

Manual Técnico De los pseudocodigos más importantes del programa

```
metodo probabilidadPlaneta(var cantidadPlanetasNeutras){
    Random random = nuevo objeto de Random();
    para (var i = 0; i < cantidadPlanetasNeutras; i++) {
        var probabilidadPlaneta = random.nextInt(100)+1;
        si (probabilidadPlaneta >= 1 y probabilidadPlaneta <= 45){
            cantTierra += 1 ;
        } sino si (probabilidadPlaneta > 45 y probabilidadPlaneta <= 70){
            cantAgua += 1 ;
        } sino si (probabilidadPlaneta > 70 y probabilidadPlaneta <= 85){
            cantFuego += 1 ;
        } sino si (probabilidadPlaneta > 85 y probabilidadPlaneta <= 95){
            cantOrganico += 1 ;
        } sino si (probabilidadPlaneta > 95 y probabilidadPlaneta <= 100){
            cantRadioactivo += 1 ;
        }
    }
}
```

```
metodo VerificarTipoPlanetaJugadores(var tipoPlaneta, var Planetaj1, var Planetaj2,
var j1, var j2, var cantPlaneta){
    si (tipoPlaneta = 1) {
        planetaNombre[cantPlaneta] = nuevo objeto de
        PlanetaNombre(Planetaj1,j1,"MOLE");
        planetaNombre[cantPlaneta+1] = nuevo objeto de
        PlanetaNombre(Planetaj2,j2,"MOLE");
    } sino si (tipoPlaneta = 2) {
        planetaNombre[cantPlaneta] = nuevo objeto de
        PlanetaNombre(Planetaj1,j1,"NEMO");
        planetaNombre[cantPlaneta+1] = nuevo objeto de
        PlanetaNombre(Planetaj2,j2,"NEMO");
    } sino si (tipoPlaneta = 3) {
        planetaNombre[cantPlaneta] = nuevo objeto de
        PlanetaNombre(Planetaj1,j1,"MAGMA");
        planetaNombre[cantPlaneta+1] = nuevo objeto de
        PlanetaNombre(Planetaj2,j2,"MAGMA");
    } sino si (tipoPlaneta = 4) {
        planetaNombre[cantPlaneta] = nuevo objeto de
        PlanetaNombre(Planetaj1,j1,"GROOT");
        planetaNombre[cantPlaneta+1] = nuevo objeto de
        PlanetaNombre(Planetaj2,j2,"GROOT");
    } sino si (tipoPlaneta = 5) {
        planetaNombre[cantPlaneta] = nuevo objeto de
        PlanetaNombre(Planetaj1,j1,"FISION GUY");
    }
}
```

```
metodo RellenarPlanetas(){
    si (cantTierra > 0) {
        tierra = nuevo objeto de Tierra[cantTierra];
        para (var i = 0; i < tierra de su tamaño; i++) {
            caracter nombre = convertir a caracter el (i+65);
            var nOmbre = convertir a cadena(nombre);
            tierra[i] = nuevo objeto de
            Tierra(nOmbre,planeta.RandomPorcentajeMuertes(),planeta.RandomcantDinero(),planeta.R
            andomCantNaves(),tierra1.RandomCantGuerreroFinalizarTurno(),tierra1.RandomCantDinero
            Turno(),1);
        }
    }
}
```

```

        planetaNombre[i] = nuevo objeto de
PlanetaNombre(nOmbre,"Neutral","MOLE");

    }
}
si (cantAgua > 0) {
    agua = nuevo objeto de Agua[cantAgua];
    para (var i = 0; i < agua de su tamaño; i++) {
        caracter nombre = (caracter)(i+65+cantTierra);
        var nOmbre =convertir a cadena(nombre);
        agua[i] = nuevo objeto de Agua
(nOmbre,planeta.RandomPorcentajeMuertes(),planeta.RandomcantDinero(),planeta.Rando
mCantNaves(),agua1.RandomCantGuerreroFinalizarTurno(),agua1.RandomCantDineroTurno()
,1);
        planetaNombre[i+cantTierra] = nuevo objeto de
PlanetaNombre(nOmbre,"Neutral","NEMO");

    }
}
si (cantFuego > 0) {
    fuego = nuevo objeto de Fuego[cantFuego];
    para (var i = 0; i < fuego de su tamaño; i++) {
        caracter nombre = (caracter)(i+65+cantTierra+cantAgua);
        var nOmbre =convertir a cadena(nombre);
        fuego[i] = nuevo objeto de
Fuego(nOmbre,planeta.RandomPorcentajeMuertes(),planeta.RandomcantDinero(),planeta.R
andomCantNaves(),fuego1.RandomCantGuerreroFinalizarTurno(),fuego1.RandomCantDinero
Turno(),1);
        planetaNombre[i+cantTierra+cantAgua] = nuevo objeto de
PlanetaNombre(nOmbre,"Neutral","MAGMA");

    }
}
si (cantOrganico > 0) {
    organico = nuevo objeto de Organico[cantOrganico];
    para (var i = 0; i < organico de su tamaño; i++) {
        caracter nombre = (caracter)(i+65+cantTierra+cantAgua+cantFuego);
        var nOmbre =convertir a cadena(nombre);
        organico[i] = nuevo objeto de
Organico(nOmbre,planeta.RandomPorcentajeMuertes(),planeta.RandomcantDinero(),planet
a.RandomCantNaves(),organico1.RandomCantGuerreroFinalizarTurno(),organico1.RandomC
antDineroTurno(),1);
        planetaNombre[i+cantTierra+cantAgua+cantFuego] = nuevo objeto de
PlanetaNombre(nOmbre,"Neutral","GROOT");

    }
}
si (cantRadioactivo > 0) {
    radioactivo = nuevo objeto de Radioactivo[cantRadioactivo];
    para (var i = 0; i < radioactivo de su tamaño; i++) {
        caracter nombre =
(caracter)(i+65+cantTierra+cantAgua+cantFuego+cantOrganico);
        var nOmbre =convertir a cadena(nombre);
        radioactivo[i] = nuevo objeto de
Radioactivo(nOmbre,planeta.RandomPorcentajeMuertes(),planeta.RandomcantDinero(),pla
neta.RandomCantNaves(),radioactivo1.RandomCantGuerreroFinalizarTurno(),radioactivo1.R
andomCantDineroTurno(),1);
        planetaNombre[i+cantTierra+cantAgua+cantFuego+cantOrganico] = nuevo
objeto de PlanetaNombre(nOmbre,"Neutral","FISION GUY");
    }
}

