

Universidad Tecnológica De la Mixteca



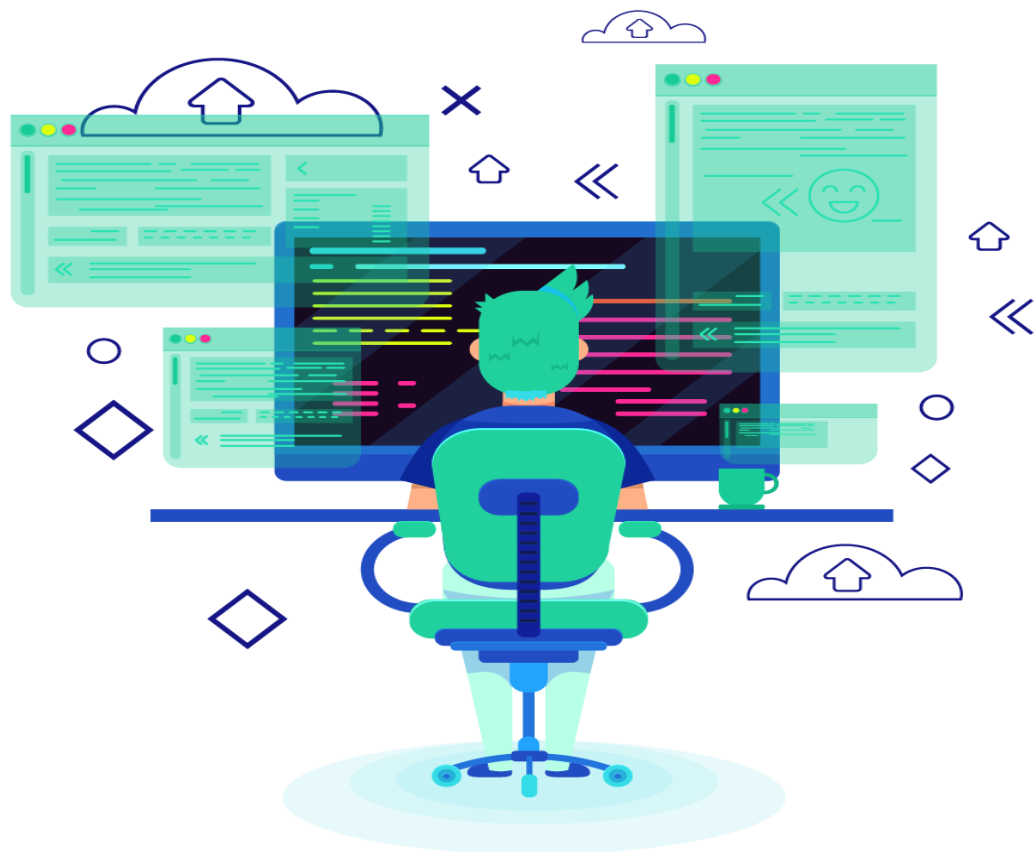
Programación Estructurada

Alumno: Leonel Santiago Rosas

Grupo: 102-A

Catedrático: Dr. Christian Eduardo Millán
Hernandez

Ejercicios Deitel cap 7: 7.12



Contenido

Problema:.....	2
introducción:.....	2
Código:	2
Bibliografía:	5
Conclusiones	5
Pruebas:	5
Código con posible solución: (no terminado)	7
Pruebas eje 2:	9

Problema:

problemas, deberá estar en condiciones de resolverlos con justicia.

7.12 Modifique el programa de la figura 7.24, de tal forma que la función de distribución de naipes reparta una mano de póker de cinco naipes. A continuación escriba las funciones adicionales siguientes:

- Determine si la mano contiene un par.
- Determine si la mano contiene dos pares.
- Determine si la mano contiene tres de un tipo (es decir, por ejemplo, tres jacks).
- Determine si la mano contiene cuatro de un tipo (por ejemplo, cuatro ases).
- Determine si la mano contiene un color (es decir, todos los cinco naipes del mismo palo).
- Determine si la mano contiene una flor imperial (es decir, cinco naipes de valores nominales consecutivos).

Utilice el ejercicio 7.12 para escribir un programa que distribuya

introducción:

El problema es bastante complejo, pero creo que todo con programación se puede hacer, a un que no llegue a probar realmente si hay una forma de sacar una flor imperial, muchos intentos y ninguna vez me salió, el problema consistía en analizar las posiciones, de igual manera con el full color, no conseguí un resultado ya que las probabilidades de ambas son realmente bajas.

