**Escuela Superior de Ingeniería Mecánica y Eléctrica Unidad Culhuacán**

**Proyecto Final**

Mata Juárez Christian

**Profesor: Juan Ángel Rodríguez Gómez**

**GRUPO 2CV33**

**Índice**

1.- Introducción..............................................................................................3

2.- Análisis del problema…...…………………………………………………….4

3.- Elaboración de algoritmo………………………………………………..……4

4.- Módulos de actualización:……………………………………………………8

5.-Carga puntual…………………………………………………………………..8

6.- Capacitancia.…………………………………………………………………14

5.- Codificación e implementación………………………………………..…....10

7.- Pruebas modulares e integrales ….…………………………………..…...20

8.- Mantenimiento….………….……………………………………………..…..26

9.- Conclusión…..….………….……………………………………………..…..26

Nota: El proyecto final de Electricidad y Magnetismo está incluido dentro de mi proyecto final de Programación Orientada a Objetos. Los apartados del proyecto se encuentran específicamente en los Carga puntual y Capacitancia.

**Introducción:**

El programa desarrollado como proyecto final para la clase de programación orientada a objetos (incluyendo el proyecto final del Electricidad y magnetismo) puede ser considerado como una a herramienta de ayuda para los alumnos de primer y segundo semestre, pensada principalmente para realizar aquellas actividades matemáticas, que presentan tiempos largos de resolución, de manera sencilla y rápida. Si bien el alumno que lo utilice debe aprender los métodos para resolver los problemas matemáticos contemplados en este programa. El mismo se encuentra pensado para que a largo plazo, y contemplando aquellas lagunas generadas por el tiempo, el usuario pueda aprovechar el software de tal forma que le sea posible resolver los problemas que se le presenten de manera práctica y eficiente.

El mismo está constituido por ocho módulos principales que contemplan los siguientes temas: calculadora, factorización de ecuaciones de primero y segundo grado, calculadora de complejos, traductor binario y el demo de un videojuego, análisis combinatorio, carga puntual y capacitancia. Todos y cada uno de los módulos, si bien pueden parecer aleatorios, se encuentran enfocados en distintas estrategias de programación que se aprendieron durante los dos primeros semestres de la carrera, optando siempre por una interfaz amigable y que resulte entendible para todos los usuarios del nivel medio superior o superior.

Actualización:

Para la presente actualización del programa 'Calculadora' se ha realizado la transcripción completa del código fuente del lenguaje C a C++, pero no se limita únicamente a esto, sino que, del código fuente original se lograron eliminar cerca de 900 líneas que en su momento parecieron ser adecuadas, pero que en un análisis posterior se identificaron como prescindibles. De igual forma se implementaron tres nuevos módulos que son, cálculo de carga puntual, capacitancia y análisis combinatorio.

Este proyecto crecerá con el paso del tiempo y busca ser una herramienta para todos los estudiantes de ingeniería en computación.

**Análisis del problema**

En este caso el software está diseñado con la intensión de aprobar el curso de programación orientada a objetos, pero en la medida que esto se vuelve una tarea con cierta libertad, mi objetivo secundario, que presenta mayor importancia en lo personal, consiste en realizar un programa que permita realizar tareas matemáticas y lingüísticas complejas de manera sencilla, incrementando tiempos de producción y eficiencia en mis labores.

**Elaboración del algoritmo**

Tomando en consideración lo anterior, el Software se dividirá de la siguiente manera: pantalla de inicio, esta desplegará un mensaje general del programa “Calculadora”. Ventana de introducción, donde se especificará el objetivo principal del programa; Menú principal, mostrará los módulos principales del programa y le permitirá al usuario seleccionar cualquiera de las 19 opciones disponibles, o bien salir del programa. El usuario será capaz de volver a esta sección cada que lo desea al abandonar los módulos principales, los cuales son los siguientes.

Cambiar color de la terminal:

Un módulo sencillo que le permitirá al usuario seleccionar el color de la terminal entra una selección de 15 colores, esto es una opción que se añade en el programa para brindar opciones de personalización en favor del desempeño gráfico, y para agradar al usuario.

|  |  |
| --- | --- |
| Clase: Calculadoras | Se encuentra en las líneas  140-195 |
| Codificación |

Calculadora:

Un módulo que, como su nombre lo indica, sirve como una calculadora normal y que poseerá cierta nomenclatura que busque un trato más amigable con el usuario, se constituirá por una interfaz visual que explique todos los comandos que maneja la misma, dos secciones para mostrar el resultado: una con punto decimal, la otra en fracción (sólo en caso de que se requiera) y una entrada de tres datos, 2 double y un carácter, mismos que servirán para realizar todas las operaciones contenidas dentro del programa. Las operaciones contempladas de primera cuenta serán: suma, multiplicación, resta, división, residuo, potencia, raíz, logaritmo neperiano, logaritmo base 10, funciones trigonométricas (en radianes y grados), y el caso para salir al menú principal.

