UNIVERSIDAD DE PANAMÁ

CENTRO REGIONAL UNIVERSITARIO DE VERAGUAS

FACULTAD DE INFORMÁTICA ELECTRÓNICA Y COMUNICACIÓN

LICENCIATURA PARA LA GESTION EDUCATIVA Y EMPRESARIAL

PROGRAMACION II

PROYECTO

ESTADÍSTICA DE UNA ELECCION

ESTUDIANTES:

RODRIGO GUTIERREZ

JESAID TORIBIO

PROFESORA:

GIANNINA NÚÑEZ

2018

1. Tareas:
   1. Elaborar el diagrama HIPO.  Considere los conceptos de programación modular para especificar la tabla IPO de cada módulo.

cargamatriz

|  |  |  |
| --- | --- | --- |
| Entrada | Proceso | Salida |
| Se le especifica la cantidad de votantes, se le pasa la lista ya creada de Cedula, Nombre, Centro de votación y la mesa. | Se hace un ciclo para cargar una matriz y se introducen los datos: Cedula, Nombre, Centro de votación y la mesa de cada votante. | Matriz con los datos cargados. |

imprecionmatriz

|  |  |  |
| --- | --- | --- |
| Entrada | Proceso | Salida |
| Se le dará la matriz ya creada, la cantidad de filas y columnas.  a=matriz  b=cantidad de filas (votantes).  c=cantidad columnas (datos). | Consta de dos ciclos para imprimir una matriz.  Una será para la cantidad de fila (votantes) y otra para las columnas (datos). | Impresión de la matriz que se le indique |

creanomb

|  |  |  |
| --- | --- | --- |
| Entrada | Proceso | Salida |
| Se le da la lista en memoria de los nombres y la lista de apellidos, también se le especifica la cantidad de nombres que se crearan(variable cantidad) | Se generan 2 índices aleatoriamente entre 0 y la cantidad de la listas de nombre y apellido, estos números generados se utilizan como índice en cada lista y se hace una suma de cadenas que da como resultado un nombre. | Lista de combinación de nombres y apellidos aleatoriamente. |

cedula

|  |  |  |
| --- | --- | --- |
| Entrada | Proceso | Salida |
| Se le da la cantidad que se realizara el ciclo(cantidad de personas) | Se utilizan tres variables donde ce generara el número de provincia y los siguientes dígitos, estos se convertirán en cadena, se sumaran las tres cadenas y se creara una lista de cedulas. | Lista de cedula, tipo de lista cadena. |

centrovot

|  |  |  |
| --- | --- | --- |
| Entrada | Proceso | Salida |
| a=cantidad de votantes.  b=lista en memoria con los centro de votación. | Se generara un numero entre 0 y la cantidad valores en la lista y se verifica en la lista que está cargada en memoria si es igual al texto del ciclo y se agrega en la lista. | Lista de cadena con centro de votación para cada votante. |

mesa

|  |  |  |
| --- | --- | --- |
| Entrada | Proceso | Salida |
| Se le especifica la cantidad de personas y la lista de los centro de votación creada con la función anterior  a=cantidad de personas.  b=lista de centro de votación creada. | Dependiendo el centro de votación que se le asigno le toca x mesa. Y se agregara a una lista. | Lista de número que simbolizan las mesa que le toco a cada votante. |

verifica

|  |  |  |
| --- | --- | --- |
| Entrada | Proceso | Salida |
| Se le pasa la lista ind, el contador i y el valor a verificar | Se le pasa un valor y si no está en la lista devuelve 0 y si esta 1.  Si devuelve 0 el valor generado (índice) se agrega a una lista.  Esta función se utiliza para genera x cantidad de numero pero que no se repitan | Devuelve un valor buleano 0/1 |

