123,12},{3,13},{4,13},{5,13},{6,13},{22,13},{23,13},{24,13},{25,13},{33,13},{34,13},{35,13},{36,13},{38,13},{39,13},{40,13},{41,13},{47,13},{48,13},{49,13},{50,13},{54,13},{55,13},{56,13},{57,13},{61,13},{62,13},{63,13},{64,13},{67,13},{68,13},{69,13},{70,13},{76,13},{77,13},{78,13},{79,13},{82,13},{83,13},{84,13},{85,13},{91,13},{92,13},{93,13},{94,13},{98,13},{99,13},{100,13},{101,13},{102,13},{103,13},{104,13},{105,13},{106,13},{107,13},{108,13},{109,13},{113,13},{114,13},{115,13},{116,13},{120,13},{121,13},{122,13},{123,13},{3,14},{4,14},{5,14},{6,14},{18,14},{19,14},{20,14},{21,14},{22,14},{23,14},{24,14},{25,14},{33,14},{34,14},{35,14},{36,14},{38,14},{39,14},{40,14},{41,14},{47,14},{48,14},{49,14},{50,14},{54,14},{55,14},{56,14},{57,14},{61,14},{62,14},{63,14},{64,14},{67,14},{68,14},{69,14},{70,14},{76,14},{77,14},{78,14},{79,14},{82,14},{83,14},{84,14},{85,14},{91,14},{92,14},{93,14},{94,14},{97,14},{98,14},{99,14},{100,14},{101,14},{106,14},{107,14},{108,14},{109,14},{113,14},{114,14},{115,14},{116,14},{120,14},{121,14},{122,14},{123,14},{4,15},{5,15},{6,15},{17,15},{18,15},{19,15},{20,15},{22,15},{23,15},{24,15},{25,15},{33,15},{34,15},{35,15},{36,15},{38,15},{39,15},{40,15},{41,15},{47,15},{48,15},{49,15},{50,15},{54,15},{55,15},{56,15},{57,15},{61,15},{62,15},{63,15},{64,15},{67,15},{68,15},{69,15},{70,15},{76,15},{77,15},{78,15},{79,15},{82,15},{83,15},{84,15},{85,15},{91,15},{92,15},{93,15},{94,15},{96,15},{97,15},{98,15},{99,15},{106,15},{107,15},{108,15},{109,15},{113,15},{114,15},{115,15},{116,15},{120,15},{121,15},{122,15},{123,15},{4,16},{5,16},{6,16},{7,16},{17,16},{18,16},{19,16},{20,16},{23,16},{24,16},{25,16},{26,16},{32,16},{33,16},{34,16},{35,16},{38,16},{39,16},{40,16},{41,16},{47,16},{48,16},{49,16},{50,16},{54,16},{55,16},{56,16},{57,16},{61,16},{62,16},{63,16},{64,16},{67,16},{68,16},{69,16},{70,16},{76,16},{77,16},{78,16},{79,16},{82,16},{83,16},{84,16},{85,16},{91,16},{92,16},{93,16},{94,16},{96,16},{97,16},{98,16},{99,16},{105,16},{106,16},{107,16},{108,16},{109,16},{113,16},{114,16},{115,16},{116,16},{120,16},{121,16},{122,16},{123,16},{5,17},{6,17},{7,17},{8,17},{16,17},{17,17},{18,17},{19,17},{23,17},{24,17},{25,17},{26,17},{32,17},{33,17},{34,17},{35,17},{38,17},{39,17},{40,17},{41,17},{47,17},{48,17},{49,17},{50,17},{54,17},{55,17},{56,17},{57,17},{61,17},{62,17},{63,17},{64,17},{67,17},{68,17},{69,17},{70,17},{76,17},{77,17},{78,17},{79,17},{82,17},{83,17},{84,17},{85,17},{90,17},{91,17},{92,17},{93,17},{94,17},{96,17},{97,17},{98,17},{99,17},{105,17},{106,17},{107,17},{108,17},{109,17},{113,17},{114,17},{115,17},{116,17},{120,17},{121,17},{122,17},{123,17},{5,18},{6,18},{7,18},{8,18},{9,18},{10,18},{14,18},{15,18},{16,18},{17,18},{18,18},{19,18},{24,18},{25,18},{26,18},{27,18},{28,18},{30,18},{31,18},{32,18},{33,18},{34,18},{38,18},{39,18},{40,18},{41,18},{47,18},{48,18},{49,18},{50,18},{54,18},{55,18},{56,18},{57,18},{61,18},{62,18},{63,18},{64,18},{67,18},{68,18},{69,18},{70,18},{76,18},{77,18},{78,18},{79,18},{83,18},{84,18},{85,18},{86,18},{87,18},{89,18},{90,18},{91,18},{92,18},{93,18},{94,18},{97,18},{98,18},{99,18},{100,18},{101,18},{103,18},{104,18},{105,18},{106,18},{107,18},{108,18},{109,18},{113,18},{114,18},{115,18},{116,18},{120,18},{121,18},{122,18},{123,18},{6,19},{7,19},{8,19},{9,19},{10,19},{11,19},{12,19},{13,19},{14,19},{15,19},{16,19},{17,19},{18,19},{24,19},{25,19},{26,19},{27,19},{28,19},{29,19},{30,19},{31,19},{32,19},{33,19},{34,19},{38,19},{39,19},{40,19},{41,19},{47,19},{48,19},{49,19},{50,19},{54,19},{55,19},{56,19},{57,19},{58,19},{59,19},{61,19},{62,19},{63,19},{64,19},{67,19},{68,19},{69,19},{70,19},{76,19},{77,19},{78,19},{79,19},{83,19},{84,19},{85,19},{86,19},{87,19},{88,19},{89,19},{90,19},{91,19},{92,19},{93,19},{94,19},{97,19},{98,19},{99,19},{100,19},{101,19},{102,19},{103,19},{104,19},{105,19},{106,19},{107,19},{108,19},{109,19},{113,19},{114,19},{115,19},{116,19},{117,19},{118,19},{120,19},{121,19},{122,19},{123,19},{8,20},{9,20},{10,20},{11,20},{12,20},{13,20},{14,20},{15,20},{16,20},{26,20},{27,20},{28,20},{29,20},{30,20},{31,20},{32,20},{38,20},{39,20},{40,20},{41,20},{47,20},{48,20},{49,20},{50,20},{55,20},{56,20},{57,20},{58,20},{59,20},{61,20},{62,20},{63,20},{64,20},{67,20},{68,20},{69,20},{70,20},{76,20},{77,20},{78,20},{79,20},{84,20},{85,20},{86,20},{87,20},{88,20},{89,20},{90,20},{91,20},{92,20},{93,20},{94,20},{98,20},{99,20},{100,20},{101,20},{102,20},{103,20},{104,20},{105,20},{106,20},{107,20},{108,20},{109,20},{114,20},{115,20},{116,20},{117,20},{118,20},{120,20},{121,20},{122,20},{123,20},