|  |  |
| --- | --- |
| Clase:Calculadora | Se encuentra en las líneas  216-440 |
| Codificación |

Ecuaciones de primer y segundo grado:

Un módulo dedicado a la factorización hasta ecuaciones de segundo grado. Constará con una interfaz más sencilla que el apartado anterior pero más potente en la capacidad de resolución, ya que será un módulo dedicado a la factorización de estas. En este sentido permitirá introducir los coeficientes de la ecuación, mostrará el discriminante, las raíces, mencionará el tipo de ecuación con la que se está trabajando, y como punto más importante, la factorización. De igual forma, en caso de que el discriminante sea negativo se mostrará un mensaje de error indicando la naturaleza de sus raíces imaginarias. Después de cada operación se mostrará al centro de la pantalla un cuadro de texto para continuar en el módulo, o bien salir al menú principal.

|  |  |
| --- | --- |
| Clase: Factorización | Se encuentra en las líneas  444-942 |
| Codificación |

Calculadora de complejos:

Módulo que tiene como objetivo operaciones básicas con números complejos, el mismo buscará una interfaz sencilla para introducir los números complejos de la forma “(x+iy)+(x1-iy1)” y el operador principal variará dependiendo de la operación deseada. Los elementos que se mostrarán en pantalla serán las operaciones que se pueden realizas(suma, resta, multiplicación y división) y el resultado en tres formas distintas: complejo, exponencial y polar. Después de cada operación se mostrará al centro de la pantalla un cuadro de texto para continuar en el módulo, o bien salir al menú principal.

|  |  |
| --- | --- |
| Clase: Calculadora\_complejos | Se encuentra en las líneas  944-1275 |
| Codificación |

Traductor binario/texto:

Este módulo un tanto distante de los anteriores, consistirá en un traductor de lenguaje ordinario a lenguaje máquina en consonancia con el código ASCII; es decir, permitirá introducir el texto que se requiera que el programa lo traduzca en ambos sentidos de lenguaje natural a código binario y viceversa. Por loa naturaleza del módulo se mostrará primeramente un recuadro para que el usuario pueda seleccionar el tipo de traducción que utilizará, para esto podrás sólo escribir la letra “t” si desea traducir del lenguaje ordinario, la letra “b” si desea traducir de binario y la letra “s” si desea salir al menú principal. En la traducción de lenguaje natural al binario se mostrarán dos secciones, una para introducir el texto y otra donde se desplegará el código binario en paquetes de bytes con salto de línea cada ocho bytes, al mismo tiempo el programa almacenará la traducción en el archivo “Binarios.txt” que se encuentra en la carpeta del programa para que la traducción pueda ser enviada a otro usuario del programa. En la traducción de binario a lenguaje ordinario la interfaz será similar pero en este caso el programa leerá el texto binario y traducirá a lenguaje natural aplicando un salto de línea cada 80 caracteres. El programa después de realizar cada traducción regresará al cuadro de selección.

|  |  |
| --- | --- |
| Clase: Traductores | Se encuentra en las líneas  1279-1423 |
| Codificación |

Order 66:

El último módulo del programa es un módulo que no se mostrará en el menú principal, pero sin embargo será ejecutable si seleccionas el número “66” desde el menú principal, el mismo sólo podrá ejecutarse una vez cada que se ejecute el programa. Consiste básicamente en la demostración del primer nivel de un juego de calabozos que se encuentra en fase de diseño. El módulo iniciará con animaciones implementadas con arreglos tanto unidimensionales como bidimensionales. Estará completamente basado en caracteres del código ASCII y consistirá en un ladrón de bancos que debe salir con el botín completo, las principales teclas a usar serán “w”, “a”, “s” y “d”, las cuales serán ocupadas como parte de las mecánicas de juego e inmersión dentro del mapa principal, el cual estará contará con un sistema de colisiones perfectamente mapeado en el que el usuario podrá navegar sin encontrar errores en el sistema de colisiones, se colocarán accesorios que sumarán al botín del ladrón; habrá trampas y paredes secretas que deberá sortear si quiere salir con su botín intacto; si el usuario. El juego termina cuando el personaje es capturado o cuando logra llegar a la salida.

|  |  |
| --- | --- |
| Clase: Juego | Se encuentra en las líneas  1425-2127 |
| Codificación |

Módulos complementarios:

Dentro del programa se tuvieron que desarrollar tres módulos que no estaban contemplados en el proceso de desarrollo pero su creación ayuda a no sólo un módulo, sino a varios y por esto se encuentran deslocalizados de las partes principales del programa. Estos son:

* Módulo error: arroja un mensaje de alerta que nos indica un problema con los datos que se están introduciendo.

|  |  |
| --- | --- |
| Clase: Calculadoras | Se encuentra en las líneas  36-45 |
| Codificación |