```

```

    }
}

metodo DibujarMapa(var tamañoX, var tamañoY){
    Escribir("");
    para (var i = 1; i <= tamañoX; i++) {
        caracter ascci = (caracter)(i+64);
        Escribir Con Estilo("%15C",ascci);
    }
    Escribir("");
    para (var y = 0; y < tamañoY; y++) {
        Escribir Con Estilo("\n %d ",y+1);
        para (var x = 0; x < tamañoX; x++) {
            si (tablero[x][y] = no hay nada) {
                Escribir Con Estilo(" %23s ", " ");
            } sino{
                Escribir Con Estilo(" %23s ",tablero[x][y]);
            }
        }
        Escribir Con Estilo("\n \n");
    }
}

metodo VerificarJuego(var jugador){
    var contador = 0;
    para (var i = 0; i < planetaNombre de su tamaño; i++) {
        si (jugador.=(planetaNombre[i].getDueñoPlaneta())) {
            contador += 1;
        }
    }
    si (contador = planetaNombre de su tamaño) {
        Escribir("Felicidades "+jugador+" acaba de ganar a su enemigo");
        finJuego = verdadero;
        turnoJugador = falso;
    }
}

metodo MedicionDistancias(){
    Escribir("Ingrese el planeta de origen hacia el planeta destino separado por una
coma (,)");
    var medicionDistancia = scanner.nextLine();
    var[] planetas = medicionDistancia.split(",");
    var posicionPlanetaOrigenX = PosicionX(planetas[0]);
    var posicionPlanetaOrigenY = PosicionY(planetas[0]);
    var posicionPlanetaDestinoX = PosicionX(planetas[1]);
    var posicionPlanetaDestinoY = PosicionY(planetas[1]);
    double distancia = 0;
    si ((posicionPlanetaOrigenX-posicionPlanetaDestinoX) = 0) {
        si (posicionPlanetaOrigenY > posicionPlanetaDestinoY) {
            distancia = posicionPlanetaOrigenY - posicionPlanetaDestinoY;
        }sino{
            distancia = posicionPlanetaDestinoY - posicionPlanetaOrigenY;
        }
        Escribir("La distancia desde "+planetas[0]+" hasta "+planetas[1]+" es de: "+
distancia+ " años luz");
    } sino si ((posicionPlanetaOrigenY-posicionPlanetaDestinoY) = 0) {
        si (posicionPlanetaOrigenX > posicionPlanetaDestinoX) {