};

int marime[336][2]{

{30,0},{31,0},{32,0},{3,3},{4,3},{5,3},{6,3},{7,3},{8,3},{9,3},{10,3},{11,3},{12,3},{13,3},{14,3},{18,3},{19,3},{20,3},{21,3},{22,3},{25,3},{26,3},{27,3},{28,3},{29,3},{30,3},{31,3},{32,3},{34,3},{35,3},{36,3},{37,3},{38,3},{39,3},{40,3},{41,3},{42,3},{43,3},{44,3},{45,3},{49,3},{50,3},{51,3},{52,3},{53,3},{3,4},{4,4},{5,4},{6,4},{8,4},{9,4},{10,4},{11,4},{13,4},{14,4},{15,4},{17,4},{18,4},{21,4},{22,4},{23,4},{25,4},{26,4},{27,4},{28,4},{30,4},{31,4},{32,4},{34,4},{35,4},{36,4},{37,4},{39,4},{40,4},{41,4},{42,4},{44,4},{45,4},{46,4},{48,4},{49,4},{50,4},{52,4},{53,4},{54,4},{3,5},{4,5},{5,5},{8,5},{9,5},{10,5},{13,5},{14,5},{15,5},{16,5},{17,5},{18,5},{21,5},{22,5},{23,5},{25,5},{26,5},{27,5},{30,5},{31,5},{32,5},{34,5},{35,5},{36,5},{39,5},{40,5},{41,5},{44,5},{45,5},{46,5},{47,5},{48,5},{49,5},{53,5},{54,5},{55,5},{3,6},{4,6},{5,6},{8,6},{9,6},{10,6},{13,6},{14,6},{15,6},{20,6},{21,6},{22,6},{23,6},{25,6},{26,6},{27,6},{30,6},{31,6},{32,6},{34,6},{35,6},{36,6},{39,6},{40,6},{41,6},{44,6},{45,6},{46,6},{47,6},{48,6},{49,6},{53,6},{54,6},{55,6},{3,7},{4,7},{5,7},{8,7},{9,7},{10,7},{13,7},{14,7},{15,7},{17,7},{18,7},{19,7},{20,7},{21,7},{22,7},{23,7},{25,7},{26,7},{27,7},{30,7},{31,7},{32,7},{34,7},{35,7},{36,7},{39,7},{40,7},{41,7},{44,7},{45,7},{46,7},{47,7},{48,7},{49,7},{50,7},{51,7},{52,7},{53,7},{54,7},{55,7},{3,8},{4,8},{5,8},{8,8},{9,8},{10,8},{13,8},{14,8},{15,8},{16,8},{17,8},{18,8},{21,8},{22,8},{23,8},{25,8},{26,8},{27,8},{30,8},{31,8},{32,8},{34,8},{35,8},{36,8},{39,8},{40,8},{41,8},{44,8},{45,8},{46,8},{47,8},{48,8},{49,8},{3,9},{4,9},{5,9},{8,9},{9,9},{10,9},{13,9},{14,9},{15,9},{16,9},{17,9},{18,9},{21,9},{22,9},{23,9},{25,9},{26,9},{27,9},{30,9},{31,9},{32,9},{34,9},{35,9},{36,9},{39,9},{40,9},{41,9},{44,9},{45,9},{46,9},{47,9},{48,9},{49,9},{3,10},{4,10},{5,10},{8,10},{9,10},{10,10},{13,10},{14,10},{15,10},{16,10},{17,10},{18,10},{20,10},{21,10},{22,10},{23,10},{25,10},{26,10},{27,10},{30,10},{31,10},{32,10},{34,10},{35,10},{36,10},{39,10},{40,10},{41,10},{44,10},{45,10},{46,10},{48,10},{49,10},{50,10},{53,10},{54,10},{55,10},{3,11},{4,11},{5,11},{8,11},{9,11},{10,11},{13,11},{14,11},{15,11},{17,11},{18,11},{19,11},{20,11},{21,11},{22,11},{23,11},{24,11},{25,11},{26,11},{27,11},{30,11},{31,11},{32,11},{34,11},{35,11},{36,11},{39,11},{40,11},{41,11},{44,11},{45,11},{46,11},{49,11},{50,11},{51,11},{52,11},{53,11},{54,11},

};

int orientare[411][2]{

{6,3},{7,3},{8,3},{9,3},{10,3},{11,3},{20,3},{21,3},{22,3},{42,3},{43,3},{5,4},{6,4},{7,4},{10,4},{11,4},{12,4},{41,4},{42,4},{43,4},{4,5},{5,5},{6,5},{11,5},{12,5},{13,5},{41,5},{42,5},{43,5},{4,6},{5,6},{12,6},{13,6},{14,6},{15,6},{16,6},{17,6},{18,6},{19,6},{20,6},{21,6},{22,6},{26,6},{27,6},{28,6},{29,6},{30,6},{33,6},{34,6},{35,6},{36,6},{37,6},{38,6},{39,6},{40,6},{41,6},{42,6},{43,6},{44,6},{45,6},{47,6},{48,6},{49,6},{50,6},{51,6},{54,6},{55,6},{56,6},{57,6},{58,6},{59,6},{61,6},{62,6},{63,6},{64,6},{65,6},{3,7},{4,7},{5,7},{12,7},{13,7},{14,7},{15,7},{16,7},{17,7},{18,7},{20,7},{21,7},{22,7},{25,7},{26,7},{27,7},{29,7},{30,7},{31,7},{33,7},{34,7},{35,7},{36,7},{38,7},{39,7},{40,7},{41,7},{42,7},{43,7},{46,7},{47,7},{50,7},{51,7},{52,7},{54,7},{55,7},{56,7},{57,7},{60,7},{61,7},{62,7},{64,7},{65,7},{66,7},{3,8},{4,8},{5,8},{12,8},{13,8},{14,8},{15,8},{16,8},{17,8},{20,8},{21,8},{22,8},{24,8},{25,8},{26,8},{30,8},{31,8},{32,8},{33,8},{34,8},{35,8},{38,8},{39,8},{40,8},{41,8},{42,8},{43,8},{45,8},{46,8},{47,8},{50,8},{51,8},{52,8},{54,8},{55,8},{56,8},{59,8},{60,8},{61,8},{65,8},{66,8},{67,8},{3,9},{4,9},{5,9},{12,9},{13,9},{14,9},{15,9},{16,9},{17,9},{20,9},{21,9},{22,9},{24,9},{25,9},{26,9},{30,9},{31,9},{32,9},{33,9},{34,9},{35,9},{38,9},{39,9},{40,9},{41,9},{42,9},{43,9},{49,9},{50,9},{51,9},{52,9},{54,9},{55,9},{56,9},{59,9},{60,9},{61,9},{65,9},{66,9},{67,9},{3,10},{4,10},{5,10},{12,10},{13,10},{14,10},{15,10},{16,10},{17,10},{20,10},{21,10},{22,10},{24,10},{25,10},{26,10},{27,10},{28,10},{29,10},{30,10},{31,10},{32,10},{33,10},{34,10},{35,10},{38,10},{39,10},{40,10},{41,10},{42,10},{43,10},{46,10},{47,10},{48,10},{49,10},{50,10},{51,10},{52,10},{54,10},{55,10},{56,10},{59,10},{60,10},{61,10},{62,10},{63,10},{64,10},{65,10},{66,10},{67,10},{4,11},{5,11},{12,11},{13,11},{14,11},{15,11},{16,11},{17,11},{20,11},{21,11},{22,11},{24,11},{25,11},{26,11},{33,11},{34,11},{35,11},{38,11},{39,11},{40,11},{41,11},{42,11},{43,11},{45,11},{46,11},{47,11},{50,11},{51,11},{52,11},{54,11},{55,11},{56,11},{59,11},{60,11},{61,11},{4,12},{5,12},{6,12},{11,12},{12,12},{13,12},{15,12},{16,12},{17,12},{20,12},{21,12},{22,12},{24,12},{25,12},{26,12},{33,12},{34,12},{35,12},{38,12},{39,12},{40,12},{41,12},{42,12},{43,12},{45,12},{46,12},{47,12},{50,12},{51,12},{52,12},{54,12},{55,12},{56,12},{59,12},{60,12},{61,12},{5,13},{6,13},{7,13},{10,13},{11,13},{12,13},{15,13},{16,13},{17,13},{20,13},{21,13},{22,13},{25,13},{26,13},{27,13},{30,13},{31,13},{32,13},{33,13},{34,13},{35,13},{38,13},{39,13},{40,13},{41,13},{42,13},{43,13},{45,13},{46,13},{47,13},{49,13},{50,13},{51,13},{52,13},{54,13},{55,13},{56,13},{60,13},{61,13},{62,13},{65,13},{66,13},{67,13},{6,14},{7,14},{8,14},{9,14},{10,14},{11,14},{15,14},{16,14},{17,14},{20,14},{21,14},{22,14},{26,14},{27,14},{28,14},{29,14},{30,14},{31,14},{33,14},{34,14},{35,14},{38,14},{39,14},{40,14},{42,14},{43,14},{44,14},{45,14},{46,14},{47,14},{48,14},{49,14},{50,14},{51,14},{52,14},{53,14},{54,14},{55,14},{56,14},{61,14},{62,14},{63,14},{64,14},{65,14},{66,14},