* Módulo repetición: muestra una ventana emergente que te permite realizar una operación de la misma índole; o bien, regresar al menú principal.

|  |  |
| --- | --- |
| Clase: Calculadoras | Se encuentra en las líneas  47-69 |
| Codificación |

* Módulo fracciones: un apartado dedicado a convertir cualquier número con punto decimal en una fracción de la forma (p/q), este es de los módulos más importantes ya que permite el correcto desarrollo de la mayoría de los módulos y a pesar de ser un programa sencillo su alcance superó las expectativas que tenía durante su desarrollo.

|  |  |
| --- | --- |
| Clase: Calculadoras | Se encuentra en las líneas  87-127 |
| Codificación |

**Módulos de actualización (Proyecto final de electricidad):**

Es importante comentar que la actualización tuvo como resultado la eliminación de cerca de 900 líneas de código que se encontraban mientras el programa fue escrito en C, siendo una reducción considerable de 33% la que encontramos en la versión 2.0 escrita en C++. Con los tres módulos nuevos el código es incluso menor que la versión inicial.

Carga Puntual:

Como su nombre lo indica, este módulo está dedicado a calcular la fuerza que se pone en juego cuando se encuentran dos cargas puntuales en el espacio, entre las variables que se utilizan se encuentra el valor de las cargas, Radio, Ángulo, prefijo. Con estos datos el programa logra proporcionarnos los datos fundamentales que necesitamos saber sobre nuestro objeto de estudio, en específico. El resultado de la fuerza total y cómo se descompone luego en la fuerza actuante sobre el eje x y la fuerza actuante sobre el eje y. En esta sección sólo se utiliza una constante, k=8.99x10^9 y siempre se trabaja especificando las medidas que utilizamos.

|  |  |
| --- | --- |
| Clase: Electricidad\_y\_magnetismo | Se encuentra en las líneas  2446-2519 |
| Codificación |

Tema:

Le fuerza eléctrica desempeña un papel aún más profundo en nuestras vidas. De acuerdo con la teoría atómica, las fuerzas eléctricas entre átomos y moléculas son las responsables de mantenerlos unidos para formar líquidos y sólidos, además de que las fuerzas eléctricas también están implicadas en los procesos metabólicos que ocurren en el interior de nuestros cuerpos. La palabra electricidad proviene de la palabra griega elektron, que significa “ámbar”. El ámbar es una resina de árbol petrificada; los antiguos sabían que si frotaban un pedazo de ámbar con una tela, el ámbar atraería hojas pequeñas o polvo. Una pieza de goma o de caucho rígida, una varilla de vidrio o una regla de plástico frotada contra una tela también presentan este “efecto ámbar”, o electrostática, como lo llamamos hoy. En efecto, nosotros podemos levantar pequeños pedacitos de papel con un peine de plástico o una regla que se ha frotado vigorosamente con una toalla de papel. Hay dos (y sólo dos) tipos de carga eléctrica. Cada tipo de carga repele al mismo tipo de carga, pero atrae al tipo opuesto. Esto es, cargas iguales se repelen, pero cargas opuestas se atraen.

De igual forma conocemos la ley de conservación de la carga eléctrica, que establece que la cantidad neta de carga eléctrica producida en cualquier proceso es cero o, dicho de otra manera, no se puede crear o destruir una carga eléctrica neta. Si un objeto (o una región del espacio) adquiere una carga positiva, entonces una cantidad igual de carga eléctrica negativa se hallará en regiones u objetos cercanos.

Coulomb encontró que, si la distancia entre las cargas aumentaba, la fuerza disminuía con el cuadrado de la distancia entre ellas. Esto es, si la distancia se aumenta al doble, la fuerza disminuye a un cuarto de su valor original. Así, Coulomb concluyó que la fuerza que un pequeño objeto cargado ejerce sobre otro pequeño objeto cargado es proporcional al producto de la magnitud de la carga en un objeto, Q1, por la magnitud de la carga en el segundo objeto, Q2, e inversamente proporcional al cuadrado de la distancia r entre ellos. Como ecuación, escribimos la ley de Coulomb como:

F=K(Q1\*Q2)/r^2 rˆ21

Donde K es una constante de proporcionalidad: K=8.99X10^9 Nm2/C2.

La dirección de la fuerza eléctrica siempre es a lo largo de la línea recta que une las dos cargas. Si las dos cargas tienen el mismo signo, la fuerza en una de las cargas se manifiesta alejándose de la otra (las cargas se repelen entre sí). Si las dos cargas tienen signos opuestos, la fuerza en una de las cargas se dirige hacia la otra (las cargas se atraen).

