```
#include <iostream>
#include <random>

using namespace std;

//doppio array per il mazzo
int mazzov[32];
char mazzos[32];
//doppio array per il giocatore
int giocv[5];
char giocs[5];
//doppio array per la CPU
int cpuv[5];
char cpus[5];
//

void initProva(){
	for(int i=0; i<10; i++){
			mazzov[i]=11;
	}
	for(int i=10; i<20; i++){
			mazzov[i]=10;
	}
	for(int i=20; i<32; i++){
			mazzov[i]=12;
	}
}

void initValori(){
	int j=7;
	int count=0;
	for(int i=0; i<32; i++){
```

```
                while(count<4){
                        mazzov[i]=j;
                        count++;
                        if (count!=4){
                                i++;
                        }
                }
                j++;
                count=0;
        }
}

void initSemiProva(){
        for(int i=0; i<15; i++){
                        mazzos[i]='C';
        }
        for(int i=15; i<32; i++){
                        mazzos[i]='Q';
        }
}

void initSemi(){
        for(int i=0; i<32; i=i+4){
                        mazzos[i]='C';
                        mazzos[i+1]='Q';
                        mazzos[i+2]='F';
                        mazzos[i+3]='P';
                }
}

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

```
                cout<<mazzov[i];
                cout<<mazzos[i]<<" ";
        }
}

void vediCarteGiocatore(){
        for(int i=0; i<5; i++){
                cout<<giocv[i];
                cout<<giocs[i]<<" ";
        }
}

void vediCarteCPU(){
        for(int i=0; i<5; i++){
                cout<<cpuv[i];
                cout<<cpus[i]<<" ";
        }
}

int estraiNum(){
        random_device rd;
        mt19937 gen(rd());
        int min=0;
        int max=31;
        uniform_int_distribution<int> dist(min, max);
        int numero = dist(gen);
        return numero;
}

void estraiCarteGioc(){
        for(int i=0; i<5; i++){
                int n = estraiNum();
```

```
                if(mazzov[n]!=-1){
                        giocv[i]=mazzov[n];
                        giocs[i]=mazzos[n];
                        mazzov[n]=-1;
                }else{
                        //cout<<" *** carta doppia! ***"<<endl;
                        i--;
                }
        }
}

void estraiCarteCPU(){
        for(int i=0; i<5; i++){
                int n = estraiNum();
                if(mazzov[n]!=-1){
                        cpuv[i]=mazzov[n];
                        cpus[i]=mazzos[n];
                        mazzov[n]=-1;
                }else{
                        //cout<<" *** carta doppia! ***"<<endl;
                        i--;
                }
        }
}

void carteGiocSuTavolo(){
        cout<<"******* ******* ********** ******* *******"<<endl;
        cout<<"*   * *   * *   * *   * *   *"<<endl;
        cout<<"*   * *   * *   * *   * *   *"<<endl;
        for(int i=0; i<5; i++){
        cout<<"* "<<giocv[i]<<giocs[i];
                        if(giocv[i]<10)cout<<"  * ";else cout<<" * ";
```

```
	}
	cout<<endl;
	cout<<"*   * *   * *   * *   * *   *"<<endl;
	cout<<"*   * *   * *   * *   * *   *"<<endl;
	cout<<"******* ******* ********** *******"<<endl;
}

void carteCPUSuTavolo(){
	cout<<"******* ******* ********** *******"<<endl;
	cout<<"*   * *   * *   * *   * *   *"<<endl;
	cout<<"*   * *   * *   * *   * *   *"<<endl;
	for(int i=0; i<5; i++){
	cout<<"* "<<cpuv[i]<<cpus[i];
			if(cpuv[i]<10)cout<<"  * ";else cout<<" * ";
	}
	cout<<endl;
	cout<<"*   * *   * *   * *   * *   *"<<endl;
	cout<<"*   * *   * *   * *   * *   *"<<endl;
	cout<<"******* ******* ********** *******"<<endl;
}

int verificaOccorenze(int carte[5], int trova){
	int cont=0;
	for(int i=0; i<5; i++){
		if(carte[i]==trova){
			cont++;
		}
	}
	return cont;
}

