

#pragma once	自動作成される
static int card[55]; static int hai[12]; static int keta[12]; static int yaku[12]; static int start_flag=0; static int hoji1=0; static int hoji2=0; static int hoji3=0; static int hoji4=0; static int hoji5=0; static int tokuten=0;	カードの配列 1-52 を入れ その後 1-13、101-113、201-213、301-313 にする 最初の 5 枚と後からの 5 枚のカードデータを収納 上記を下 1 ケタにする 313 なら 13 とする ペアの解析に使用する ルーチンの通過を制御する チェックに使用する 点数に使用する
void DispHai(int nohai,int xx){ Graphics^ gr=this->CreateGraphics(); int X0=10,Y0=10; String^ filename; filename="c:¥¥card¥¥1_sp.gif"; if(nohai==0) filename="c:¥¥card¥¥0_r1.gif" ; //Pen^ pen1=gcnew Pen(col); if(nohai>0 && nohai<14) filename="c:¥¥card¥¥"+nohai+"_sp.gif" ; if(nohai>100 && nohai<114){nohai=nohai-100; filename="c:¥¥card¥¥"+nohai+"_cl.gif" ;} if(nohai>200 && nohai<214){nohai=nohai-200; filename="c:¥¥card¥¥"+nohai+"_di.gif" ;} if(nohai>300 && nohai<314){nohai=nohai-300; filename="c:¥¥card¥¥"+nohai+"_he.gif" ;} //原画像を左上に描画する if(xx==1){ Bitmap^ bmap1; //bmap の確保 bmap1=gcnew Bitmap(filename); gr->DrawImage(bmap1,X0,Y0);	トランプの図柄をファイルから読み込む Gif を読み込むためにモードをセット カードの表示位置 カードのファイル名のセット とりあえず、スペードの 1 にする 0 の時、カードの裏面のファイルを設定 1-13 の時スペードを設定 101 から 113 の時クローバを設定 201-213 のときダイヤを設定 301-313 の時、ハートをセット xx=1 のとき 1 枚目(左)を表示する xx=2 のとき 2 枚目(左から 2 枚目)を表示する

<pre> } if(xx==2){ Bitmap^ bmap2; //bmap の確保 bmap2=gcnew Bitmap(filename); gr->DrawImage(bmap2,X0+100,Y0); } if(xx==3){ Bitmap^ bmap3; //bmap の確保 bmap3=gcnew Bitmap(filename); gr->DrawImage(bmap3,X0+200,Y0); } if(xx==4){ Bitmap^ bmap4; //bmap の確保 bmap4=gcnew Bitmap(filename); gr->DrawImage(bmap4,X0+300,Y0); } if(xx==5){ Bitmap^ bmap5; //bmap の確保 bmap5=gcnew Bitmap(filename); gr->DrawImage(bmap5,X0+400,Y0); } } </pre>	<p>xx=3 のとき 3 枚目(左から 3 枚目)を表示する</p> <p>xx=4 のとき 4 枚目(左から 4 枚目)を表示する</p> <p>xx=5 のとき 5 枚目(いちばん右)を表示する</p>
<pre> #pragma endregion private: System::Void Form1_Paint(System::Object^ sender, System::Windows::Forms::PaintEventArgs^ e) { //int X0=10,Y0=10; Graphics^ gr=e->Graphics; //Pen^ pen; //Color color_src,color_dest; } </pre>	<p>図を描く用意をする</p>
<pre> private: System::Void button1_Click(System::Object^ sender, System::EventArgs^ e) { if(start_flag ==0){ //一度通過したら 2 度目を禁止す る </pre>	<p>コマンドボタン1を押した場合のプログラム【ゲーム開始】</p> <p>一度通過したら 2 度目を禁止する</p> <p>文字列の用意</p>