La constante k en la ecuación 21-1 generalmente se escribe en términos de otra

constante, ϵ0, llamada permitividad del espacio vacío. Se relaciona con k mediante. La ley de Coulomb puede escribirse entonces como:

F=(1/4Πϵ0\*((Q1\*Q2))/r^2 )

Campo eléctrico

En principio, podemos investigar el campo eléctrico que rodea a una carga o a un

grupo de cargas midiendo la fuerza sobre una pequeña carga de prueba positiva en reposo. Por carga de prueba queremos expresar una carga tan pequeña que la fuerza que ella ejerce no afecta significativamente a las cargas que generan el campo. Si se coloca una carga de prueba positiva pequeña q en varias posiciones, alrededor de una carga positiva individual Q. En particular, el campo eléctrico, en cualquier punto del espacio se define como la fuerza ejercida sobre una pequeña carga de prueba positiva localizada en ese punto y dividida entre la magnitud de la carga de prueba q:

E=F/q

Con mayor precisión, se define como el límite de , mientras q se hace más y más pequeña. Esto es, q es tan pequeña que, en esencia, no ejerce fuerza sobre las otras cargas que produjeron el campo eléctrico. El campo eléctrico en cualquier punto del espacio puede medirse usando la definición. Para situaciones simples que impliquen una o más cargas puntuales, podemos calcular Por ejemplo, el campo eléctrico a una distancia r de una carga individual Q tendría la magnitud.

E=F/q=(kqQ/r^2 )/q

E=k(Q/r^2 )

E=(1/4Πϵ0)\*(Q/r^2 )

Si se nos da el campo eléctrico en un punto dado del espacio, entonces podemos

calcular la fuerza sobre cualquier carga q localizada en ese punto escribiendo.

F=q(E)

Esta relación es válida aun si q no es pequeña, en tanto q no ocasione que se muevan las cargas que producen el campo Si q es positiva, y apuntan en la misma dirección. Si q es negativa, y E apuntan en direcciones opuestas.

Potencial eléctrico

El potencial eléctrico se define como la energía potencial por unidad de carga. Para aplicar la conservación de la energía necesitamos definir la energía potencial eléctrica, como lo hicimos para otros tipos de energía potencial. La energía potencial puede definirse sólo para fuerzas conservativas. El trabajo que efectúa una fuerza conservativa sobre un objeto en movimiento, entre dos posiciones cualesquiera, es independiente de la trayectoria que siga el objeto. La fuerza electrostática entre dos cargas es conservativa, ya que la dependencia en la posición es justo como la dependencia en la posición en el caso de la fuerza gravitacional, la cual es una fuerza conservativa. Así que podemos definir también la energía potencial U para la fuerza electrostática.

Definimos el cambio en la energía potencial eléctrica, Ub 2 Ua, cuando una carga puntual q se mueve de un punto “a” a otro punto “b”, como el negativo del trabajo que efectúa la fuerza eléctrica para mover la carga desde a hasta b. El trabajo W que efectúa el campo eléctrico E para mover la carga una distancia d es:

W=Fd=qEd

El cambio en energía potencial eléctrica es igual al negativo del trabajo realizado por la fuerza eléctrica:

Ub-Ua=-w=-qEd

De manera similar, es útil definir el potencial eléctrico (o simplemente el potencial, cuando se entiende que es “eléctrico”), como la energía potencial eléctrica por unidad de carga. Usamos el símbolo V para representar el potencial eléctrico. Si una carga de prueba positiva q tiene una energía potencial eléctrica Ua en algún punto a (con respecto a algún nivel cero de energía potencial previamente establecido), el potencial eléctrico Va en este punto es:

Va=Ua/q

Sólo la diferencia de potencial o la diferencia en el potencial, entre dos puntos a y b, es susceptible de ser medida. Cuando la fuerza eléctrica efectúa trabajo positivo sobre la carga, la energía cinética aumenta y la energía potencial disminuye. La diferencia en energía potencial, Ub - Ua, es igual al negativo del trabajo,Wba, que realiza el campo eléctrico para mover la carga desde a hasta b; por lo mismo, la diferencia de potencial Vba es:

Va=ΔV=Vb-Va=(Ub-Ua)/q=-Wba/q

Observe que el potencial eléctrico, al igual que el campo eléctrico, no depende de nuestra carga de prueba q. V depende de las otras cargas que generan el campo, pero no de q; q adquiere energía potencial cuando está inmersa en el potencial V debido a otras cargas.