cargamatrizdevotantes

|  |  |  |
| --- | --- | --- |
| Entrada | Proceso | Salida |
| Se le da la cantidad de personas(a), el padrón electoral(b) y la lista de índice creada anteriormente(c) | Se genera una matriz donde se agregara los datos de los votantes pero solo de los que se generaron aleatoriamente en la lista (ind) | Una nueva matriz con los votantes seleccionados. |

genevotos

|  |  |  |
| --- | --- | --- |
| Entrada | Proceso | Salida |
| Se llama la función cuando se necesite no se le pasa argumentos | Genera un valor entre 0 y 4 que es la cantidad de candidatos. | Un valor entero  Entre 0 y 4 |

concentro

|  |  |  |
| --- | --- | --- |
| Entrada | Proceso | Salida |
| Se le especifica la cantidad que asistieran a las urnas, el padrón de votantes, lo que se quiere contar (cadenas o números) y la columna en la que se ubica. | Se hace un ciclo y se verifica si el argumento que se pasó está en la columna especificada se cuenta. | El resultado del contador |

contmesa

|  |  |  |
| --- | --- | --- |
| Entrada | Proceso | Salida |
| Se le asigna los argumentos (cantidad, lista con los votos y el valor que representa daca candidato) | Recorre la lista y cuenta los cuantos valores son iguales al argumento que se les paso. | El resultado del contador |

canditogana

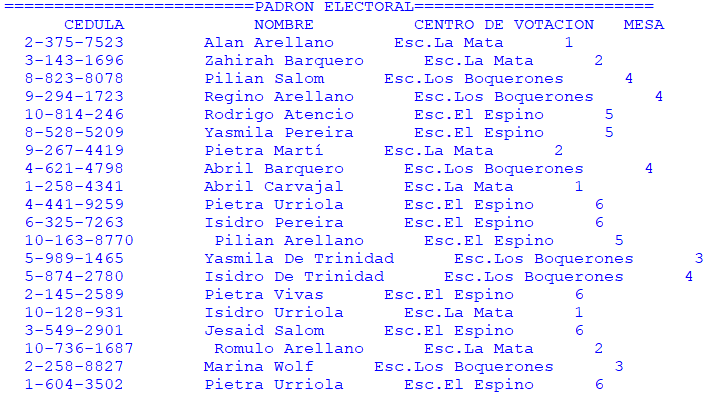
|  |  |  |
| --- | --- | --- |
| Entrada | Proceso | Salida |
| Se le pasa el argumento de la lista donde están los votos totales de cada candidato. | Un ciclo que busca el mayor en la lista y guarda la posición del mismo. | La posición de la lista que tiene un mayor valor |

porcentaje

|  |  |  |
| --- | --- | --- |
| Entrada | Proceso | Salida |
| La cantidad de voto de cada representare y el total que votaron | Una regla de tres para saber a qué porcentaje equivale la cantidad de voto para cada representare y voto en blanco. | Porcentaje de los votos. |

* 1. Justificar las estructuras de datos (vectores y matrices) a emplear para almacenar los diferentes tipos de datos y presentar los resultados.

Padrón electora



Consta de 4 columnas donde se almacenará la cedula, el nombre, centro de votación y mesa.

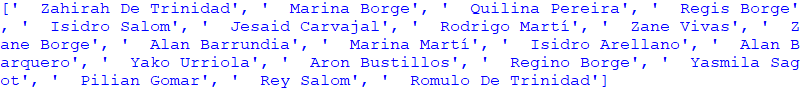
Utilizamos esta forma porque en la guía se nos dio un ejemplo.

Cedula



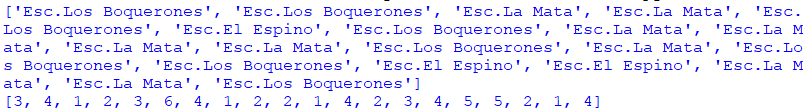
Se generó la cedula en un lista para luego ser introducida al padrón electoral.