};

bool limit(int x, int y) {

return (x >= 85 && x <= 1285 && y >= 75 && y <= 725) ? 1 : 0;

}

void \_brush(int x, int y, COLORREF color) {

for (int d = 0; d < 360; d++) {

for (int r = 0; r <= paint\_brush.size; r++) {

if (limit(x + (r \* cos(d \* pi / 180)), y + (r \* sin(d \* pi / 180))))

SetPixel(consoleDC, x + (r \* cos(d \* pi / 180)), y + (r \* sin(d \* pi / 180)), color);

}

}

}

void \_eraser(int x, int y) {

for (int d = 0; d < 360; d++) {

for (int r = 0; r <= eraser.size; r++) {

if (limit(x + (r \* cos(d \* pi / 180)), y + (r \* sin(d \* pi / 180))))

SetPixel(consoleDC, x + (r \* cos(d \* pi / 180)), y + (r \* sin(d \* pi / 180)), RGB(255, 255, 255));

}

}

}

int \_bucket(int f\_x, int f\_y, COLORREF start\_color, BOOL LIMIT) {

if (LIMIT) {

int bx[] = { 0,0,1,-1,1,-1,1,-1 };

int by[] = { 1,-1,0,0,-1,1,1,-1 };

int d[] = { 90,180,270,360 };

COLORREF color = GetPixel(consoleDC, f\_x, f\_y);

if (color == start\_color)

return 0;

vector<pair<int, int>> v;

v.push\_back(make\_pair(f\_x, f\_y));

SetPixel(consoleDC, f\_x, f\_y, start\_color);

int p = 0;

while (p < v.size()) {

for (int i = 0; i < 4; i++) {

if (GetPixel(consoleDC, v[p].first + bx[4 + i], v[p].second + by[4 + i]) == color && limit(v[p].first + bx[4 + i], v[p].second + by[4 + i]))

v.push\_back(make\_pair(v[p].first + bx[4 + i], v[p].second + by[4 + i])), SetPixel(consoleDC, v[p].first + bx[4 + i], v[p].second + by[4 + i], start\_color);

}

p++;

}

p = 0;

while (p < v.size()) {

for (int i = 0; i < 4; i++) {

if (GetPixel(consoleDC, v[p].first + bx[i], v[p].second + by[i]) == color)

SetPixel(consoleDC, v[p].first + bx[i], v[p].second + by[i], start\_color);

}

p++;

}

return 0;

}

else {

int bx[] = { 0,0,1,-1,1,-1,1,-1 };

int by[] = { 1,-1,0,0,-1,1,1,-1 };

int d[] = { 90,180,270,360 };

COLORREF color = GetPixel(consoleDC, f\_x, f\_y);

if (color == start\_color)

return 0;

vector<pair<int, int>> v;

v.push\_back(make\_pair(f\_x, f\_y));

SetPixel(consoleDC, f\_x, f\_y, start\_color);

int p = 0;

while (p < v.size()) {

for (int i = 0; i < 4; i++) {

if (GetPixel(consoleDC, v[p].first + bx[4 + i], v[p].second + by[4 + i]) == color)

v.push\_back(make\_pair(v[p].first + bx[4 + i], v[p].second + by[4 + i])), SetPixel(consoleDC, v[p].first + bx[4 + i], v[p].second + by[4 + i], start\_color);

}

p++;

}

p = 0;

while (p < v.size()) {

for (int i = 0; i < 4; i++) {

if (GetPixel(consoleDC, v[p].first + bx[i], v[p].second + by[i]) == color)

SetPixel(consoleDC, v[p].first + bx[i], v[p].second + by[i], start\_color);

}

p++;

}

return 0;

}

}

void \_pen(int x, int y, COLORREF color) {

for (int i = -pen.size; i <= pen.size; i++) {

SetPixel(consoleDC, x + (i \* cos(pen.orientation \* pi / 180)), y + (i \* sin(pen.orientation \* pi / 180)), color);

}

}

void \_line(double x1, double y1, double x2, double y2, COLORREF color) {

double d = sqrt((((x2 - x1) \* (x2 - x1)) + ((y2 - y1) \* (y2 - y1))));

double degree = atan2(y2 - y1, x2 - x1);

for (int i = 0; i <= d; i++) {

for (int j = -line.size; j <= line.size; j++) {

SetPixel(consoleDC, x1 + (i \* cos(degree)) + (j \* cos(degree + 90)), y1 + (i \* sin(degree)) + (j \* sin(degree + 90)), color);

}

}

}

double intensity = 1;

double cr, cg, cb;

//--FLOW--

/\*\_brush(x, y, RGB(intensity < 255 / 2 ? cr / (255 / 2) \* intensity : cr + (255 - cr) / (255 / 2) \* (intensity - (255 / 2)),

intensity < 255 / 2 ? cg / (255 / 2) \* intensity : cg + (255 - cg) / (255 / 2) \* (intensity - (255 / 2)),

intensity < 255 / 2 ? cb / (255 / 2) \* intensity : cb + (255 - cb) / (255 / 2) \* (intensity - (255 / 2)))), intensity++;

\_pen(x, y, RGB(cr + (r \* intensity), cg + (g \* intensity), cb + (b \* intensity))), intensity += 0.01;

if (intensity > 255)

intensity = 0;\*/

queue<pixel> q, r\_p, g\_p, b\_p, obscurity, ccp, b\_size, b\_orientation;

void draw\_cursor() {

int x = mouse\_pos.x, y = mouse\_pos.y;

if (brush\_t == "paint\_brush") {

int r = paint\_brush.size;

for (int d = 0; d < 360; d += 20) {

if (limit(x + (r \* cos(d \* pi / 180)), y + (r \* sin(d \* pi / 180)))) {

q.push({ x + (r \* cos(d \* pi / 180)),y + (r \* sin(d \* pi / 180)),

GetPixel(consoleDC,x + (r \* cos(d \* pi / 180)),y + (r \* sin(d \* pi / 180))) });

SetPixel(consoleDC, mouse\_pos.x + (r \* cos(d \* pi / 180)), mouse\_pos.y + (r \* sin(d \* pi / 180)), RGB(125, 125, 125));