Formulario

F=K(Q1\*Q2)/r^2 rˆ21

F=(1/4Πϵ0\*((Q1\*Q2))/r^2 )

E=F/q

E=F/q=(kqQ/r^2 )/q

E=k(Q/r^2 )

E=(1/4Πϵ0)\*(Q/r^2 )

F=q(E)

W=Fd=qEd

Ub-Ua=-w=-qEd

Va=Ua/q

Va=ΔV=Vb-Va=(Ub-Ua)/q=-Wba/q

Capacitancia:

Para este apartado sirve para calcular la capacitancia equivalente de una fracción del circuito, su principal propósito consiste en simplificar la tarea del estudiante en la resolución de problemas que tengan que ver con la materia. Para trabajar estas especificaciones utilizamos tres variables únicamente en la matriz gráfica. El primero de ellos es el tipo de circuito con el que trabajamos, ya sea en serie o paralelo, prefijo para saber cómo estamos trabajando las unidades de medida y la cantidad de capacitores, dependiendo de cuantos sean el programa puede realizar el cálculo en un instante y lo arroja en la fila del resultado.

|  |  |
| --- | --- |
| Clase: Electricidad\_y\_magnetismo | Se encuentra en las líneas  2521-2595 |
| Codificación |

¿Qué es un capacitor?

Un capacitor es un aparato que sirve para almacenar energía eléctrica; por lo general, consiste en dos objetos conductores (placas u hojas), colocados uno cerca del otro, pero sin tocarse. Los capacitores son ampliamente utilizados en circuitos electrónicos. Permiten almacenar energía eléctrica que habrá de usarse posteriormente (por ejemplo, en el flash de una cámara fotográfica y para almacenar energía en computadoras cuando falla la corriente eléctrica). Los capacitores también sirven para bloquear picos de carga y energía con la finalidad de proteger circuitos. Las computadoras usan capacitores muy delgados para la memoria de “unos” y “ceros” del código binario en la memoria de acceso aleatorio (RAM).

Un capacitor simple consiste en un par de placas paralelas de área *A* separadas por una pequeña distancia *d* (Figura a) Por lo general, las dos placas están enrolladas en forma de cilindro con papel, plástico u otro aislante para separar las placas (figura b). En un diagrama, el símbolo

Imagen que contiene Texto

Descripción generada automáticamente Diagrama, Esquemático

Descripción generada automáticamente Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

Si se aplica un voltaje a través de un capacitor conectado a una batería mediante cables conductores, las dos placas se cargan rápidamente: una placa adquiere carga negativa y la otra una carga positiva de la misma magnitud. Cada terminal de la batería y la placa del capacitor conectada a ella están al mismo potencial, así que el voltaje completo de la batería aparece a través del capacitor. Para un capacitor dado, se encuentra que la cantidad de carga Q que adquiere una de las placas es proporcional a la magnitud de la diferencia de potencial V entre las placas.

Texto

Descripción generada automáticamente con confianza media

La constante de proporcionalidad C, en la relación anterior, se llama la capacitancia del capacitor. La unidad de capacitancia es el coulomb entre volt, y recibe el nombre de farad (F). Los capacitores ordinarios tienen una capacitancia en el intervalo de 1 pF (picofarad 5=10^-2 F) a 10^3 mF (microfarad = 10^-6 F). La relación representada por la estas mecuaciones fue propuesta inicialmente por Alejandro Volta a fines del siglo XVIII. En general, la capacitancia C no depende de Q ni de V. Su valor depende del tamaño, la forma y la posición relativa entre los dos conductores, así como del material que los separa.

Capacitores en serie y en paralelo

Los capacitores se encuentran en muchos circuitos eléctricos. Por circuito eléctrico entendemos una trayectoria cerrada de conductores —generalmente alambres que conectan capacitores u otros dispositivos— por los cuales puede fluir carga y que incluyen una fuente de voltaje, como una batería, por ejemplo. El voltaje de la batería por lo general se representa con el símbolo *V*, lo que significa que *V* representa una *diferencia* de potencial. Se pueden conectar capacitores entre sí de distintas maneras. Las formas más comunes son en *serie* o en *paralelo.*

Se dice que los capacitores están conectados en “paralelo” porque cuando se conecta una batería de voltaje V a los puntos a y b, este voltaje V = Vab existe a través de cada uno de los capacitores. Esto es, ya que todas las placas del lado izquierdo de todos los capacitores se conectan mediante conductores, se encuentran al mismo potencial Va cuando se conectan a la batería; de igual modo, las placas del lado derecho de cada capacitor están al mismo potencial Vb. Cada placa de capacitor adquiere una carga dada por Q1 = C1V, Q2 = C2V y Q3 = C3V. La carga total Q que debe abandonar la batería es, por lo tanto:

Q = Q1 + Q2 + Q3 = C1 V + C2 V + C3 V.

Encontraremos un capacitor equivalente único con la misma carga Q al mismo voltaje V = Vab. Éste tendría una capacitancia Ceq dada por

Q = Ceq\*V.

Combinando las dos ecuaciones anteriores tenemos,

Ceq\*V = C1 V + C2 V + C3 V = (C1 + C2 + C3)V

o Ceq=(C1+C2+C3)

Diagrama, Esquemático

Descripción generada automáticamente o [en paralelo]. El efecto neto de conectar capacitores en paralelo es incrementar la capacitancia. Lo anterior tiene sentido porque básicamente estamos incrementando el área de las placas donde puede acumularse carga (véase, por ejemplo, la ecuación 24-2).

