创意说明

**主题：**是否有这样一个铭心刻骨的旧爱，能在你的每一个新欢上找到影子。青梅枯萎，竹马老去，从此爱上的每一个人都像你。

程序用一个矩阵模拟一个人某些方面的特质。

程序开始，随机生成一个矩阵，代表一个至爱的人。程序每次运行，生成的矩阵都不一样，表示每个人的至爱都不一样（废话！！！）。

接着的恋爱生活，采用**Robert J. Sternberg**的**爱情三角理论**，该理论认为所有的爱情体验都是由**激情**、**亲密**和**承诺**三大要素所构成的。爱情需要经营，爱情三要素同时具备时，称之为**完美爱情。**当它们荡然无存时，爱情的巨轮也将黯然触礁。理论上，一段爱情可以很长很长，但我们知道，这很难很难，程序中靠RP，现实中靠智慧。

之后的相遇，是谁？在什么时候？就像程序中的随机数一样，仍然是个未知数。

**矩阵相同的元素数>=5,表示此人与至爱相像，成为恋人；否则为不像，变为路人或朋友。**

无休止的循环，模拟在爱与被爱间重复的一生，如果哪天你爱不动了，请按下Esc键，也许是爱请的终结，又或者是另一种爱的开始。

//以下是代码的word版本

#include<iostream>

#include<ctime>

#include<cstdlib>

#include<iomanip>

#include<Windows.h>

using namespace std;

void show\_my\_sweatheart(int \*sweatheart,int times) //秀恩爱

{

cout << " ┏━━━┓\n"

<< " ┃" << sweatheart[0] << ' ' << sweatheart[1] << " " << sweatheart[2] << "┃\n"

<< " ┃" << sweatheart[3] << ' ' << sweatheart[4] << " " << sweatheart[5] << "┃"

<< "\t\t 第" << setw(2) <<times<< "次相遇\n"

<< " ┃" << sweatheart[6] << ' ' << sweatheart[7] << " " << sweatheart[8] << "┃"

<< " \t 试问人生能有几次相遇？\n"

<< " ┗┳━┳┛\n"

<< " ┤├\n\n";

}

void sweat\_life() //默然相爱，寂静欢喜

{

bool Intimacy = true,Passion = true,Commitment = true;//亲密、激情、承诺缺一不可

int days= 0; //爱情长跑的时间

while (Intimacy || Passion || Commitment) //爱情的经营

{

days+=60;

cout << " 第" << setw(3) << days << " 天 ";

if (!(rand() % 2)&&Intimacy)

Intimacy = false, cout << "失了亲密 ";

if (!(rand() % 2)&& Passion)

Passion = false, cout << "少了激情 ";

if (!(rand() % 2)&& Commitment)

Commitment = false, cout << "没了承诺 ";

if (Intimacy && Passion && Commitment)

cout << "如胶似漆·形影不离";

cout << endl;

} //爱情三角轰然倒塌之时，爱情的巨轮也将黯然触礁

cout << "\n 曾经的海誓山盟，如今的劳燕分飞，有些人，一旦失去，便是一辈子\n"

<< "━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━\n";

}

int main()

{

int greatest\_love[9]; //greatest\_love代表至爱

int new\_sweatheart[9],times=1; //new\_sweatheart代表新的恋人、time代表相遇的次数

for (int i = 0; i < 9; ++i)

greatest\_love[i] = rand() % 2; //毫无疑问，每个人心中那个至爱的人都是不一样的

cout << "\n >那个我至爱的人<\n";

show\_my\_sweatheart(greatest\_love,1);

sweat\_life();

Sleep(5000); //让我等等那迟到的爱情

srand(time(NULL));

int sth\_in\_common; //两个人相同的地方

cout<< " ┏━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┓\n"

<< "━━┫曾经无数次的回眸，才换来今生的一次相遇，爱情，求不得，求之不得┣━━━━━"

<< " ┗━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━┛\n";

while (true) //无休止的循环，在爱与被爱间重复的一生

{

times++;

sth\_in\_common = 0;

Sleep((rand() % 10 + 1) \* 1000); //未知的时间

for (int i = 0; i < 9; ++i) //未知的人

{

new\_sweatheart[i] = rand() % 2;

if (greatest\_love[i] == new\_sweatheart[i])

sth\_in\_common++; //相同的地方+1

}

if (sth\_in\_common >= 5) //请让我找到至爱的影子

{

cout << " 我的恋人,每月的情人节，让我送你一支玫瑰花\n";

show\_my\_sweatheart(new\_sweatheart,times);

sweat\_life();

}

else //原谅我不能将就

{

show\_my\_sweatheart(new\_sweatheart,times);

cout << " 咫尺天涯，能够相遇，已是不易，而你，注定要从我的全世界路过，我不能将就\n"

<< "━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━\n";

}

if (GetAsyncKeyState(VK\_ESCAPE)) //漫漫人生路，如果哪天你再也爱不动了，请按下Esc键

break;

}

cout << " 青梅枯萎，竹马老去，从此爱上的每一个人都像你\n"

<< " 所谓伊人，在水一方，从此爱上的每一个人都像你\n"

<< " 此去经年，红烛依旧，从此爱上的每一个人都像你\n\n"

<< " 愿那些我爱过的的人一切安好，愿那个我至爱的人一生幸福...\n\n\n\n";

system("pause");

return 0;

}