}

}

}

else if (brush\_t == "eraser") {

int r = eraser.size;

for (int d = 0; d < 360; d += 20) {

if (limit(x + (r \* cos(d \* pi / 180)), y + (r \* sin(d \* pi / 180)))) {

q.push({ x + (r \* cos(d \* pi / 180)),y + (r \* sin(d \* pi / 180)),

GetPixel(consoleDC,x + (r \* cos(d \* pi / 180)),y + (r \* sin(d \* pi / 180))) });

SetPixel(consoleDC, mouse\_pos.x + (r \* cos(d \* pi / 180)), mouse\_pos.y + (r \* sin(d \* pi / 180)), RGB(125, 125, 125));

}

}

}

else if (brush\_t == "pen") {

for (int i = -pen.size; i <= pen.size; i += 5) {

if (limit(x + (i \* cos(pen.orientation \* pi / 180)), y + (i \* sin(pen.orientation \* pi / 180)))) {

q.push({ x + (i \* cos(pen.orientation \* pi / 180)), y + (i \* sin(pen.orientation \* pi / 180)),

GetPixel(consoleDC,x + (i \* cos(pen.orientation \* pi / 180)), y + (i \* sin(pen.orientation \* pi / 180))) });

SetPixel(consoleDC, x + (i \* cos(pen.orientation \* pi / 180)), y + (i \* sin(pen.orientation \* pi / 180)), RGB(125, 125, 125));

}

}

}

}

void clear\_pattern(queue<pixel>& q) {

while (!q.empty()) {

SetPixel(consoleDC, q.front().x, q.front().y, q.front().color);

q.pop();

}

}

void display\_cursor() {

clear\_pattern(q);

draw\_cursor();

}

void set\_indicator(queue<pixel>& q, double x, double y) {

for (int i = -1; i < 16; i++) {

q.push({ x,y + i,GetPixel(consoleDC,x,y + i) });

SetPixel(consoleDC, x, y + i, th == 0 ? RGB(0, 0, 0) : RGB(255, 255, 255));

}

}

void show\_pointer(queue<pixel>& q, double x, double y) {

for (int i = 0; i < 360; i += 20) {

q.push({ x + (5 \* cos(i \* pi / 180)),y + (5 \* sin(i \* pi / 180)),GetPixel(consoleDC,x + (5 \* cos(i \* pi / 180)),y + (5 \* sin(i \* pi / 180))) });

SetPixel(consoleDC, x + (5 \* cos(i \* pi / 180)), y + (5 \* sin(i \* pi / 180)), RGB(255, 255, 255));

}

}

void brush\_type(int x, int y) {

if (brush\_t == "paint\_brush") {

\_brush(x, y, RGB(r / 100 \* shade\_value, g / 100 \* shade\_value, b / 100 \* shade\_value));

}

else if (brush\_t == "bucket")

\_bucket(x, y, RGB(r / 100 \* shade\_value, g / 100 \* shade\_value, b / 100 \* shade\_value), true);

else if (brush\_t == "pen")

\_pen(x, y, RGB(r / 100 \* shade\_value, g / 100 \* shade\_value, b / 100 \* shade\_value));

else if (brush\_t == "line") {

double x2, y2;

this\_thread::sleep\_for(chrono::milliseconds(100));

while (true) {

GetCursorPos(&mouse\_pos);

ScreenToClient(consoleWindow, &mouse\_pos);

if ((GetAsyncKeyState((unsigned short)VK\_RBUTTON) & 0x8000) && limit(mouse\_pos.x, mouse\_pos.y)) {

x2 = mouse\_pos.x;

y2 = mouse\_pos.y;

break;

}

}

\_line(x, y, x2, y2, RGB(r / 100 \* shade\_value, g / 100 \* shade\_value, b / 100 \* shade\_value));

this\_thread::sleep\_for(chrono::milliseconds(300));

}

else if (brush\_t == "eraser")

\_eraser(x, y);

else if (brush\_t == "picker") {

COLORREF local\_color = GetPixel(consoleDC, x, y);

r = GetRValue(local\_color);

g = GetGValue(local\_color);

b = GetBValue(local\_color);

clear\_pattern(r\_p);

set\_indicator(r\_p, 1305 + (r / 1.7), 209);

clear\_pattern(g\_p);

set\_indicator(g\_p, 1305 + (g / 1.7), 239);

clear\_pattern(b\_p);

set\_indicator(b\_p, 1305 + (b / 1.7), 269);

}

}

void display\_color\_palette(int pos\_x, int pos\_y) {

int r = 0;

for (double intensity = 255; intensity > 133; intensity -= 1.5, r += 1) {

float degree = 0;

cr = 255, cg = 0, cb = 0;

for (degree; degree < 60; degree += 0.5, cg += 2.125) {

SetPixel(consoleDC, pos\_x + (r \* cos(degree \* pi / 180)), pos\_y + (r \* sin(degree \* pi / 180)), RGB(cr + (255 - cr) / (255 / 2) \* (intensity - (255 / 2)), cg + (255 - cg) / (255 / 2) \* (intensity - (255 / 2)), cb + (255 - cb) / (255 / 2) \* (intensity - (255 / 2))));

}

for (degree; degree < 120; degree += 0.5, cr -= 2.125) {

SetPixel(consoleDC, pos\_x + (r \* cos(degree \* pi / 180)), pos\_y + (r \* sin(degree \* pi / 180)), RGB(cr + (255 - cr) / (255 / 2) \* (intensity - (255 / 2)), cg + (255 - cg) / (255 / 2) \* (intensity - (255 / 2)), cb + (255 - cb) / (255 / 2) \* (intensity - (255 / 2))));

}

for (degree; degree < 180; degree += 0.5, cb += 2.125) {

SetPixel(consoleDC, pos\_x + (r \* cos(degree \* pi / 180)), pos\_y + (r \* sin(degree \* pi / 180)), RGB(cr + (255 - cr) / (255 / 2) \* (intensity - (255 / 2)), cg + (255 - cg) / (255 / 2) \* (intensity - (255 / 2)), cb + (255 - cb) / (255 / 2) \* (intensity - (255 / 2))));

}

for (degree; degree < 240; degree += 0.5, cg -= 2.125) {

SetPixel(consoleDC, pos\_x + (r \* cos(degree \* pi / 180)), pos\_y + (r \* sin(degree \* pi / 180)), RGB(cr + (255 - cr) / (255 / 2) \* (intensity - (255 / 2)), cg + (255 - cg) / (255 / 2) \* (intensity - (255 / 2)), cb + (255 - cb) / (255 / 2) \* (intensity - (255 / 2))));

}

for (degree; degree < 300; degree += 0.5, cr += 2.125) {

SetPixel(consoleDC, pos\_x + (r \* cos(degree \* pi / 180)), pos\_y + (r \* sin(degree \* pi / 180)), RGB(cr + (255 - cr) / (255 / 2) \* (intensity - (255 / 2)), cg + (255 - cg) / (255 / 2) \* (intensity - (255 / 2)), cb + (255 - cb) / (255 / 2) \* (intensity - (255 / 2))));

}

for (degree; degree < 360; degree += 0.5, cb -= 2.125) {

SetPixel(consoleDC, pos\_x + (r \* cos(degree \* pi / 180)), pos\_y + (r \* sin(degree \* pi / 180)), RGB(cr + (255 - cr) / (255 / 2) \* (intensity - (255 / 2)), cg + (255 - cg) / (255 / 2) \* (intensity - (255 / 2)), cb + (255 - cb) / (255 / 2) \* (intensity - (255 / 2))));

}

}

}

void draw(int x, int y) {

brush\_type(x, y);

