**Universidad Tecnológica de Panamá**

**Facultad de Ingeniería de sistemas computacionales**

**Departamento de programa de computadora**

**Proyecto No.1**

**Desarrollo de software I**

**Integrantes: Madelaine Barahona 8-922-1246**

**Yusuf Rawat 8-986-603**

**Davis Santos 8-973-405**

**Anthony Cardoze 8-850-1742**

**Profesor(a): Ludia Melendez**

**Grupo: 1LS705**

**Panamá 2021**

# **Introducción**

# **Enunciado del problema**

La clínica de pacientes de Covi que fue creada en medio de la pandemia desea conocer

cuántos pacientes se atendieron el día de hoy por los 5 médicos de 5 especialidades

diferentes, de las cuales se maneja un código neu= Neumólogo, car=Cardiólogo,

carV=Cardiovascular, endo= Endocrinólogo y medG= Medicina general.

Por cada médico se cuenta con un código de médico, cod de especialidad y un arreglo de

8 posiciones, que contiene un 1 =atendió paciente o un 0 =no atendió paciente.

Cada paciente que solicitó una cita, suministro la siguiente información:

Código del médico, código de la especialidad deseada, de haber cupo se debió ubicar en

el arreglo del médico solicitado un 1. Informe a cada paciente solamente si será atendido o

no en el día de hoy. El código del médico es un numero entero que debe estar en el rango

de (0 a 4). Valide que la especialidad del médico solicitado sea la que se maneja en la

clínica.

Su algoritmo debe realizar los siguientes procesos una vez se tenga los datos del día:

* Cuántos pacientes atendió cada médico
* Cuál es el médico que más pacientes atendió, cuántos fueron y de qué especialidad, imprima el nombre de la especialidad completo.
* Qué especialidad fue la más solicitada imprima el nombre completo de la especialidad
* Total, de pacientes atendidos en el día.

Su algoritmo debe:

1. Manejar funciones de acuerdo con los procesos solicitados.

2. Leer e imprimir en el algoritmo principal.

3. Validaciones si se ameritan.

# **Análisis y diseño del problema**

|  |  |
| --- | --- |
| **Análisis y Diseño** | |
| **Datos** | **Descripción** |
| **entrada** | **Codigo del medico(soliciCodMed), especialidad del medico(soliciCodEsp)** |
| **procesos** | **Repetir mientras se desean solicitar más citas:**   * **se validan si los datos son correctos y coinciden** * **se valida si el doctor tiene cupos disponibles** |
| **salida** | **Imprimir total de citas de cada medico**  **Imprimir cual medico atendió mas personas**  **Imprimir cual es la especialidad mas solicitada**  **Imprimir total de pacientes atendidos** |

# **Algoritmo**

// funciones

entero validaMed (entero codMed, cadena codEspe){

si(codMed < 0 O codMed > 4){

retorna 0;

}

de otro modo si(codMed == 0 Y codEspe == "neu"){

retorna 1;

}

de otro modo si (codMed == 1 Y codEspe == "car"){

retorna 1;

}

de otro modo si (codMed == 2 Y codEspe == "carV"){

retorna 1;

}

de otro modo si (codMed == 3 Y codEspe == "endo"){

retorna 1;

}

de otro modo si (codMed == 4 Y codEspe == "medG"){

retorna 1;

}

de otro modo {

retorna 0;

}

}

entero contAtencion (entero array[]){

entero i, contador = 0;

// array 0 al 7

para(i = 0; i < 8; i = i + 1){

si(array[i] == 1){

contador = contador + 1;

}

}

retornar contador;

}

entero topContador (entero array1[], entero array2[], entero array3[], entero array4[], entero array5[]){

entero contador1;

entero contador2;

entero contador3;

entero contador4;

entero contador5;

// contar cada arreglo

contador1 = contAtencion(array1);

contador2 = contAtencion(array2);

contador3 = contAtencion(array3);

contador4 = contAtencion(array4);

contador5 = contAtencion(array5);

si(contador1 > contador2 Y contador1 > contador3 Y contador1 > contador4 Y contador1 > contador5){

retornar 1;