```

```

        distancia = posicionPlanetaOrigenX - posicionPlanetaDestinoX;
    }sino{
        distancia = posicionPlanetaDestinoX - posicionPlanetaOrigenX;
    }
    Escribir("La distancia desde "+planetas[0]+" hasta "+planetas[1]+" es de: "+
distancia+ " años luz");
    } sino{
        distancia = Raiz cuadrada
(Elevada(posicionPlanetaDestinoX-posicionPlanetaOrigenX,2)+Elevada(posicionPlanetaDestinoY-posicionPlanetaOrigenY,2));

        Escribir("La distancia desde "+planetas[0]+" hasta "+planetas[1]+" es de: "+
distancia+ " años luz");
    }

}

funcionPosicionX(var cadena){
    caracter posX = cadena tomar el caracter de la posicion(0);
    var POSX =convertir a cadena(posX);
    POSX = Colocar todo a mayuscula(POSX);
    posX = POSX tomar el caracter de la posicion(0);
    var posicionX = ((convertir a numero)posX)-65);
    Devolver posicionX;
}

funcionPosicionY(var cadena){
    si (cadena de su tamaño() > 2) {
        var PosY = "";
        para (var i = 1; i < cadena de su tamaño(); i++) {
            PosY = PosY + cadena tomar el caracter de la posicion(i);
        }
        var posicionY = transformar a numero entero(PosY)-1;
        Devolver posicionY;
    } sino{
        caracter posY = cadena tomar el caracter de la posicion(1);
        var POSY =convertir a cadena(posY);
        var posicionY = transformar a numero entero(POSY)-1;
        Devolver posicionY;
    }
}

metodo ConsultaFlotas(var turnoActual, var jugador){
    Escribir("El turno actual: "+ turnoActual);
    Escribir("Envios de naves que no han llegado a su destino: \n");
    var n = 0;
    si (registroTotal[0] = no hay nada) {
        Escribir("\n No hay naves viajando en este momento \n");
    }sino si (registroTotal[0] <>no hay nada y cantEnvios = 1 y
registroTotal[0].isEntregado() = verdadero) {
        si (jugador.=(registroTotal[0].getNombreJugador())) {
            Escribir(registroTotal[0].MostrarDatos());
        }
    } sino{
        para (var i = 0; i < registroTotal de su tamaño; i++) {
            si
(jugador.=(registroTotal[i].getNombreJugador())yregistroTotal[i].isEntregado() = verdadero)
{
                Escribir(registroTotal[i].MostrarDatos());
                n = 1;
            }
        }
    }
}

```

```

    }
    si (n = 0) {
        Escribir("\n No hay naves viajando en este momento\n");
    }
}

metodo RedimensionarRegistroTotal(){
    si (registroTotal[0] = no hay nada) {
        registroTotal = nuevo objeto de RegistroEnvios[1];
    } sino{
        RegistroEnvios[] registro = nuevo objeto de RegistroEnvios[registroTotal de
su tamaño];
        para (var i = 0; i < registroTotal de su tamaño; i++) {
            registro[i] = registroTotal[i];
        }
        registroTotal = nuevo objeto de RegistroEnvios[cantEnvios];
        para (var i = 0; i < (registroTotal de su tamaño-1); i++) {
            registroTotal[i] = registro[i];
        }
    }
}

metodo cambiarDueño(var nombrePlaneta,var jugador){
    para (var i = 0; i < planetaNombre de su tamaño; i++) {
        si (nombrePlaneta.=(planetaNombre[i].getNombrePlaneta())) {
            planetaNombre[i].setDueñoPlaneta(jugador);
            posicionPlaneta = i;
            romper ciclo;
        }
    }
}

metodo cambiarColor(var codigoJugador,var PosicionPlaneta,var randomX,var
randomY){
    si (codigoJugador = 1) {
        tablero[randomX][randomY]= fondo color rojo+"Nombre:
"+planetaNombre[PosicionPlaneta].getNombrePlaneta()+" Dueño:
"+planetaNombre[PosicionPlaneta].getDueñoPlaneta();
        tableroNombreDueño[randomX][randomY] =
planetaNombre[PosicionPlaneta].getDueñoPlaneta();
    } sino si (codigoJugador = 2) {
        tablero[randomX][randomY]= fondo color azul+"Nombre:
"+planetaNombre[PosicionPlaneta].getNombrePlaneta()+" Dueño:
"+planetaNombre[PosicionPlaneta].getDueñoPlaneta();
        tableroNombreDueño[randomX][randomY] =
planetaNombre[PosicionPlaneta].getDueñoPlaneta();
    }
}

metodo UbicarPlanetasMapa(var cantidadPlanetasNeutrales,var tamX,var tamY, var j1,
var j2){
    para (var i = 0; i < cantidadPlanetasNeutrales+2; i++) {
        var ocupado = verdadero;
        mientras(ocupado = verdadero){
            var randomX = Numero aleatorio (tamX);
            var randomY = Numero aleatorio (tamY);
            si (tablero[randomX][randomY] = no hay nada) {
                cambiarColorDueño(randomX,randomY,j1,j2,i);
            }
        }
    }
}

```

```
        ocupado = falso;
    }
}
}
```

Nombre: Sergio Rolando Hernández Pérez
Carnet: 201931555