int amax,xrand,nohai;	
int xx,yy,y;	
char buf[80];	グラフィックの用意
Graphics^ gr=this->CreateGraphics();	文字列の用意
//文字の用意	文字列の用意
String^ string1;	
String^ string2;	省略可能
Brush^ brush=gcnew SolidBrush(Color::Black);	省略可能
System::Drawing::Font^ font=gcnew	省略可能
System::Drawing::Font("MS ゴシック",10);	省略可能
//文字の用意終了	文字の用意終了
Random^ rand=gcnew Random(); //で乱数 randを	乱数の初期化
生成し	
ておき、	
//2) 0 以上 int amax 未満の乱数を発生したい場合	52枚をだぶらないように配布
には、	初期化 0 を入れる
string2="乱数値";	
amax=52;	y < 20 に成るまでLOOPの間を繰り返す
for(xx=1;xx<54;xx++) card[xx]=0;	乱数 1 から 52
yy=1;	ダブりの確認値 0
while(yy<20){	同じ数字が出たらダブリ確認値 1 にする
xrand=rand->Next(amax)+1;	ダブらない時のみ乱数を代入する
int flag=0;	配列 card(z)に乱数を保存する
for(xx=1;xx<53;xx++) {if(card[xx]==xrand)	発生された乱数値を繋げていく
flag=1;}	loop数yを1増加させる
if(flag==0){ card[yy]=xrand;	
//文字で乱数を確認	
sprintf_s(buf,80," %3d ",xrand);	
string1=gcnew String(buf);	While y < 53 の終了
//gr->DrawString(string1,font,brush,(float)16*yy,	
(float)450);	
string2=string2+string1;	
//文字で確認終了	テキストボックス 3 にこの結果を表示
yy++;}	
}	1から52のカードをわかりやすく

<pre>//文字で表示 //MessageBox::Show(string2); textBox3->Text=string2; string2="変換値"; for(y=1;y<53;y++){ if(card[y] > 13 && card[y] <= 26) card[y] = card[y] + 100 - 13; if(card[y] > 26 && card[y] <= 39) card[y] = card[y] + 200 - 26; if(card[y] > 39 && card[y] <= 52) card[y] = card[y] + 300 - 39; } //end for y for(xx=1;xx<6;xx++){ hai[xx]=card[xx]; //文字で乱数を確認 sprintf_s(buf,80," %3d ",hai[xx]); string1=gcnw String(buf); string2=string2+string1; //文字で確認終了 } //文字で表示 textBox4->Text=string2; for(xx=1;xx<6;xx++){ nohai=hai[xx]; DispHai(nohai,xx); } hoji1=0; hoji2=0; hoji3=0; hoji4=0; hoji5=0; start_flag=1; //一度通過したら 2 度目を禁止するフ ラッグ</pre>	<p>1-13 101-113 201-213 301-313に変更</p> <p>13以下はクラブ</p> <p>card(y) > 13 の場合 card(y)に100を足し13を引く たとえば 14の場合101となる。ダイヤとする</p> <p>card(y) > 26の場合 card(y)に200を足し26を引く たとえば 27の場合202となる。ハートとする</p> <p>card(y) > 39の場合 card(y)に300を足</p> <p>発生された乱数値を繋げていく</p> <p>テキストボックス 4 にこの結果を表示</p> <p>この数値をトランプとして表示する</p> <p>表示ルーチンに飛ぶ</p> <p>チェックボックス 1 値を 0 に初期化</p> <p>チェックボックス 2 値を 0 に初期化</p> <p>チェックボックス 3 値を 0 に初期化</p> <p>チェックボックス 4 値を 0 に初期化</p> <p>チェックボックス 5 値を 0 に初期化</p>
--	---