Código:

```
#include<stdio.h>
#include<stdlib.h>
#include<time.h>
void shuffle(int[][13]);
void deal(const int[][13], const char *[], const char*[]);
int Analizar(int xcartas[], int scartas[]);
int pares=0, terci=0;

int main()
{
    srand(time(NULL));
    const char *suit[4]={"Corazones", "Diamantes", "Trebol", "Espadas"};
    const char *face[13]={"AS", "Dos", "Tres", "Cuatro", "Cinco", "Seis", "Siete",
    "Ocho"}
```

```

        , "Nueve", "Diez", "Jack", "Reyna", "Rey"};

//int xF[53] = {0};
//int xN[53] = { 0 };

int deck[4][13]= {0};

shuffle(deck);
deal(deck, face, suit);
return 0;
}

void shuffle(int wDeck[][13])
{
    int card, row, column;

    for (card = 1; card <=52; card++)
    {
        row = rand() % 4;
        column = rand() % 13;

        while (wDeck[row][column]!=0)
        {
            row = rand () % 4;
            column = rand() %13;
        }
        wDeck[row][column]=card;
    }
}

void deal(const int wDeck[][13], const char *wFace[], const char *wSuit[])
{
    int card, row, column;
    int *cartas;
    int n[13];
    int s[13];
    //int *tmp2[4][13]={0};

    for ( card = 1; card <= 5; card++)
    {
        for ( row = 0; row<=3; row++)
        {
            for ( column = 0; column <= 12; column++)
            {
                if (wDeck[row][column]==card)

```

```

        {
            printf("\n%5s de %s", wFace[column], wSuit[row]);
            n[card] = (int)wFace[column];
            s[card] = (int)wSuit[row];
        }
    }
}

}
//cartas = n;
Analizar(n,s);
//tmp2 = wDeck[row][column];
//Analizar(tmp2);
}

int Analizar( int xcartas[],int scartas[])
{
    int rojas=0, negras=0;
    int poker=0;
    int flor=0;
    int temp2[14]={0};
    int *cards; //int Num[15];
    //int num2[52]={0};
    for (int i = 1; i <=5; i++)
    {
        for (int j = 1; j <= 5; j++)
        {
            if (xcartas[i]==*(xcartas+j) || xcartas[j]==*(xcartas+i))
            {
                if (pares==10)
                {
                    poker++;
                    //pares = 0;
                }
                if (pares==8)
                {
                    tercia++;
                    flor++;
                }
                pares++;
            }
        }
    }
}

```

```

for (int i = 0; i <=3; i++)
{
    if (scartas[i]==scartas[0])
    {
        rojas++;
        flor++;
    }
}
printf("\n\n\nTienes pares: %d", (pares-5)/2);
printf("\nTercias: %d", tertia);
printf("\nPokers: %d", poker);
printf("\nFull color: %d", rojas/4);
printf("\nFlo imperial : %d", flor=0);
}

```

Bibliografía:

Elvex U. Decsia 2012, Punteros y sus funciones

Conclusiones

A un que encontré un método que realmente resuelve los métodos a base de sus funciones logre hallar un segundo método donde analizo las cadenas de caracteres, las pruebas marcan que es mas confiable pero al trabajar poco tiempo en el proyecto no me dio tiempo a terminar el problema con ese método, pero con las posiciones y analizando cada una de ellas si.

Pruebas:

```

Cuatro de Trebol
Rey de Diamantes
Tres de Espadas
Cinco de Trebol
Tienes pares: 0
Tercias: 0
PS C:\Users\Leonel\Documents\UTM\S
ProgramacionEstructurada\codigos\E

```

```

AS de Corazones
Jack de Trebol
Seis de Trebol
Tres de Trebol
Tres de Diamantes
Tienes pares: 1
Tercias: 0

```

```

Siete de Corazones
Reyna de Diamantes
Siete de Diamantes
Cuatro de Diamantes
Siete de Espadas
Tienes pares: 1
Tercias: 1
PS C:\Users\Leonel\Do