}

de otro modo si (contador2 > contador3 Y contador2 > contador4 Y contador2 > contador5){

retornar 2;

}

de otro modo si (contador3 > contador4 Y contador3 > contador5){

retornar 3;

}

de otro modo si (contador4 > contador5){

retornar 4;

}

de otro modo {

retornar 5;

}

}

Algoritmo CovidClinica {

//bloque declarativa

//Neumólogo

entero medicoUno = 0;

cadena espeMedUno = "neu";

entero arrAtenUno[8];

//Cardiólogo,

entero medicoDos = 1;

cadena espeMedDos = "car";

entero arrAtenDos[8];

//Cardiovascular

entero medicoTres = 2;

cadena espeMedTres = "carV";

entero arrAtenTres[8];

//Endocrinólogo

entero medicoCuatro = 3;

cadena espeMedCuatro = "endo";

entero arrAtenCuatro[8];

//Medicina general

entero medicoCinco = 4;

cadena espeMedCinco = "medG";

entero arrAtenCinco[8];

cadena confirm = 's';

entero soliciCodMed;

entero soliciCodEsp;

entero i;

// bloque de instrucciones

// inicializar todos los arreglos en zero.

para(i = 0; i < 8; i = i + 1){

arrAtenUno[i] = 0;

arrAtenDos[i] = 0;

arrAtenTres[i] = 0;

arrAtenCuatro[i] = 0;

arrAtenCinco[i] = 0;

}

mientras(confirm == 's' O confirm == 'S'){

imprimir("Introduzca el codigo de medico: ");

leer(soliciCodMed);

imprimir("Introduzca el codigo de especialidad: ");

leer(soliciCodEsp);

//validacion de medico y su especialidad

mientras(validaMed(soliciCodMed, soliciCodEsp) == 0){

imprimir("Introduzca el codigo de medico correcto: ");

leer(soliciCodMed);

imprimir("Introduzca el codigo de especialidad correcto: ");

leer(soliciCodEsp);

}

//se valida el doctor y la cantidad de pacientes que ocupa, si el contador de pacientes es menor a 8 que es el cupo maximo pone el entero 1 en la casilla vacia y se imprime al usuario si tiene cita y con quien.

si(soliciCodMed == medicoUno Y contAtencion(arrAtenUno) <= 8){

arrAtenUno[contAtencion(arrAtenUno)] = 1;

imprimir("Usted sera atendido por el medico: " + medicoUno + " el dia de hoy");

}

de otro modo si (soliciCodMed == medicoDos Y contAtencion(arrAtenDos) <= 8){

arrAtenDos[contAtencion(arrAtenDos)] = 1;

imprimir("Usted sera atendido por el medico: " + medicoDos + " el dia de hoy");

}

de otro modo si (soliciCodMed == medicoTres Y contAtencion(arrAtenTres) <= 8){

arrAtenTres[contAtencion(arrAtenTres)] = 1;

imprimir("Usted sera atendido por el medico: " + medicoTres + " el dia de hoy");

}

de otro modo si (soliciCodMed == medicoCuatro Y contAtencion(arrAtenCuatro) <= 8){

arrAtenCuatro[contAtencion(arrAtenCuatro)] = 1;

imprimir("Usted sera atendido por el medico: " + medicoCuatro + " el dia de hoy");

}

de otro modo si (soliciCodMed == medicoCinco Y contAtencion(arrAtenCinco) <= 8){

arrAtenCinco[contAtencion(arrAtenCinco)] = 1;

imprimir("Usted sera atendido por el medico: " + medicoCinco + " el dia de hoy");

}

de otro modo{

imprimir("No hay cupo para el dia de hoy.");

}

imprimir("Desea solicitar otra cita?");

leer(confirm);

}

// impresion de los valores acumulados una vez el usuario salga del proceso.