};

void color\_indicator(COLORREF color) {

for (int d = 0; d < 360; d++) {

for (int r = 0; r <= 11; r++) {

SetPixel(consoleDC, 1450 + (r \* cos(d \* pi / 180)), 156 + (r \* sin(d \* pi / 180)), color);

}

}

}

void ColorPaletteUI(COLORREF body, COLORREF bezels) {

int origin\_x = 1375, origin\_y = 85;

/// COLOR RGB(0, 0, 13)

for (int i = 0; i < 835; i++) {

for (int x = origin\_x - 87; x < 1500; x++) {

SetPixel(consoleDC, x, origin\_y - 108 + i, body);

}

SetPixel(consoleDC, origin\_x - 88, origin\_y - 108 + i, bezels);

SetPixel(consoleDC, origin\_x - 89, origin\_y - 108 + i, bezels);

}

display\_color\_palette(origin\_x + 5, origin\_y);

for (int i = 0; i < 2; i++)

for (float d = 0; d < 360; d += 0.5) {

SetPixel(consoleDC, origin\_x + 5 + ((81 - i) \* cos(d \* pi / 180)), origin\_y + ((81 - i) \* sin(d \* pi / 180)), bezels);

}

for (double d = 0; d < 360; d += 0.5) {

SetPixel(consoleDC, origin\_x + 75 + (12 \* cos(d \* pi / 180)), origin\_y - 4 + 75 + (12 \* sin(d \* pi / 180)), bezels);

}

color\_indicator(RGB(r / 100 \* shade\_value, g / 100 \* shade\_value, b / 100 \* shade\_value));

// Red

for (int lvl = 0; lvl < 15; lvl++) {

for (int i = 150; i >= 0; i--) {

SetPixel(consoleDC, origin\_x - 70 + i, origin\_y + 125 + lvl, RGB(1.7 \* i, 0, 0));

}

}

// Green

for (int lvl = 0; lvl < 15; lvl++) {

for (int i = 150; i >= 0; i--) {

SetPixel(consoleDC, origin\_x - 70 + i, origin\_y + 155 + lvl, RGB(0, 1.7 \* i, 0));

}

}

// Blue

for (int lvl = 0; lvl < 15; lvl++) {

for (int i = 150; i >= 0; i--) {

SetPixel(consoleDC, origin\_x - 70 + i, origin\_y + 185 + lvl, RGB(0, 0, 1.7 \* i));

}

}

//Shade

for (int lvl = 0; lvl < 15; lvl++) {

for (int i = 150; i >= 0; i--) {

SetPixel(consoleDC, origin\_x - 70 + 150 - i, origin\_y + 215 + lvl, RGB(1.7 \* i, 1.7 \* i, 1.7 \* i));

}

}

}

void tool\_bar(COLORREF body, COLORREF bezels) {

for (int i = 0; i < 835; i++) {

for (int x = 0; x < 35; x++) {

SetPixel(consoleDC, x, i, body);

}

SetPixel(consoleDC, 35, i, bezels);

SetPixel(consoleDC, 36, i, bezels);

}

}

COLORREF color[5][2] = {

{ RGB(99,98,90),RGB(68,0,204)},

{ RGB(0,0,0),RGB(0,0,0)},

{ RGB(140,145,127),RGB(34,8,127)},

{ RGB(200,200,200),RGB(25,25,25)},

{ RGB(73, 136, 245),RGB(112, 54, 199)},

};

void top\_bar(COLORREF body, COLORREF bezels) {

for (int y = 0; y < 29; y++) {

for (int x = 0; x < 1286; x++) {

SetPixel(consoleDC, x, y, body);

}

}

for (int x = 0; x < 1288; x++) {

SetPixel(consoleDC, x, 29, bezels);

}

for (int x = 0; x < 1288; x++) {

SetPixel(consoleDC, x, 30, bezels);

}

//size

for (int lvl = 0; lvl < 10; lvl++) {

for (int i = 150; i >= 0; i--) {

SetPixel(consoleDC, 600 - 70 + i, 10 + lvl, color[4][th]);

}

}

for (int i = 0; i <= 336; i++) {

SetPixel(consoleDC, 600 - 140 + marime[i][0], 8 + marime[i][1], th == 0 ? RGB(0, 0, 0) : RGB(255, 255, 255));

}

//orientation

for (int lvl = 0; lvl < 10; lvl++) {

for (int i = 150; i >= 0; i--) {

SetPixel(consoleDC, 600 + 270 - 70 + i, 10 + lvl, color[4][th]);

}

}

for (int i = 0; i <= 410; i++) {

SetPixel(consoleDC, 600 + 270 - 140 + orientare[i][0], 4 + orientare[i][1], th == 0 ? RGB(0, 0, 0) : RGB(255, 255, 255));

}

}

void initialize\_canvas() {

int origin\_x = 85, origin\_y = 75;

for (int y = 0; y < 650; y++) {

for (int x = 0; x < 1200; x++) {

SetPixel(consoleDC, origin\_x + x, origin\_y + y, RGB(255, 255, 255));

}

}

}

void Theme() {

for (int y = 0; y < 500; y++) {

for (int x = 0; x < 800; x++) {

SetPixel(consoleDC, x, y, RGB(255, 255, 255));

}

}

for (int i = 0; i <= 7366; i++) {

SetPixel(consoleDC, 20 + intro[i][0], 20 + intro[i][1], RGB(0, 0, 0));

}

//

for (int x = 50; x < 350; x++)

SetPixel(consoleDC, x, 99 + 50, RGB(50, 50, 50));//

for (int y = 100 + 50; y < 300 + 50; y++) {//

SetPixel(consoleDC, 49, y, RGB(50, 50, 50));//

for (int x = 50; x < 350; x++) {//

SetPixel(consoleDC, x, y, RGB(0, 0, 0));//

}

SetPixel(consoleDC, 350, y, RGB(50, 50, 50));//

}

for (int x = 50; x < 350; x++)//

SetPixel(consoleDC, x, 300 + 50, RGB(50, 50, 50));//

//

for (int y = 120 + 50; y < 300 + 50; y++) {//

for (int x = 0; x < 20; x++)//

SetPixel(consoleDC, 50 + x, y, RGB(242, 242, 242));//

SetPixel(consoleDC, 50 + 20, y, RGB(200, 200, 200));//

SetPixel(consoleDC, 50 + 20 + 1, y, RGB(200, 200, 200));//

}

for (int x = 50; x < 275; x++) {//

for (int y = 0 + 50; y < 20 + 50; y++)//

SetPixel(consoleDC, x, 100 + y, RGB(242, 242, 242));

SetPixel(consoleDC, x, 120 + 50, RGB(200, 200, 200));//

SetPixel(consoleDC, x, 120 + 1 + 50, RGB(200, 200, 200));//

}

for (int y = 100 + 50; y < 300 + 50; y++) {//

for (int x = 0; x < 75; x++) {//

SetPixel(consoleDC, 275 + x, y, RGB(242, 242, 242));//

}//

SetPixel(consoleDC, 275, y, RGB(200, 200, 200));//

SetPixel(consoleDC, 275 + 1, y, RGB(200, 200, 200));//

}

//

for (int x = 50; x < 350; x++)

SetPixel(consoleDC, x + 375, 99 + 50, RGB(50, 50, 50));

for (int y = 100 + 50; y < 300 + 50; y++) {

SetPixel(consoleDC, 49 + 375, y, RGB(50, 50, 50));

for (int x = 50; x < 350; x++) {

SetPixel(consoleDC, x + 375, y, RGB(0, 0, 0));