Capacitores en paralelo

Los capacitores también pueden conectarse en serie; esto es, extremo con extremo. Una carga 1Q fluye de la batería a una de las placas de C1, y una carga –Q fluye a una de las placas de C3. Las regiones A y B entre los capacitores estaban inicialmente neutras, así que la carga neta debe ser cero. La carga 1Q en la placa izquierda de C1 atrae una carga –Q en la placa opuesta. Como la región A debe tener una carga neta cero, hay, por lo tanto, una carga 1Q en la placa izquierda de C2. La misma consideración se aplica a los otros capacitores; así, vemos que la carga en cada capacitor posee el mismo valor Q. Un capacitor sencillo capaz de remplazar estos tres capacitores en serie sin afectar el circuito (esto es, con los mismos valores de Q y V) tendría una capacitancia equivalente Ceq, donde

Q = Ceq\*V

Ahora, el voltaje total V a través de los tres capacitores en serie debe ser igual a la suma de los voltajes a través de cada capacitor:

V = V1 + V2 + V3

También se cumple para cada capacitor que Q = C1V1, Q = C2V2 y Q = C3V3, así que sustituimos V, V1, V2 y V3 en la última ecuación y obtenemosImagen de la pantalla de un celular de un mensaje en letras negras

Descripción generada automáticamente con confianza baja

O  
Gráfico, Gráfico de cajas y bigotes

Descripción generada automáticamente

Observe que la capacitancia equivalente Ceq es menor que la capacitancia más pequeña utilizada.

Formulario

Q=𝐶𝑉

C=𝑄/𝑉

V=𝑄/𝐶

Circuitos en paralelo

Q = Q1 + Q2 + Q3 = C1 V + C2 V + C3 V.

Q = Ceq\*V.

Circuitos en serie

Ceq\*V = C1 V + C2 V + C3 V = (C1 + C2 + C3)V o Ceq=(C1+C2+C3)

V = V1 + V2 + V3

(Q/Ceq)=[(Q/C1)+(Q/C2)+(Q/C3)]=Q[(1/C1)+(1/C2)+(1/C3)] o (1/Ceq)=[(1/C1)+(1/C2)+(1/C3)]

Analisis\_combinatorio:

Este módulo condensa un tema visto durante matemáticas discretas y tiene como finalidad el reducir el tiempo de cálculo con las expresiones más sencillas del Análisis combinatorio que se ven durante esta materia. El módulo se divide en 6 sección con las que el usuario puede operar: permutaciones ordinarias, permutaciones con repetición, variación ordinaria, variación con repetición, combinación ordinaria, combinación con repetición. Este módulo es el único que posee una tabla pedagógica para el usuario, con la finalidad de que la comprensión que le otorgue sea la mejor para que comprenda el tema.

|  |  |
| --- | --- |
| Clase: Analisis\_combinatorio | Se encuentra en las líneas  2129-2431 |
| Codificación |

**Codificación e implementación**

En Software desarrolla se transcribió del lenguaje C a C++ y se desarrollará según las estructuras y procesos del entorno de desarrollo Dev-C++ y pensado para equipos que posean como sistema operativo Windows, ya que se aprovecharán las rutinas de este para llevar a cabo acciones como “CLS” para limpiar la pantalla o “Sleep” para algunas de las animaciones que posee el programa.

Las librerías que se utilizarán serán stdio.h, conio.h, windows.h, math.h, ctype.h y string.h, iostream y locale. Ya que proporcionan todas las funciones que se necesitarán para desarrollar el programa.

Con una extensión de 2706 líneas de código el programa se encuentra enteramente desarrollado, puede consultarse directamente la codificación e implementación de este en el archivo “MJC-Proyecto final Versión 2.0.cpp”.

**Pruebas modulares e integrales:**

Aunque no cuento con todas las versiones previas a la implementación final del programa, los archivos correspondientes al profesor par que tenga una idea de las varias optimizaciones y actualizaciones que ha tenido el programa desde el inicio. Dentro de la carpeta “Pruebas”, se podrán observar los prototipos que no quedaron en el producto final y varios que se modificaron tanto que no terminan de ser meras ideas que se quedaron al aire.

Para las pruebas modulares se ejecutará la introducción de datos que demuestren que la lógica de los argumentos no falla en ningún sentido y que el programa es completamente funcional.

Las pruebas sobre el funcionamiento del programa consistieron en estableces una serie de problemas concernientes a cada uno de los módulos que demostraran el correcto funcionamiento de los mismos, a continuación se pueden consultar las tablas de resultados.