Nombre y apellido



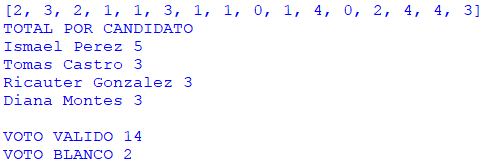
Se muestra una lista creada con su respectiva función con el nombre y apellido.

Centro de votación y mesa



Cada centro de votación tiene 2 mesas y a cada persona le toca un centro de votación, dependiendo de qué centro le toque, se le asignara un número de mesa.

Resultados de votos



El valor del voto es el siguiente

1=primer candidato en la lista

2=segundo candidato en la lista

3=tercer candidato en la lista

4=cuarto candidato en la lista

Pasa saber cuántos votos le toco a cada candidato se utilizó la función creada para contar en una lista y se especificó el número de cada candidato.

Votos en blanco se usó la función de contar y tienen el valor de 0.

Para los votos válidos se restó del total de votos emitidos – votos en blanco.

* 1. Describa cada una de las variables de su programa en una tabla

|  |  |  |
| --- | --- | --- |
| Nombre | Descripción/Utilidad. | Tipo |
| nomb | Lista de nombres. | Lista |
| apell | Lista de apellidos. | Lista |
| centro | Nombre de los centro de votación. | Lista |
| repre | Nombres de los candidatos a representantes. | Lista |
| cantidad | Cantidad de personas que se generan para el padrón. | Entero |
| votrandom | Un valor random que serán la cantidad que votaran. | Entero |
| b | Lista de cedulas/Conforman parte del padrón | Lista |
| a | Guarda una lista creada de la combinación de nombres y apellidos. | Lista |
| c | Guarda una lista centros de votación aleatoriamente. | Lista |
| d | Almacena una lista de enteros que conforman que mesa le toca a cada votante | Lista |
| padron | Guarda un matriz creada con la función cargamateriz /Padrón electoral | Lista |
| indi | Lista con números aleatorios del padrón que serán los que asistirán a las urnas/ Se utilizan como índice para seleccionar del padrón electoral. | Lista |
| i | Controla el ciclo donde se almacenan el índice para generar la matriz de las personas que asistirán a los centros de votación. | Contador |
| padronvotaron | Guarda una nueva matriz con los votantes cargados. | Matriz |
| totalcen1 | Total de votos emitidos en centro de votación “Esc. La mata”. | Entero |
| totalcen2 | Total de votos emitidos en centro de votación “Esc. Los Boquerones”. | Entero |
| totalcen3 | Total de votos emitidos en centro de votación “Esc. El Espino”. | Entero |
| totalmesa1 | Almacena el total de votos emitido por mesa de cada centro de votación. | Entero |
| totalmesa2 | Almacena el total de votos emitido por mesa de cada centro de votación. | Entero |
| totalmesa3 | Almacena el total de votos emitido por mesa de cada centro de votación. | Entero |
| totalmesa4 | Almacena el total de votos emitido por mesa de cada centro de votación. | Entero |
| totalmesa5 | Almacena el total de votos emitido por mesa de cada centro de votación. | Entero |
| totalmesa6 | Almacena el total de votos emitido por mesa de cada centro de votación. | Entero |
| votosrepre | Almacena los votos generados por los que asistieron a las urnas | Lista |
| cand1 | Total de votos del candidato 1 | Entero |
| cand2 | Total de votos del candidato 2 | Entero |
| cand3 | Total de votos del candidato 3 | Entero |
| cand4 | Total de votos del candidato 4 | Entero |
| nulo | Almacena el total de los votos en blanco. | Entero |
| vototal | Se almacena el total de votos por candidatos | Lista |
| indrepre | Almacena el índice del que obtuvo mayor votos | Entero |
| a1 | Almacena el porcentaje de votos que obtuvo cada candidato | Flotante |
| a2 | Almacena el porcentaje de votos que obtuvo cada candidato | Flotante |
| a3 | Almacena el porcentaje de votos que obtuvo cada candidato | Flotante |
| a4 | Almacena el porcentaje de votos que obtuvo cada candidato | Flotante |
| blanco | Porcentaje de votos en blanco | Flotante |