}

SetPixel(consoleDC, 350 + 375, y, RGB(50, 50, 50));

}

for (int x = 50; x < 350; x++)

SetPixel(consoleDC, x + 375, 300 + 50, RGB(50, 50, 50));

//

for (int y = 120 + 50; y < 300 + 50; y++) {

for (int x = 0; x < 20; x++)

SetPixel(consoleDC, 375 + 50 + x, y, RGB(7, 9, 15));

SetPixel(consoleDC, 375 + 50 + 20, y, RGB(34, 45, 64));

SetPixel(consoleDC, 375 + 50 + 20 + 1, y, RGB(34, 45, 64));

}

for (int x = 50; x < 275; x++) {

for (int y = 0 + 50; y < 20 + 50; y++)

SetPixel(consoleDC, 375 + x, 100 + y, RGB(7, 9, 15));

SetPixel(consoleDC, 375 + x, 120 + 50, RGB(34, 45, 64));

SetPixel(consoleDC, 375 + x, 120 + 1 + 50, RGB(34, 45, 64));

}

for (int y = 100 + 50; y < 300 + 50; y++) {

for (int x = 0; x < 75; x++) {

SetPixel(consoleDC, 375 + 275 + x, y, RGB(7, 9, 15));

}

SetPixel(consoleDC, 375 + 275, y, RGB(34, 45, 64));

SetPixel(consoleDC, 375 + 275 + 1, y, RGB(34, 45, 64));

}

//

for (int r = 0; r < 20; r++) {

for (double d = 0; d < 360; d += 0.1) {

SetPixel(consoleDC, 725 + (r \* cos(d \* pi / 180)), 425 + (r \* sin(d \* pi / 180)), RGB(42, 116, 235));

}

}

for (int r = -13; r < 16; r++) {

SetPixel(consoleDC, 725 + r, 425, RGB(255, 255, 255));

}

for (double r = 0; r < 10; r += 0.5) {

SetPixel(consoleDC, 725 + 15 + (r \* cos(130 \* pi / 180)), 425 + (r \* sin(130 \* pi / 180)), RGB(255, 255, 255));

}

for (double r = 0; r < 10; r += 0.5) {

SetPixel(consoleDC, 725 + 15 + (r \* cos(-130 \* pi / 180)), 425 + (r \* sin(-130 \* pi / 180)), RGB(255, 255, 255));

}

//

for (int i = 0; i <= 1134; i++) {

SetPixel(consoleDC, 575 + continuati[i][0], 415 + continuati[i][1], RGB(0, 0, 0));

}

}

int pb[] = {

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,2,2,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,2,2,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,2,2,2,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,2,2,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,2,2,2,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,2,2,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,2,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,2,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,1,1,1,2,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,1,1,2,2,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,1,1,2,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,1,1,2,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,1,3,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,1,1,3,3,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,1,1,1,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,1,1,1,2,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,1,1,1,2,2,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,1,1,1,2,2,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,1,1,1,2,2,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,1,1,1,2,2,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,1,1,2,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,1,2,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

};

int bk[] = {

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,2,2,2,1,1,1,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,2,2,2,2,2,2,1,1,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,1,3,1,2,2,2,2,2,2,2,1,1,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,1,3,3,1,1,2,2,2,2,2,2,2,1,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,1,3,3,3,3,1,1,2,2,2,2,2,2,2,1,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,1,3,3,3,3,3,3,1,1,1,3,3,3,3,3,3,1,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,1,1,3,3,3,3,3,3,0,0,0,0,0,

0,0,0,0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,1,1,1,3,3,3,3,1,0,0,0,0,

0,0,0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,1,1,3,3,3,3,0,0,0,0,

0,0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,1,3,3,3,3,0,0,0,0,

0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,1,1,3,3,3,0,0,0,0,

0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,1,3,3,3,0,0,0,0,

0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,1,3,0,0,0,0,0,

0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,1,3,0,0,0,0,0,

0,0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,1,3,0,0,0,0,0,

0,0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,1,3,0,0,0,0,0,

0,0,0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,1,0,3,0,0,0,0,0,

0,0,0,0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,1,0,0,3,3,0,0,0,0,

0,0,0,0,0,0,0,0,0,1,1,3,3,3,3,3,3,3,3,3,3,3,1,0,0,0,3,3,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,1,0,0,0,0,3,3,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,1,3,3,3,3,3,3,3,1,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,3,3,3,3,1,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

};

int pn[] = {

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,3,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,3,3,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,3,3,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,3,3,3,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,3,3,3,3,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,2,1,1,1,1,1,1,1,3,3,3,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,2,2,2,2,2,1,1,1,1,3,3,3,3,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,2,2,2,2,2,2,2,1,1,1,3,3,3,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,2,2,2,2,2,2,2,2,3,3,3,3,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,2,2,2,2,0,2,2,2,3,3,3,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,2,2,2,0,0,2,2,2,2,2,3,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,2,0,0,2,2,2,2,2,2,2,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,2,0,2,2,2,2,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,2,2,2,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

};

int fg[] = {

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,1,1,0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,1,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,

0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,0,0,0,

0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,0,0,0,

0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,0,0,0,

0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,

0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,

0,0,0,0,0,1,0,0,0,0,1,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,1,1,0,0,1,0,0,1,1,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

};

int ln[] = {

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,1,0,0,2,2,2,2,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,1,0,2,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,1,0,2,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,1,0,2,0,0,1,0,0,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,1,1,1,1,0,0,0,0,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,2,0,0,0,0,0,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,2,0,0,0,0,0,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,2,0,0,0,0,0,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,2,0,0,0,0,0,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,2,0,0,0,0,0,2,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,2,0,0,0,0,0,2,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,0,0,0,0,0,2,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,0,0,0,1,1,1,1,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2,2,2,0,1,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2,2,2,2,0,0,1,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2,2,2,0,0,0,1,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

};

int er[] = {

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2,2,2,2,1,1,1,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,2,2,2,2,2,2,2,1,1,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,1,3,1,2,2,2,2,2,2,2,1,1,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,1,3,3,1,1,2,2,2,2,2,2,2,1,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,1,3,3,3,3,1,1,2,2,2,2,2,2,2,1,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,1,3,3,3,3,3,3,1,1,2,2,2,2,2,2,2,1,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,1,2,2,2,2,2,2,2,1,0,0,0,0,

0,0,0,0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,1,2,2,2,2,2,2,2,1,0,0,0,

0,0,0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,3,3,1,2,2,2,2,2,2,2,1,0,0,

0,0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,1,2,2,2,2,2,2,1,0,0,

0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,1,1,2,2,2,2,1,0,0,

0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,1,2,2,2,1,0,0,

0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,1,2,1,0,0,0,

0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,1,0,0,0,0,

0,0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,1,0,0,0,0,0,

0,0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,1,0,0,0,0,0,0,

0,0,0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,3,1,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,3,3,3,3,3,1,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,1,1,3,3,3,3,3,3,3,3,3,3,3,1,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,1,3,3,3,3,3,3,3,3,3,1,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,1,3,3,3,3,3,3,3,1,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

};

int pk[] = {

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,2,2,2,0,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,2,0,0,2,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,2,0,1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,2,1,1,2,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,2,0,0,2,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,2,0,2,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,1,3,0,0,0,0,1,0,2,2,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,1,3,3,3,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,1,3,3,3,3,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,1,3,3,3,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,

};

void display\_icon(int v[32 \* 32], double x, double y) {

for (double d = 0; d < 360; d += 0.5) {

for (int r = 1; r <= 14; r++) {

SetPixel(consoleDC, x + 16 + (r \* cos(d \* pi / 180)), y + 16 + (r \* sin(d \* pi / 180)), color[3][th]);