Pruebas de calculadora.

|  |  |  |  |
| --- | --- | --- | --- |
| Operación | Resultado obtenido. | Resultado esperado. | Evaluación |
| Suma (5+10) | 15.0000 | 15 | Cumple |
| Resta (85-47) | 38.0000 | 38 | Cumple |
| Multiplicación(6.7\*9.1) | 60.9700 | 60.97 | Cumple |
| División (1889/489) | 3.8630 | 3.8629 | Cumple/falta precisión |
| Residuo (50/4) | 2.0000 | 2 | Cumple |
| Potencia (6^8) | 1679616.0000 | 1679616 | Cumple |
| Raíz (3|27) | 3.0000 | 3 | Cumple |
| Logaritmo natura (1l1) | 0.0000 | 0 | Cumple |
| Logaritmo 10 (2l10) | 1.0000 | 1 | Cumple |
| Seno G (1S90) | 1.0000 | 1 | Cumple |
| Seno R (2Spi/2) | 1.0000 | 1 | Cumple |
| Arcoseno G (3S1) | 90.0000 | 90 | Cumple |
| Arcoseno R (4S1) | 1.5707 | 1.5707. | Cumple |
| Coseno G (1C0) | 1.0000 | 1 | Cumple |
| Coseno R (2Cpi/2) | 0.0000 | 0 | Cumple |
| Arcocoseno G (3C.5) | 60.0000 | 60 | Cumple |
| Arcocoseno R (4C1) | 0.0000 | 0 | Cumple |
| Tangente G (1T45) | 1.0000 | 1 | Cumple |
| Tangente R (2Tpi) | 0.0000 | 0 | Cumple |
| Arcotangente G (3T.5) | 26.5651 | 26.5650 | Cumple/falta precisión |
| Arcotangente R (4T1) | 0.78541 | 0.7853 | Cumple/falta precisión |
| Cotangente G (-1T45) | 1.0000 | 1 | Cumple |
| Cotangente R (-2Tpi/9) | 2.7475 | 2.7474 | Cumple/falta precisión |
| Secante G (-1S0) | 1.0000 | 1 | Cumple |
| Secante R (-2Spi) | -1.0000 | -1 | Cumple |
| Cosecante G (-1C0) | Math Error. | Math error. | Cumple |
| Cosecante R (-2Cpi) | 1.0000 | 1.0000 | Cumple |

Pruebas de factorización:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Coeficiente A | Coeficiente B | Coeficiente C | Factorización | Evaluación |
| 0 | 0 | 0 | 0 | Cumple |
| 0 | 0 | 5 | Constante | Cumple |
| 0 | 5 | 0 | No se factoriza | Cumple |
| 0 | 5 | 25 | Primer grado | Cumple |
| 1 | 0 | 0 | No se factoriza | Cumple |
| 3 | 0 | -36 | (sqrt(3)x+6)(sqrt(3)x-6) | Cumple |
| 6 | 36 | 0 | 6.0x(x+6.0) | Cumple |
| 1 | 2 | 5 | Raíces imaginarias | Cumple |
| -7 | 5 | 2 | -(7x+2)(x-1) | Cumple |
| -1 | 2 | 1 | No factoriza | No Cumple |
| 7 | -5 | -2 | (x-1)(7x+2) | Cumple |
| 2 | 4 | -6 | (1x-1)(2x+6) | Cumple |

Pruebas de complejos:

|  |  |  |
| --- | --- | --- |
| Coeficiente C | Resultado | Evaluación |
| (8+i9)+(10-i9) | 18 | Cumple |
| (19+i8)-(100-i17) | -81.00+i25.00 | Cumple |
| (7.5+i8)\*(10+i9) | 3+i(295/2) | Cumple |
| (7+i5)/(15-i7) | (35/137)+i(62/137) | Cumple |
| (8+i9)/(0-i0) | Error: denominador igual a 0. | Cumple |

Pruebas de traductor:

|  |  |  |
| --- | --- | --- |
| Texto a traducir: | Código | Evaluación |
| Espero pasar el semestre. | 01000101 01110011 01110000 01100101 01110010 01101111 00100000 01110000 01100001 01110011 01100001 01110010 00100000 01100101 01101100 00100000 01110011 01100101 01101101 01100101 01110011 01110100 01110010 01100101 00101110 | Cumple |
| Todo verdor perecerá dijo la voz de la escritura como siempre implacable. | 01010100 01101111 01100100 01101111 00100000 01110110 01100101 01110010 01100100 01101111 01110010 00100000 01110000 01100101 01110010 01100101 01100011 01100101 01110010 10100000 00100000 01100100 01101001 01101010 01101111 00100000 01101100 01100001 00100000 01110110 01101111 01111010 00100000 01100100 01100101 00100000 01101100 01100001 00100000 01100101 01110011 01100011 01110010 01101001 01110100 01110101 01110010 01100001 00100000 01100011 01101111 01101101 01101111 00100000 01110011 01101001 01100101 01101101 01110000 01110010 01100101 00100000 01101001 01101101 01110000 01101100 01100001 01100011 01100001 01100010 01101100 01100101 00101110 | Cumple |