* 1. Presente y explique el diseño de su entrada:
* Los datos personales se generan aleatoriamente.
* De igual manera el centro de votación y la mesa se generan aleatoriamente.
* El voto de cada persona que asista a las urnas será aleatoria mente.
* En resumen el usuario no tendrá que ingresar ningún dato.
* En un caso especial se pude modificar el código para que usuario introduzca la población de x corregimiento.
  1. Presente y explique el diseño de su salida:
     + Se desplegara un informe con cada uno de los puntos tales como:
       1. Total del corregimiento.
       2. El total de votos emitidos por centro de votación.
       3. El total de votos emitidos por mesa.
       4. Total por candidato.
       5. Voto válido y voto en blanco.
       6. Candidato ganador.
       7. El porcentaje de los votos.
  2. Presente su programa fuente debidamente compilado y documentado.

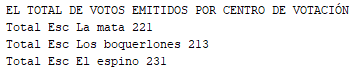
**import** random  
nomb=[**"Alan"**,**"Aron"**,**"Pietra"**,**"Pilian"**,**"Quilina"**,**"Rey"**,**"Regis"**,**"Regino"**,**"Romulo"**,**"Yasmila"**,**"Yako"**,**"Zane"**,**"Zahirah"**,**"Abril"**,**"Rodrigo"**,**"Jose"**,**"Marina"**,**"Jesaid"**,**"Isidro"**]*#NombresEnMemoria*apell=[**"Borge"**,**"Arellano"**,**"De Trinidad"**,**"Gomar"**,**"Carvajal"**,**"Martí"**,**"Sagot"**,**"Wolf"**,**"Salom"**,**"Vivas"**,**"Barrundia"**,**"Barquero"**,**"Bustillos"**,**"Pereira"**,**"Atencio"**,**"Urriola"**]*#ApellidoEnMemoria*centro=[**"Esc.La Mata"**,**"Esc.Los Boquerones"**,**"Esc.El Espino"**]*#Lista de los centro de votacion*repre=[**"Ismael Perez"**,**"Tomas Castro"**,**"Ricauter Gonzalez"**,**"Diana Montes"**]*#Candidatos a representante*cantidad=2000*#Es numero votantes de la poblacion del corregimiento*votrandom=random.randint(1,(cantidad/2))*#Indica la cantidad aleatoria de personas que asistiran a las urnas***def** cargamatriz(f, a, b, c, d):*#Funcion para cargar el padron electoral* m=[]  
 **for** j **in** range(f):  
 m.append([])  
 **for** i **in** range(1):  
 m[j].append(b[j])  
 m[j].append(a[j])  
 m[j].append(c[j])  
 m[j].append(d[j])  
 **return** (m)  
**def** imprecionmatriz(a,b,c):*#Funcion para la imprecion de matriz* **for** j **in** range(b):  
 **for** i **in** range(c):  
 print(**" "**,a[j][i],**" "**,end=**" "**)  
 print()  
**def** creanomb(a,b,c):*#Funcion para crear n cantidad nombres aleatorios* t=[]  
 **for** j **in** range(c):  
 d=random.randint(0,len(a)-1)  
 e = random.randint(0, len(b) - 1)  
 t.append(**" "**+a[d]+**' '**+b[e])  
 **return** (t)  
**def** cedula(z): *#Crea n cantidad de Cedulas aleatoriamente* t=[]  
 **for** j **in** range(z):  
 a=random.randint(1,10)*#N.Provincia* b=random.randint(100,999)*#Segundo numero* c=random.randint(1,9999)*#Tercer numero* t.append(str(a)+**"-"**+str(b)+**"-"**+str(c))  
 **return** (t)  
**def** centrovot(a,b):*#Gnera una lista con n cantidad de centro de votacion para los votantes* t=[]  
 **for** j **in** range(a):  
 r=random.randint(0,len(b)-1)  
 **if** b[r]==**"Esc.La Mata"**:  
 t.append(**"Esc.La Mata"**)  