<pre> } //end if start_flag=1;一度通過したら2度目を禁止 する } </pre>	
<pre> private: System::Void checkBox1_CheckedChanged(System::Object^ sender, System::EventArgs^ e) { hoji1=1; } private: System::Void checkBox2_CheckedChanged(System::Object^ sender, System::EventArgs^ e) { hoji2=1; } private: System::Void checkBox3_CheckedChanged(System::Object^ sender, System::EventArgs^ e) { hoji3=1; } private: System::Void checkBox4_CheckedChanged(System::Object^ sender, System::EventArgs^ e) { hoji4=1; } private: System::Void checkBox5_CheckedChanged(System::Object^ sender, System::EventArgs^ e) { hoji5=1; } </pre>	<p>チェックボックス 1 にレ点がついたら値を 1 にする</p> <p>チェックボックス 2 にレ点がついたら値を 1 にする</p> <p>チェックボックス 3 にレ点がついたら値を 1 にする</p> <p>チェックボックス 4 にレ点がついたら値を 1 にする</p> <p>チェックボックス 5 にレ点がついたら値を 1 にする</p>
<pre> private: System::Void button2_Click(System::Ob ject^ sender, System::EventArgs^ e) { if(start_flag==1){ int nohai; int xx=6; int x,y,w; char buf[80]; </pre>	<p>ボタン 2 を押した処理</p> <p>xx = 6</p> <p>チェックボク1にチェックが付いていない場合[カード交換となる]</p> <p>card(6) 配列6番目を1枚目とする</p>

<pre> /* テキストボックス入力を読み取るには、 String^ string1=textBox1->Text; テキストボックスに表示するには、 textBox1->Text=string1; */ String^ string1= "ブタです"; textBox1->Text=string1; if (hoji1==0){hai[1]=card[xx]; xx++;} if (hoji2==0){hai[2]=card[xx]; xx++;} if (hoji3==0){hai[3]=card[xx]; xx++;} if (hoji4==0){hai[4]=card[xx]; xx++;} if (hoji5==0){hai[5]=card[xx]; } for(xx=1;xx<6;xx++){ nohai=hai[xx]; DispHai(nohai,xx); } //end for //Rem フラッシュ int club = 0; int daiya = 0; int hart = 0; int supedo = 0; int flash = 0; for(x=1;x<6;x++){ if(hai[x] > 0 && hai[x] <= 13) {supedo = supedo + 1;} if(hai[x] > 100 && hai[x] <= 113) {club=club + 1;} if(hai[x] > 200 && hai[x] <= 213) {daiya = daiya + 1;} if(hai[x] > 300 && hai[x] <= 313) {hart = hart + 1;} } //Next x if(club == 5){ string1= "クラブのフラッシュ"; </pre>	<pre> x = 7 チェックボク2にチェックが付いていない場合[カ ード交換となる] card(7) 配列7番目を2枚目とする x = 8 チェックボク3にチェックが付いていない場合[カ ード交換となる] card(8) 配列8番目を3枚目とする x = 9 同様に 9 枚目 10 枚目 変更されたカードを表示 // 実行しないコメント 役解析 フラッシュの処理 色が同じか判断する 変数クラブ、ダイヤ、ハート、スペードを0とする 変数フラッシュを0とする xを1から5まで繰り返す hai(x)が0より大きく13以下の場合 スペードなので変数クラブを1増加 hai (x)が100より大きく113以下の場合クラブ なので変数を1増加する。 hai d(x)が200より大きく213以下の場合ダイヤ なので変数を1増加する。 hai (x)が300より大きく313以下の場合ハート なので変数を1増加する。 xを1から5まで繰り返し終了 もし変数クラブが5になっている場合5枚ともクラ </pre>
---	---