}

}

for (int i = 0; i < 32; i++) {

for (int j = 0; j < 32; j++) {

switch (v[i \* 32 + j]) {

case 1:

SetPixel(consoleDC, x + j, y + i, color[0][th]);

break;

case 2:

SetPixel(consoleDC, x + j, y + i, color[1][th]);

break;

case 3:

SetPixel(consoleDC, x + j, y + i, color[2][th]);

break;

}

}

}

}

int main() {

COLORREF body = NULL, bezels = NULL;

RECT ConsoleRect;

GetWindowRect(consoleWindow, &ConsoleRect);

MoveWindow(consoleWindow, ConsoleRect.left, ConsoleRect.top, 800, 500, TRUE);

Theme();

while (true) {

GetCursorPos(&mouse\_pos);

ScreenToClient(consoleWindow, &mouse\_pos);

if (GetAsyncKeyState((unsigned short)VK\_RBUTTON) & 0x8000) {

if (sqrt((mouse\_pos.x - 725) \* (mouse\_pos.x - 725)) + ((mouse\_pos.y - 425) \* (mouse\_pos.y - 425)) <= 19 && (bezels != NULL && body != NULL)) {

break;

}

if (mouse\_pos.x > 49 && mouse\_pos.x < 350 && mouse\_pos.y>49 && mouse\_pos.y < 300) {

body = RGB(242, 242, 242), bezels = RGB(200, 200, 200), th = 0;

for (int x = 50; x < 350; x++)

SetPixel(consoleDC, x, 99 + 50, RGB(0, 153, 255));

for (int y = 100 + 50; y < 300 + 50; y++) {

SetPixel(consoleDC, 49, y, RGB(0, 153, 255));

SetPixel(consoleDC, 350, y, RGB(0, 153, 255));

}

for (int x = 50; x < 350; x++)

SetPixel(consoleDC, x, 300 + 50, RGB(0, 153, 255));

//

for (int x = 50; x < 350; x++)

SetPixel(consoleDC, x + 375, 99 + 50, RGB(50, 50, 50));

for (int y = 100 + 50; y < 300 + 50; y++) {

SetPixel(consoleDC, 49 + 375, y, RGB(50, 50, 50));

SetPixel(consoleDC, 350 + 375, y, RGB(50, 50, 50));

}

for (int x = 50; x < 350; x++)

SetPixel(consoleDC, x + 375, 300 + 50, RGB(50, 50, 50));

}

if (mouse\_pos.x > 424 && mouse\_pos.x < 725 && mouse\_pos.y>99 && mouse\_pos.y < 300) {

body = RGB(7, 9, 15), bezels = RGB(34, 45, 64), th = 1;

for (int x = 50; x < 350; x++)

SetPixel(consoleDC, x + 375, 99 + 50, RGB(0, 153, 255));

for (int y = 100 + 50; y < 300 + 50; y++) {

SetPixel(consoleDC, 49 + 375, y, RGB(0, 153, 255));

SetPixel(consoleDC, 350 + 375, y, RGB(0, 153, 255));

}

for (int x = 50; x < 350; x++)

SetPixel(consoleDC, x + 375, 300 + 50, RGB(0, 153, 255));

//

for (int x = 50; x < 350; x++)

SetPixel(consoleDC, x, 99 + 50, RGB(50, 50, 50));

for (int y = 100 + 50; y < 300 + 50; y++) {

SetPixel(consoleDC, 49, y, RGB(50, 50, 50));

SetPixel(consoleDC, 350, y, RGB(50, 50, 50));

}

for (int x = 50; x < 350; x++)

SetPixel(consoleDC, x, 300 + 50, RGB(50, 50, 50));

}

}

}

MoveWindow(consoleWindow, ConsoleRect.left, ConsoleRect.top, 1500, 850, TRUE);

this\_thread::sleep\_for(chrono::seconds(1));

initialize\_canvas();

ColorPaletteUI(body, bezels);

tool\_bar(body, bezels);

top\_bar(body, bezels);

GetCursorPos(&mouse\_pos);

ScreenToClient(consoleWindow, &mouse\_pos);

set\_indicator(r\_p, 1305, 209);

set\_indicator(g\_p, 1305, 239);

set\_indicator(b\_p, 1305, 269);

set\_indicator(obscurity, 1305, 299);

set\_indicator(b\_size, 530, 8);

set\_indicator(b\_orientation, 800, 8);

display\_icon(pb, 3, 50);

\_bucket(11, 57, color[4][th], false);

display\_icon(bk, 3, 125);

display\_icon(pn, 3, 200);

display\_icon(ln, 3, 275);

display\_icon(fg, 3, 350);

display\_icon(er, 3, 425);

for (double d = 0; d < 360; d += 0.5) {

for (int r = 1; r <= 14; r++) {

SetPixel(consoleDC, 1290 + 16 + (r \* cos(d \* pi / 180)), 140 + 16 + (r \* sin(d \* pi / 180)), color[3][th]);

}

}

for (int i = 0; i < 32; i++) {

for (int j = 0; j < 32; j++) {

switch (pk[i \* 32 + j]) {

case 1:

SetPixel(consoleDC, 1290 + j, 140 + i, color[0][th]);

break;

case 2:

SetPixel(consoleDC, 1290 + j, 140 + i, color[1][th]);

break;

case 3:

SetPixel(consoleDC, 1290 + j, 140 + i, color[2][th]);

break;

}

}

}

int last = 0;

int x = mouse\_pos.x, y = mouse\_pos.y;

while (true) {

GetCursorPos(&mouse\_pos);

ScreenToClient(consoleWindow, &mouse\_pos);

if (GetAsyncKeyState((unsigned short)VK\_RBUTTON) & 0x8000) {

int pos\_x = mouse\_pos.x, pos\_y = mouse\_pos.y;

clear\_pattern(q);

if (sqrt(pow(mouse\_pos.x - (3 + 16), 2) + pow(mouse\_pos.y - (50 + 16), 2)) <= 16 && last != 0) {

brush\_t = "paint\_brush";

clear\_pattern(b\_size);

clear\_pattern(b\_orientation);

set\_indicator(b\_size, 530 + (paint\_brush.size \* 3), 8);

set\_indicator(b\_orientation, 800, 8);

\_bucket(11, 57, color[4][th], false);

last == 6 ? \_bucket(1290 + 16, 140 + 25, color[3][th], false) : \_bucket(15, 57 + (75 \* last), color[3][th], false);

last = 0;

}

else if (sqrt(pow(mouse\_pos.x - (3 + 16), 2) + pow(mouse\_pos.y - (125 + 16), 2)) <= 16 && last != 1) {

brush\_t = "bucket";

clear\_pattern(b\_size);

clear\_pattern(b\_orientation);

set\_indicator(b\_size, 530, 8);

set\_indicator(b\_orientation, 800, 8);

\_bucket(11, 134, color[4][th], false);

last == 6 ? \_bucket(1290 + 16, 140 + 25, color[3][th], false) : \_bucket(15, 57 + (75 \* last), color[3][th], false);

last = 1;

}

else if (sqrt(pow(mouse\_pos.x - (3 + 16), 2) + pow(mouse\_pos.y - (200 + 16), 2)) <= 16 && last != 2) {

brush\_t = "pen";

clear\_pattern(b\_size);

clear\_pattern(b\_orientation);

set\_indicator(b\_size, 530 + (pen.size \* 3), 8);

set\_indicator(b\_orientation, 800 + (pen.orientation \* 0.416), 8);