 **else**:  
 **if** b[r]==**"Esc.Los Boquerones"**:  
 t.append(**"Esc.Los Boquerones"**)  
 **else**:  
 **if** b[r]==**"Esc.El Espino"**:  
 t.append(**"Esc.El Espino"**)  
 **return** (t)  
**def** mesa(b,a): *#Funcion que Depende del centro de votacion asignado da el numero de mesa* t=[]  
 **for** j **in** range(a):  
 **if** b[j] == **"Esc.La Mata"**:  
 c = random.randint(1,2)  
 t.append(c)  
 **else**:  
 **if** b[j] == **"Esc.Los Boquerones"**:  
 c = random.randint(3,4)  
 t.append(c)  
 **else**:  
 **if** b[j] == **"Esc.El Espino"**:  
 c = random.randint(5,6)  
 t.append(c)  
 **return** (t)  
**def** verifica(a,n,x):*#Verifica cada índice para que no se repita* i=0  
 sw=0  
 **while** (i<n **and** sw==0):  
 **if** a[i]==x:  
 sw=1  
 i=i+1  
 **return** (sw)  
**def** cargamatrizdvotantes(a, b, c):*#Carga una matriz para cada votante "No se pude usar la anterior por formato diferente al cargar los datos"* m=[]  
 **for** j **in** range(a):  
 m.append([])  
 **for** i **in** range(4):  
 m[j].append(b[c[j]][i])  
 **return** (m)  
**def** genevotos():*#Funcion que devuelve un valor que indica el candidato por el cual voto* a=random.randint(0,4)  
 **return** (a)  
**def** concentro(a,b,c,d):*#Da como resultado el total que votaron en x centro de votacion"Matriz"* con=0  
 **for** j **in** range(a):  
 **if** b[j][d]==c:  
 con=con+1  
 **return** (con)  
**def** contmesa(a,b,c):*#Contador de voto por mesa "Nose puede utilizar la funcion anterior para las mesa porque las mesas es una lista"* cont=0  
 **for** j **in** range(a):  
 **if** b[j]==c:  
 cont=cont+1  
 **return** (cont)  
**def** canditogana(a):*#Da la poción del mayor en una lista "En este caso la poción del ganador"* mayor=0  
 **for** j **in** range(len(a)):  
 **if** a[j]>mayor:  
 mayor=a[j]  
 pos=j  
 **return** (pos)  
**def** porcentaje(a,b):*#Porcentaje de voto* c=(a/b)\*100  
 **return** (c)  
b=cedula(cantidad)*#Lista se cedulas*a=creanomb(nomb,apell,cantidad)*#Lista de nombres*c=centrovot(cantidad,centro)*#Lista de centro de votacion*d=mesa(c,cantidad)*#Lista de mesas segun centro de votacion*print(**"PADRON ELECTORAL"**.center(65,**"="**))  
print(**" cedula Nombre Centro de Votacion Mesa"**.upper())  
padron=cargamatriz(cantidad, a, b, c, d)*#Cargar matriz*imprecionmatriz(padron,cantidad,4)*#Imprecion matriz*ind=[]*#Indices aleatorios que votaran*i=0*#Control de ciclo***while** i<votrandom:  
 x=random.randint(0,cantidad-1)  
 **if** verifica(ind,i,x)==0:  
 ind.append(x)  
 i=i+1  
print()  
print(**" cedula Nombre Centro de Votacion Mesa"**.upper())  
padronvotaron=cargamatrizdvotantes(votrandom, padron, ind)  
imprecionmatriz(padronvotaron,votrandom,4)  
print()  