// cantidad de pacientes por medico en total

imprimir("Cantidad de atenciones del medico" + medicoUno + " : " + contAtencion(arrAtenUno));

imprimir("Cantidad de atenciones del medico" + medicoDos + " : " + contAtencion(arrAtenDos));

imprimir("Cantidad de atenciones del medico" + medicoTres + " : " + contAtencion(arrAtenTres));

imprimir("Cantidad de atenciones del medico" + medicoCuatro + " : " + contAtencion(arrAtenCuatro));

imprimir("Cantidad de atenciones del medico" + medicoCinco + " : " + contAtencion(arrAtenCinco));

// medico que mas pacientes atendio , cuantos fuero y de que especialidad

// especialidad mas solicitada

si(medicoUno == topContador(arrAtenUno, arrAtenDos, arrAtenTres, arrAtenCuatro, arrAtenCinco)){

imprimir("El medico " + medicoUno + " es el que tiene mas visitas con la especialidad de Neumólogo");

imprimir("La especialidad mas solicitada es Neumólogo");

}

de otro modo si (medicoDos == topContador(arrAtenUno, arrAtenDos, arrAtenTres, arrAtenCuatro, arrAtenCinco)){

imprimir("El medico " + medicoDos + " es el que tiene mas visitas con la especialidad de Cardiólogo");

imprimir("La especialidad mas solicitada es Cardiólogo");

}

de otro modo si (medicoTres == topContador(arrAtenUno, arrAtenDos, arrAtenTres, arrAtenCuatro, arrAtenCinco)){

imprimir("El medico " + medicoTres + " es el que tiene mas visitas con la especialidad de Cardiovascular");

imprimir("La especialidad mas solicitada es Cardiovascular");

}

de otro modo si (medicoCuatro == topContador(arrAtenUno, arrAtenDos, arrAtenTres, arrAtenCuatro, arrAtenCinco)){

imprimir("El medico " + medicoCuatro + " es el que tiene mas visitas con la especialidad de Endocrinólogo");

imprimir("La especialidad mas solicitada es Endocrinólogo");

}

de otro modo si (medicoCinco == topContador(arrAtenUno, arrAtenDos, arrAtenTres, arrAtenCuatro, arrAtenCinco)){

imprimir("El medico " + medicoCinco + " es el que tiene mas visitas con la especialidad de Medicina general");

imprimir("La especialidad mas solicitada es Medicina general");

}

de otro modo{

imprimir("No se registran citas.");