char verificaColore(char semi[5]){
```

```
	if (semi[0]==semi[1]&&semi[1]==semi[2]&&semi[2]==semi[3]&&semi[3]==semi[4]){
		return semi[0];
	}
	return 'n';
}

void verificaCoppie(int carte[5], int punt[2]){
	punt[0] = 0;
	punt[1] = 0;
	int j=0;
	for (int i=7; i<15; i++){
		if (verificaOccorenze(carte, i)==2)
			{
				punt[j]=i;
				j++;
			}
	}
}

int verificaTris(int carte[5]){
	for (int i=7; i<15; i++){
		if (verificaOccorenze(carte, i)==3)
			{
				return i;
			}
	}
	return 0;
}

int verificaFull(int carte[5]){
	int coppie[2];
	verificaCoppie(carte, coppie);
```

```
	//se è stata trovata una coppia verifico anche il tris
	if(coppie[0]!=0 && coppie[1]==0){
		int full =verificaTris(carte);
		if(full!=0){
			//se viene trovato un tris restituisco la carta del tris
			return full;
		}
		else{
			//se non viene trovato il tris restituisco 0 (niente full)
			return 0;
		}
	}
	//se non è stata trovata la coppia ritorno 0 (niente full)
	return 0;
}

int verificaPoker(int carte[5]){
	for (int i=7; i<15; i++){
		if (verificaOccorenze(carte, i)==4)
			{
				return i;
			}
	}
	return 0;
}

int main(){
	//initValori();
	initProva();
	//initSemi();
	initSemiProva();
	//cout<<"*** carte del mazzo ***"<<endl;
```

```
//vediMazzo();
estraiCarteGioc();
cout<<"*** carte giocatore ***"<<endl;
//vediCarteGiocatore();
carteGiocSuTavolo();
estraiCarteCPU();
cout<<"*** carte CPU ***"<<endl;
//vediCarteCPU();
carteCPUSuTavolo();
//verifica coppie giocatore
int coppie[2];
verificaCoppie(giocv, coppie);
if(coppie[0]!=0 && coppie[1]!=0){
        cout<<"G *** Doppia Coppia: "
        <<coppie[0]<<" - "<<coppie[1]<<endl;
}
if(coppie[0]!=0 && coppie[1]==0){
        cout<<"G *** Coppia: "<<coppie[0]<<endl;
}
//verifica coppie CPU
verificaCoppie(cpuv, coppie);
if(coppie[0]!=0 && coppie[1]!=0){
        cout<<"CPU *** Doppia Coppia: "
        <<coppie[0]<<" - "<<coppie[1]<<endl;
}
if(coppie[0]!=0 && coppie[1]==0){
        cout<<"CPU *** Coppia: "<<coppie[0]<<endl;
}
//verifica Tris giocatore
int tris=verificaTris(giocv);
if (tris!=0){
        cout<<"G *** Tris di "<<tris<<endl;
```

```
}
//verifica Tris  CPU
tris=verificaTris(cpuv);
if (tris!=0){
	cout<<"CPU *** Tris di "<<tris<<endl;
}
//verifica Poker giocatore
int poker=verificaPoker(giocv);
if (poker!=0){
	cout<<"G *** Poker di "<<poker<<endl;
}
//verifica Poker CPU
poker=verificaPoker(cpuv);
if (poker!=0){
	cout<<"CPU *** Poker di "<<poker<<endl;
}
//verifica Full giocatore
int full=verificaFull(giocv);
if(full!=0){
	cout<<"G *** Full di "<<full<<endl;
}
//verifica Full CPU
full=verificaFull(cpuv);
if(full!=0){
	cout<<"CPU *** Full di "<<full<<endl;
}
//verifica colore giocatore
char colore = verificaColore(giocs);
if (colore!='n'){
	cout<<"G *** Colore di "<<colore<<endl;
}
//verifica colore CPU
```

```
        colore = verificaColore(cpus);
        if (colore!='n'){
                cout<<"CPU *** Colore di "<<colore<<endl;
        }
}
```