totalcen1=concentro(votrandom,padronvotaron,**"Esc.La Mata"**,2)*#Guarda el total de centro de votacion1*totalmesa1=concentro(votrandom,padronvotaron,1,3)*#Guarda el total de votantes en la mesa 1*totalmesa2=concentro(votrandom,padronvotaron,2,3)*#Guarda el total de votantes en la mesa 2*totalcen2=concentro(votrandom,padronvotaron,**"Esc.Los Boquerones"**,2)*#Guarda el total de centro de votacion2*totalmesa3=concentro(votrandom,padronvotaron,3,3)*#Guarda el total de votantes en la mesa 3*totalmesa4=concentro(votrandom,padronvotaron,4,3)*#Guarda el total de votantes en la mesa 4*totalcen3=concentro(votrandom,padronvotaron,**"Esc.El Espino"**,2)*#Guarda el total de centro de votacion3*totalmesa5=concentro(votrandom,padronvotaron,5,3)*#Guarda el total de votantes en la mesa 5*totalmesa6=concentro(votrandom,padronvotaron,6,3)*#Guarda el total de votantes en la mesa 6*print(**"Total del Corregimiento"**.upper(),votrandom)  
print()  
print(**"El total de votos emitidos por centro de votación"**.center(30,**' '**).upper())  
print(**"Total Esc La mata"**,totalcen1)  
print(**"Total Esc Los boquerlones"**,totalcen2)  
print(**"Total Esc El espino"**,totalcen3)  
print()  
print(**"El total de votos emitidos por mesa"**.upper())  
print(**"Total Mesa 1-"**,totalmesa1)  
print(**"Total Mesa 2-"**,totalmesa2)  
print(**"Total Mesa 3-"**,totalmesa3)  
print(**"Total Mesa 4-"**,totalmesa4)  
print(**"Total Mesa 5-"**,totalmesa5)  
print(**"Total Mesa 6-"**,totalmesa6)  
print()  
votosrepre=[]*#El voto de cada votante para los candidatos***for** j **in** range(votrandom):*#Ciclo para cargar los votos emitidos* votosrepre.append(genevotos())  
print(**"Total por candidato"**.upper())  
cand1=contmesa(votrandom, votosrepre, 1)*#Guarda el total del candidato 1*cand2=contmesa(votrandom, votosrepre, 2)*#Guarda el total del candidato 2*cand3=contmesa(votrandom, votosrepre, 3)*#Guarda el total del candidato 3*cand4=contmesa(votrandom, votosrepre, 4)*#Guarda el total del candidato 4*print(**"Ismael Perez"**,cand1)  
print(**"Tomas Castro"**,cand2)  
print(**"Ricauter Gonzalez"**,cand3)  
print(**"Diana Montes"**,cand4)  
print()  
nulo=contmesa(votrandom, votosrepre, 0)*#Guarda el total de votos en blanco*print(**"Voto valido"**.upper(),votrandom-nulo)  
print(**"Voto blanco"**.upper(),nulo)  
print()  
vototal=[]*#Lista del total de votos para cada candidatos***for** j **in** range(1):*#se carga la lista con el total de votos por candidato* vototal.append(cand1)  
 vototal.append(cand2)  
 vototal.append(cand3)  
 vototal.append(cand4)  
indrepre=canditogana(vototal)*#Indice del ganador o del mayor votos*print(**"Candidato ganador"**,repre[indrepre].upper())*#Se busca el indice en la lista cargada en memoria de los precandidatos a reprecentantes*print()  
a1=porcentaje(cand1,votrandom)*#Porcentaje de votos para le candidato 1*a2=porcentaje(cand2,votrandom)*#Porcentaje de votos para le candidato 2*a3=porcentaje(cand3,votrandom)*#Porcentaje de votos para le candidato 3*a4=porcentaje(cand4,votrandom)*#Porcentaje de votos para le candidato 4*blanco=porcentaje(nulo,votrandom)*#Porcentaje de votos en blanco*print(**"Porcentaje obtenido por cada candidato"**)  
print(**'Ismael Perez'**,**'%.2f'** %a1,**'%'**)  
print(**'Tomas Castro'**,**'%.2f'** %a2,**'%'**)  
print(**'Ricauter Gonzalez'**,**'%.2f'** %a3,**'%'**)  
print(**'Diana Montes'**,**'%.2f'** %a4,**'%'**)  
print(**"Porcentaje de votos en blanco"**)  
print(**'Votos en Blanco'**,**'%.2f'** %blanco,**'%'**)