\_bucket(11, 208, color[4][th], false);

last == 6 ? \_bucket(1290 + 16, 140 + 25, color[3][th], false) : \_bucket(15, 57 + (75 \* last), color[3][th], false);

last = 2;

}

else if (sqrt(pow(mouse\_pos.x - (3 + 16), 2) + pow(mouse\_pos.y - (275 + 16), 2)) <= 16 && last != 3) {

brush\_t = "line";

clear\_pattern(b\_size);

clear\_pattern(b\_orientation);

set\_indicator(b\_size, 530 + (line.size \* 3), 8);

set\_indicator(b\_orientation, 800, 8);

\_bucket(17, 280, color[4][th], false);

last == 6 ? \_bucket(1290 + 16, 140 + 25, color[3][th], false) : \_bucket(15, 57 + (75 \* last), color[3][th], false);

last = 3;

}

else if (sqrt(pow(mouse\_pos.x - (3 + 16), 2) + pow(mouse\_pos.y - (350 + 16), 2)) <= 16 && last != 4) {

brush\_t = "figures";

clear\_pattern(b\_size);

clear\_pattern(b\_orientation);

set\_indicator(b\_size, 530, 8);

set\_indicator(b\_orientation, 800, 8);

\_bucket(17, 353, color[4][th], false);

last == 6 ? \_bucket(1290 + 16, 140 + 25, color[3][th], false) : \_bucket(15, 57 + (75 \* last), color[3][th], false);

last = 4;

}

else if (sqrt(pow(mouse\_pos.x - (3 + 16), 2) + pow(mouse\_pos.y - (425 + 16), 2)) <= 16 && last != 4) {

brush\_t = "eraser";

clear\_pattern(b\_size);

clear\_pattern(b\_orientation);

set\_indicator(b\_size, 530 + (eraser.size \* 3), 8);

set\_indicator(b\_orientation, 800, 8);

\_bucket(17, 428, color[4][th], false);

last == 6 ? \_bucket(1290 + 16, 140 + 25, color[3][th], false) : \_bucket(15, 57 + (75 \* last), color[3][th], false);

last = 5;

}

//1290 + 16 140 + 16

else if (sqrt(pow(mouse\_pos.x - (1290 + 16), 2) + pow(mouse\_pos.y - (140 + 16), 2)) <= 14 && last != 6) {

brush\_t = "picker";

clear\_pattern(b\_size);

clear\_pattern(b\_orientation);

set\_indicator(b\_size, 530, 8);

set\_indicator(b\_orientation, 800, 8);

\_bucket(1290 + 16, 140 + 25, color[4][th], false);

\_bucket(15, 57 + (75 \* last), color[3][th], false);

last = 6;

}

if (limit(pos\_x, pos\_y))

draw(pos\_x, pos\_y);

else if (sqrt(((1380 - pos\_x) \* (1380 - pos\_x)) + ((85 - pos\_y) \* (85 - pos\_y))) <= 80) {

clear\_pattern(ccp);

COLORREF local\_color = GetPixel(consoleDC, pos\_x, pos\_y);

r = GetRValue(local\_color), g = GetGValue(local\_color), b = GetBValue(local\_color);

show\_pointer(ccp, pos\_x, pos\_y);

clear\_pattern(r\_p);

set\_indicator(r\_p, 1305 + (r / 1.7), 209);

clear\_pattern(g\_p);

set\_indicator(g\_p, 1305 + (g / 1.7), 239);

clear\_pattern(b\_p);

set\_indicator(b\_p, 1305 + (b / 1.7), 269);

color\_indicator(RGB(r / 100 \* shade\_value, g / 100 \* shade\_value, b / 100 \* shade\_value));

}

else if (pos\_x >= 1305 && pos\_y >= 209 && pos\_x <= 1455 && pos\_y <= 224) {

//COLORREF local\_color = GetPixel(consoleDC, pos\_x, pos\_y);

//local\_color != RGB(255, 255, 255) ? r = GetRValue(local\_color):0;

r = (pos\_x - 1305) \* 1.7;

clear\_pattern(r\_p);

set\_indicator(r\_p, pos\_x, 209);

color\_indicator(RGB(r / 100 \* shade\_value, g / 100 \* shade\_value, b / 100 \* shade\_value));

}

else if (pos\_x >= 1305 && pos\_y >= 239 && pos\_x <= 1455 && pos\_y <= 254) {

g = (pos\_x - 1305) \* 1.7;

clear\_pattern(g\_p);

set\_indicator(g\_p, pos\_x, 239);

color\_indicator(RGB(r / 100 \* shade\_value, g / 100 \* shade\_value, b / 100 \* shade\_value));

}

else if (pos\_x >= 1305 && pos\_y >= 269 && pos\_x <= 1455 && pos\_y <= 284) {

b = (pos\_x - 1305) \* 1.7;

clear\_pattern(b\_p);

set\_indicator(b\_p, pos\_x, 269);

color\_indicator(RGB(r / 100 \* shade\_value, g / 100 \* shade\_value, b / 100 \* shade\_value));

}

else if (pos\_x >= 1305 && pos\_y >= 299 && pos\_x <= 1455 && pos\_y <= 314) {

shade\_value = 100 - (0.66 \* (pos\_x - 1305));

clear\_pattern(obscurity);

set\_indicator(obscurity, pos\_x, 299);

color\_indicator(RGB(r / 100 \* shade\_value, g / 100 \* shade\_value, b / 100 \* shade\_value));

}

else if (pos\_x >= 530 && pos\_y >= 12 && pos\_x <= 680 && pos\_y <= 22) {

if (brush\_t == "paint\_brush")

paint\_brush.size = (pos\_x - 530) \* 0.33;

else if (brush\_t == "pen")

pen.size = (pos\_x - 530) \* 0.33;

else if (brush\_t == "line")

line.size = (pos\_x - 530) \* 0.33;

else if (brush\_t == "eraser")

eraser.size = (pos\_x - 530) \* 0.33;

clear\_pattern(b\_size);

set\_indicator(b\_size, pos\_x, 8);

}

else if (pos\_x >= 800 && pos\_y >= 12 && pos\_x <= 951 && pos\_y <= 22) {

if (brush\_t == "pen")

pen.orientation = (pos\_x - 800) \* 2.4;

clear\_pattern(b\_orientation);

set\_indicator(b\_orientation, pos\_x, 8);

}

}

else if ((x != mouse\_pos.x || y != mouse\_pos.y) && limit(mouse\_pos.x, mouse\_pos.y)) {

x = mouse\_pos.x, y = mouse\_pos.y;

display\_cursor();

}

}

ReleaseDC(consoleWindow, consoleDC);

cin.ignore();

}

**Bibliografie:**

**-Referințe cod:**

https://docs.microsoft.com/en-us/windows/win32/api/wingdi/nf-wingdi-getpixel

https://docs.microsoft.com/en-us/windows/win32/api/wingdi/nf-wingdi-setpixel

https://docs.microsoft.com/en-us/windows/win32/inputdev/virtual-key-codes

https://docs.microsoft.com/en-us/windows/win32/api/winuser/nf-winuser-getcursorpos

https://docs.microsoft.com/en-us/windows/win32/api/winuser/nf-winuser-screentoclient

https://stackoverflow.com/questions/12378642/c-pixels-in-console-window

Codul a fost realizat de **Boroica Marius** iar câteva părți grafice au fost făcute de **Ganea Alexandru**.