<pre> flash = 1; tokuten = tokuten + 509; } //End If club if (daiya == 5){ string1= "ダイヤのフラッシュ"; flash = 1; tokuten = tokuten + 509; } //End If daiya if(hart == 5){ string1= "ハートのフラッシュ"; flash = 1; tokuten = tokuten + 509; } //End If hart if(supedo == 5){ string1= "スペードのフラッシュ"; flash = 1; tokuten = tokuten + 509; } //End If supedo textBox1->Text=string1; //文字で数を確認 sprintf_s(buf,80,"supedo=%3d club=%3d daiy=%3d hart=%3d",supedo,club,daiya,hart); string1=gcnew String(buf); textBox5->Text=string1; //文字で確認終了 //Rem フラッシュ end //Rem ペアー int twopai = 0; int threepai = 0; int oitoku; for(x=6;x<11;x++){ </pre>	<p>ブとの判断でフラッシュとなる</p> <p>文字列1にクラブのフラッシュと代入する</p> <p>変数フラッシュを1とする。後でストレートとこの変数を比較しストレートフラッシュか判断するため</p> <p>フラッシュの確率は509のため得点に509を増加させる。</p> <p>もし変数ダイヤが5になっている場合5枚ともダイヤとの判断でフラッシュとなる</p> <p>文字列1にダイヤのフラッシュと表示する</p> <p>変数フラッシュを1とする。後でストレートとこの変数を比較しストレートフラッシュか判断するため</p> <p>フラッシュの確率は509のため得点に509を増加させる。</p> <p>もし変数ハートが5になっている場合5枚ともハートとの判断でフラッシュとなる</p> <p>文字列1にハートのフラッシュと表示する</p> <p>変数フラッシュを1とする。後でストレートとこの変数を比較しストレートフラッシュか判断するため</p> <p>フラッシュの確率は509のため得点に509を増加させる。</p> <p>もし変数スペードが5になっている場合5枚ともクラブとの判断でフラッシュとなる</p> <p>文字列1にスペードのフラッシュと表変数フラッシュを1とする。後でストレートとこの変数を比較しストレートフラッシュか判断するため</p> <p>フラッシュの確率は509のため得点に509を増加させる。</p> <p>テキストボックス 5 に文字列フラッシュを表示</p>
---	--

<pre> keta[x] = 0; } //Next x for(x=1;x<6;x++){ y=hai[x]; if(hai[x]>100) y=hai[x]-100; if(hai[x]>200) y=hai[x]-200; if(hai[x]>300) y=hai[x]-300; keta[x]=y; } //Next x for(x=1;x<6;x++){ yaku[x] = 0; for(w=1;w<6;w++){ if(keta[x] == keta[w]) yaku[x] = yaku[x] + 1; } //Next w } //Next x for(x=1;x<6;x++){ if(yaku[x]== 2) twopai = twopai + 1; if(yaku[x]== 3) threepai = 1; if(yaku[x]== 4){ string1 = "フォーカード"; tokuten = tokuten + 4165; } //End if(yaku[x]== 4) } //Next x if(twopai == 2){ string1 = "ワンペア"; tokuten = tokuten + 3; } //End If(twopai == 2) if(twopai == 4){ string1 = "ツーペア"; tokuten = tokuten + 21; } //End If(twopai == 4) if(threepai == 1){ string1 = "スリーカード"; </pre>	<p>コメント ペア</p> <p>ツーペア変数を0に</p> <p>スリカード変数を0に</p> <p>初期化 ストレートのための</p> <p>5枚のカードを順次読む</p> <p>x が1から5まで繰り返す</p> <p>yに順次5枚のカードを入れていく</p> <p>ペアの確認をするため</p> <p>下一桁にする</p> <p>100より大きい場合100を引く</p> <p>200より大きい場合200を引く</p> <p>300より大きい場合300を引く</p> <p>繰り返しの終了</p> <p>解析開始</p> <p>同じカードがないか判断する</p> <p>x が1から5まで繰り返す</p> <p>5個の配列 yaku(x)を0にする</p> <p>wが1から5まで繰り返す</p> <p>同じカードが有れば yaku(x)を1増加する</p> <p>繰り返し w の終了</p> <p>繰り返し x の終了</p> <p>?x = 1 から 5 繰り返し</p> <p>yaku(x)が2の場合同じカードが2枚あったので</p> <p>ツーペア変数を1増加</p> <p>yaku(x)が3の場合同じカードが3枚あったので</p> <p>スリカード変数を1にして通知</p> <p>yaku(x)が4の場合同じカードが4枚あったので</p> <p>文字列1にフォーカードを代入</p> <p>フォーカードなので得点に4165点を足す</p> <p>If yaku(x) = 4 終了</p> <p>x = 1 から 5 繰り返し終了</p> <p>ツーペア変数が2の場合</p> <p>文字列1にワンペアを代入</p> <p>ワンペアなので得点に3点を足す</p>
--	---