```

Realmente solo me llego a tocar pares
y tercias.

```
Full color: 0
Flo imperial : 0
PS C:\Users\Leonel\Documents\UTM\Semestre1\ProgramacionEstructurada\codigos\EjerciciosDeDietel\Ejercicios7_0> cd "
urada\codigos\EjerciciosDeDietel\Ejercicios7_0\" ; if ($?) { gcc 7_17V2.c -o 7_17V2 } ; if ($?) { .\7_17V2 }

Nueve de Espadas
Rey de Diamantes
Nueve de Trebol
Seis de Espadas
Tres de Corazones

Tienes pares: 1
Tercias: 0
Pokers: 0
Full color: 0
Flo imperial : 0
PS C:\Users\Leonel\Documents\UTM\Semestre1\ProgramacionEstructurada\codigos\EjerciciosDeDietel\Ejercicios7_0> cd "
urada\codigos\EjerciciosDeDietel\Ejercicios7_0\" ; if ($?) { gcc 7_17V2.c -o 7_17V2 } ; if ($?) { .\7_17V2 }

Seis de Diamantes
Tres de Diamantes
Tres de Trebol
Rey de Diamantes
Seis de Espadas

Tienes pares: 2
Tercias: 1
Pokers: 0
Full color: 0
Flo imperial : 0
PS C:\Users\Leonel\Documents\UTM\Semestre1\ProgramacionEstructurada\codigos\EjerciciosDeDietel\Ejercicios7_0> █
```

Código con posible solución: (no terminado)

```
#include<stdio.h>
#include<stdlib.h>
#include<time.h>
void shuffle(int[][13]);
void deal(const int[][13], const char *[], const char *[]);
void Analizar(const char *[]);
int pares=0, tercia=0;

int main()
{
    srand(time(NULL));
    const char *suit[4]={"Corazones", "Diamantes", "Trebol", "Espadas"};
    const char *face[13]={"AS", "Dos", "Tres", "Cuatro","Cinco", "Seis", "Siete",
    "Ocho",
    "Nueve", "Diez", "Jack", "Reyna", "Rey"};

    //int xF[53] = {0};
    //int xN[53] = { 0 };

    int deck[4][13]= {0};

    shuffle(deck);
    deal(deck, face, suit);
    return 0;
}

void shuffle(int wDeck[][13])
{
    int card, row, column;

    for (card = 1; card <=52; card++)
    {
        row = rand() % 4;
        column = rand() % 13;

        while (wDeck[row][column]!=0)
        {
            row = rand () % 4;
            column = rand() %13;
        }
        wDeck[row][column]=card;
    }
}
```



```

}

void deal(const int wDeck[][13], const char *wFace[], const char *wSuit[])
{
    int card, row, column;
    int *cartas;
    const char *ana[13];

    for ( card = 1; card <= 5; card++)
    {
        for ( row = 0; row<=3; row++)
        {
            for ( column = 0; column <= 12; column++)
            {
                if (wDeck[row][column]==card)
                {
                    printf("\n%5s de %s", wFace[column], wSuit[row]);
                    ana[card] = wFace[column];
                }
            }
        }

    }
    Analizar(ana);
}

void Analizar(const char *xcartas[])
{
    for (int i = 1; i <=5; i++)
    {
        for (int j=i+1; j <=5; j++){
            if (xcartas[i]==xcartas[j])
            {
                pares++;
                if (pares == 3){
                    tercia++;
                    pares = 0;
                }
                //printf("\n%s", xcartas[i]);
            }
        }
    }

    if (pares == 3)

```

```

{
    printf("\nTienes pares: %d", 1);
}
else {

printf("\nTienes pares: %d", pares);}
printf("\nTercias: %d", tercia);
}

```

Pruebas eje 2:

martinsantiago@MacBook-Pro-de-Martin Downloads % ./a.out

```

Jack de Corazones
Ocho de Trebol
Tres de Diamantes
Diez de Corazones
Ocho de Espadas
Ocho

```

Tienes pares: 1%

martinsantiago@MacBook-Pro-de-Martin Downloads % ./a.out

```

Dos de Espadas
Ocho de Corazones
AS de Espadas
Rey de Corazones
Siete de Espadas

```

Tienes pares: 0%

martinsantiago@MacBook-Pro-de-Martin Downloads % ./a.out

```

Nueve de Trebol
Cinco de Trebol
Dos de Trebol
Seis de Corazones
Seis de Trebol
Seis

```

Tienes pares: 1%

martinsantiago@MacBook-Pro-de-Martin Downloads % ./a.out

```

Rey de Diamantes
Cinco de Espadas
Rey de Corazones
Reyna de Corazones
Cinco de Trebol
Rey
Cinco

```

Tienes pares: 2%

martinsantiago@MacBook-Pro-de-Martin Downloads % ./a.out