* 1. Presente los resultados de la ejecución de su programa con varios conjuntos de datos.

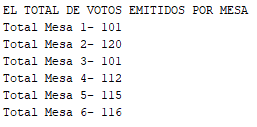
1. El total de votos emitidos en el corregimiento.



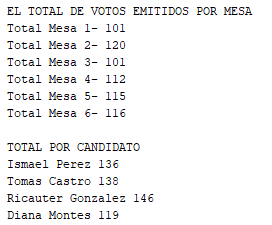
1. El total de votos emitidos por centro de votación.



1. El total de votos emitidos por mesa.



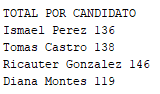
1. El total de votos emitidos por mesa y por candidatos.



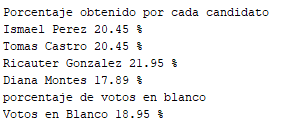
1. El total de votos válidos y de votos en blanco.

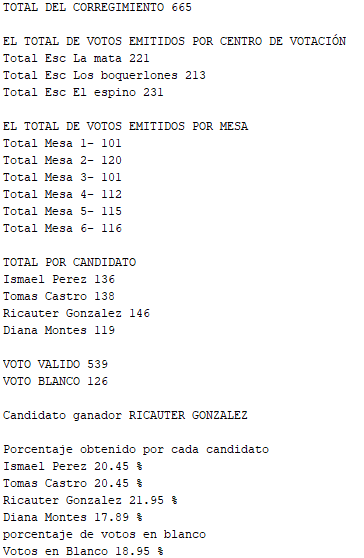


1. El total de votos por candidatos a representante.



1. El candidato ganador.
2. El porcentaje obtenido por cada candidato y el porcentaje de votos en blanco.





* 1. Evalúe individualmente su experiencia, dificultades, dedicación y aprendizajes en atención a la realización de este proyecto, así como el objetivo, y utilidad del programa.

**Rodrigo Gutierrez:**

* La experiencia en el proyecto fue amplia me toco investigar por mi cuenta probar algunas métodos que avía echo en papel para después pasarla al código.
* La dificulta que me encontré a la hora de desarrollar el proyecto era como saber cuántos votos pertenecían a cada mesa y a punta de prueba pude desarrollar una función que pudiera contar cuantos votantes eran por mesa y me resulto.
* El objetivo del programa es aprender hacer funciones y que podríamos reutilizarla para múltiples tareas en el programa principal y así nos ahorraríamos muchas líneas de código.
* La utilidad del programa es que podamos generar votos aleatoriamente y que el mismo nos diga cuantos voto se realizaron para cada candidato y de igual forma cuantos votos por mesa.

**Jesaid Toribio:**

* En este proyecto aprendimos a darle un funcionamiento múltiple de las funciones durante el desarrollo del proyecto.
* Fue un proyecto donde nos dividimos las tareas para así terminar el proyecto en el tiempo estipulado.
* El objetivo era crear un programa modular, esto nos facilita a la codificación ya es más legible el código.
* El programa nos da los resultados de x cantidad de votantes para n número de candidatos, también podemos saber cuántos votos se dieron en el corregimiento y el porcentaje de cada candidato.