<pre> tokuten = tokuten + 47; } //End If(threepai == 1) if(twopai == 2 && threepai == 1){ string1 = "フルハウス"; tokuten = tokuten + 694; } //End If textBox1->Text=string1; //Rem ペアーend //Rem ストレート for(x=1;x<6;x++){ for(y=1;y<6;y++){ if(keta[x] < keta[y]){ oitoku = keta[x]; keta[x] = keta[y]; keta[y] = oitoku; } //End If } //Next y } //Next x if (keta[1] == 1){ for(x=1;x<6;x++){ if(keta[x] = x) keta[x + 5] = keta[x] + 13; } // Next x } //End if (keta[1] == 1) int flag = 0; or(x=1;x<10;x++){ if (keta[x] == keta[x + 1] - 1) flag = flag + 1; } //Next x if(flag >= 4){ string1 = string1 + " ストレート"; tokuten = tokuten + 255; </pre>	<pre> If twopai = 2 Then の終了 ツーパー変数が4の場合 文字列1にツーパーを代入 ツーパーなので得点に21点を足す If twopai =4 の終了 スリカード変数が1の場合 文字列1にスリーカードを代入 スリーカードなので得点に47点を足す If threepai = 1 の終了 ツーパー変数が2でかつスリカード変数が1の 場合 文字列1にフルハウスを代入 フルハウスなので得点に694点を足す If twopai = 2 And threepai = 1 終了 テキストボックス 1 に文字列表示 コメント ストレート 小さい順に並び替えをする For x = 1 から 5 繰り返し For y = 1 から 5 繰り返し 小さいか比較 オイトク変数に小さいカードを入れる 大き方に小さいものを代入 小さい方にオイトク変数を代入 比較終了 For y5 繰り返し終わり For x 5 繰り返し終わり ストレートで1がある場合の処理 最初のカードに1がある場合 セル6行目に14として収納 繰り返すことにより 次の2はセル7行目に15として収納 旗変数を0とする </pre>
--	---

<pre> if(flash == 1) tokuten = tokuten + 72193 - 255; if(flash == 1 && flag >= 4 && keta[1] == 1 && keta[5] == 13){ string1 = "ローヤルストレートフラッシュ"; tokuten = tokuten + 649740; } //End If //TextBox2 = tokuten } //End If textBox1->Text=string1; //Rem ストレート end textBox2->Text=tokuten.ToString();//数値から文 字列への変換には、 //textBox2->Text=string2; start_flag=2; } //end if start_flag==1 } </pre>	<p>1がある場合5枚ではなく追加された4枚も見るため9回繰り返す</p> <p>今のカードと次のカードが連番になっている場合</p> <p>旗変数を1増加する</p> <p>繰り返し終了</p> <p>旗変数が4以上の場合はストレートとなる</p> <p>文字列1にストレートと代入する</p> <p>得点を255点増加させる</p> <p>役としてすでにフラッシュがある場合</p> <p>得点に72193-255を増加させる</p> <p>ストレートフラッシュで最初1になっていて5番目が13の時</p> <p>文字列1にローヤルストレートフラッシュと表示する</p> <p>得点に649740点を増加する</p> <p>テキストボックス2に得点を表示</p>
<pre> private: System::Void button3_Click(System::Object^ sender, System::EventArgs^ e) { if (start_flag==2){ int nohai; int xx; for(xx=1;xx<6;xx++){ nohai=0; DispHai(nohai,xx); } } </pre>	<p>カードの裏を表示して</p> <p>旗変数を初期化しボタン1を押せるようにする</p>

<pre>start_flag=0; } }</pre>	
<pre>private: System::Void button4_Click(System::Object^ sender, System::EventArgs^ e) { card[6]=1; card[7]=2; card[8]=3; card[9]=212; card[10]=313; }</pre>	<p>確認ルーチン 好きな数を入れる</p> <p>4枚目のカードをクラブの12にしてみる</p> <p>5枚目のカードをスペードのキングにしてみる</p>