}

// cantidad de pacientes atendidos en total

imprimir("Total de pacientes atendidos hoy: " +

contAtencion(arrAtenUno) +

contAtencion(arrAtenDos) +

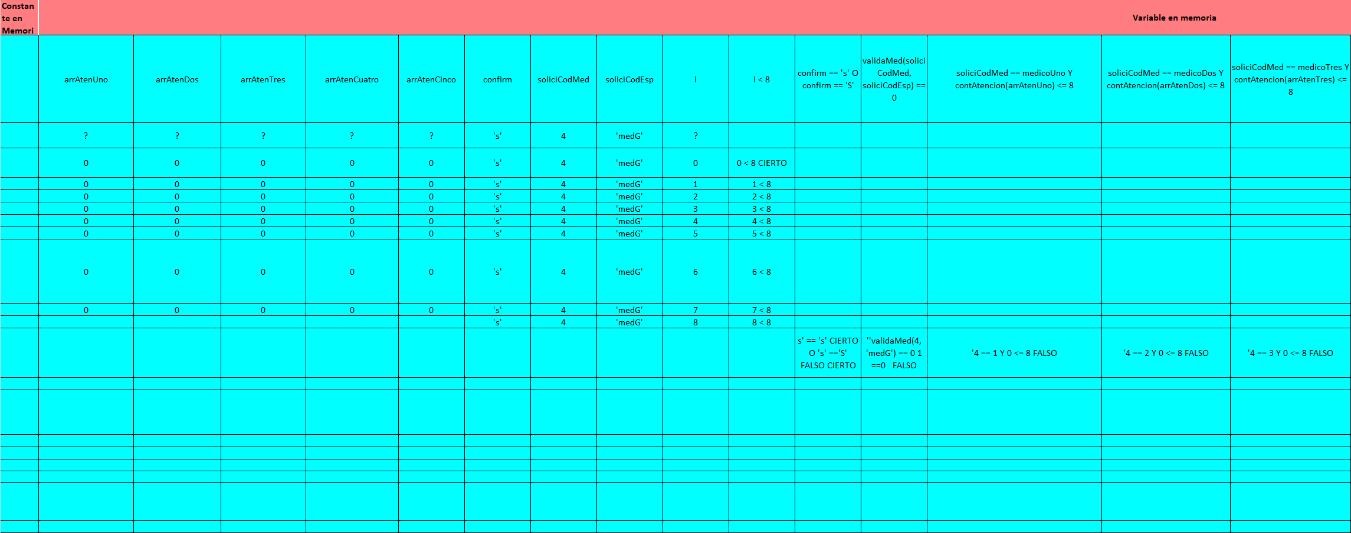
contAtencion(arrAtenTres) +

contAtencion(arrAtenCuatro) +

contAtencion(arrAtenCinco));

}

# **Prueba de escritorio**





# **Prueba de escritorio**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **MEMORIA DE LA FUNCION contAtencion (entero array[])** | | | | |
| array[] | i | contador | i < 8 | array[i] == 1 |
| 0 | 0 | 0 | 0 < 8 CIERTO | 0 == 1 FALSO |
| 0 | 0 | 0 | 1 < 8 CIERTO | 0 == 1 FALSO |
| 0 | 0 | 0 | 2 < 8 CIERTO | 0 == 1 FALSO |
| 0 | 0 | 0 | 3 < 8 CIERTO | 0 ==1 FALSO |
| 0 | 0 | 0 | 4 < 8 CIERTO | 0 == 1 FALSO |
| 0 | 0 | 0 | 5 < 8 CIERTO | 0 == 1 FALSO |
| 0 | 0 | 0 | 6 < 8 CIERTO | 0 == 1 FALSO |
| 0 | 0 | 0 | 7 < 8 CIERTO | 0 == 1 FALSO |
| 0 | 0 | 0 | 8 < 8 FALSO | 0 == 1 FALSO |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **MEMORIA DE LA FUNCION topContador (entero array1[], entero array2[], entero array3[], entero array4[], entero array5[])** | | | | | | | | | | | | | |
| array1[] | array2[] | array3[] | array4[] | array5[] | contador1 | contador2 | contador3 | contador4 | contador5 | contador1 > contador2 Y contador1 > contador3 Y contador1 > contador4 Y contador1 > contador5 | contador2 > contador3 Y contador2 > contador4 Y contador2 > contador5 | contador3 > contador4 Y contador3 > contador5 | contador4 > contador5 |
| 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 > 0 Y 0 > 0 Y 0 > 1 Y 0 > 0 | 0 > 0 Y 0 > 1 Y 0 > 0 | 0 > 1 Y 0 > 0 FALSO | 1 > 0 CIERTO |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **MEMORIA DE LA FUNCION validaMed (entero codMed, cadena codEspe)** | | | | | | | |
| codMed | codEspe | codMed < 0 O codMed > 4 | codMed == 0 Y codEspe == "neu" | codMed == 1 Y codEspe == "car" | codMed == 2 Y codEspe == "carV" | codMed == 3 Y codEspe == "endo" | codMed == 4 Y codEspe == "medG" |
| 4 | 'medG' | 4<0 O 4>4 FALSO | 4 == 0 Y "medG" == "neu" FALSO | 4 == 1 Y "medG" == "car" FALSO | 4 == 2 Y "medG" == "carV" FALSO | 4 == 3 Y "medG" == "endo" FALSO | 4 == 4 Y "medG" == "medG" CIERTO |

**Conclusiones**