|  |  |  |
| --- | --- | --- |
| Código | Texto | Evaluación |
| 01011001 00100000 01110000 01101111 01110010 00100000 01110100 01110101 00100000 01110010 01101111 01110011 01110100 01110010 01101111 00100000 01110011 01101001 01101110 01100011 01100101 01110010 01101111 00100000 01111001 00100000 01110100 01110101 00100000 01110000 01100001 01110011 01101111 00100000 01110110 01100001 01100111 01100001 01100010 01110101 01101110 01100100 01101111 00100000 01111001 00100000 01110100 01110101 00100000 01101100 01101100 01100001 01101110 01110100 01101111 00100000 01110000 01101111 01110010 00100000 01100101 01101100 00100000 01101101 01110101 01101110 01100100 01101111 00100000 01110000 01101111 01110010 01110001 01110101 01100101 00100000 01110011 01101111 01110011 00100000 01110000 01110101 01100101 01100010 01101100 01101111 00100000 01110100 01100101 00100000 01110001 01110101 01101001 01100101 01110010 01101111 00101110 | Y por tu rostro sincero y tu paso vagabundo y tu llanto por el mundo porque sos pueblo te quiero. | Cumple |
| 01001110 01101111 00100000 01101100 01101111 00100000 01100011 01110010 01100101 01101111 00100000 01110100 01101111 01100100 01100001 01110110 10100001 01100001 00100000 01100101 01110011 01110100 10100000 01110011 00100000 01101100 01101100 01100101 01100111 01100001 01101110 01100100 01101111 00100000 01100001 00100000 01101101 01101001 00100000 01101100 01100001 01100100 01101111 00100000 01111001 00100000 01101100 01100001 00100000 01101110 01101111 01100011 01101000 01100101 00100000 01100101 01110011 00100000 01110101 01101110 00100000 01110000 01110101 10100100 01100001 01100100 01101111 00100000 01100100 01100101 00100000 01100101 01110011 01110100 01110010 01100101 01101100 01101100 01100001 01110011 00100000 01111001 00100000 01100100 01100101 00100000 01100001 01101100 01100101 01100111 01110010 10100001 01100001 | No lo creo todavía estás llegando a mi lado y la noche es un puñado de estrellas y de alegría | Cumple |

Pruebas Order 66:

Para las pruebas que se establecieron en este apartado fueron todas implementadas y por testers que no reportaron tener problemas con el juego y se consideró que todas las paredes quedaron mapeadas de buena manera para evitar que el juego se viera comprometido por cualquier exploiter que quisiera aprovechar errores de diseño para saltar muros, romper el juego o pervertir el carácter lúdico del juego.

Cargas puntuales

|  |  |  |
| --- | --- | --- |
| Entradas: | Salidas: | Evaluación |
| C1=5 C2=-5 Ángulo=0 | F=-89.9N Fx=89.9N Fy=0 | Cumple |
| C1=15M C2=16M Ángulo=45 | F=2157600N Fx=1525.6538N Fy=1525.6538N | Cumple |
| C1=18m C2=-15m Ángulo=0 | F=2.4273N  Fx=2.4273N  Fy=0 | Cumple |
| C1=6C C2=-6C Ángulo=30 | F=129456000  Fx=-1121121.9172N Fy=-647279.8779N | Cumple |
| C1=5M C2=-5M Ángulo=0 | F=224750.0000 Fx=224750.0000  Fy=0 | Cumple |
| C1=200 C2=200 Ángulo=45 | F=22.4750  Fx=15.8922N Fy=15.8922 | Cumple |

Pruebas Análisis combinatorio:

|  |  |  |
| --- | --- | --- |
| Entradas: | Salidas: | Evaluación |
| Pn=3 | 6 | Cumple |
| PR -> 2/3 | 3 | Cumple |
| V-> 2/3 | 6 | Cumple |
| VR->2/3 | 9 | Cumple |
| C->2/3 | 3 | Cumple |
| CR->2/3 | 6 | Cumple |

Mantenimiento:

El programa aún presenta errores y se planea su mantenimiento semestral en la medida que yo como programador avance en los créditos de la carrera, se realizarán actualizaciones y se pulirán los apartados en la medida que sean utilizados a futuro. En este sentido el programa no se reescribirá en otro lenguaje de aquí en adelante, siendo C++ el lenguaje último para él. De igual forma aquellos módulos transcritos del lenguaje C serán revisados detenidamente para que su codificación sea fiel al paradigma de la programación orientada objetas al que se migro.

**Conclusión:**

Considero que el software desarrollado durante el semestre cumple con las expectativas sobre manera, y el mismo fue creciendo al punto de sobrepasar los planes principales y convertirse en algo completamente distinto a lo que se imaginó al inicio del proyecto. Este proyecto crece cada tanto para mostrarnos que no hay límite en la programación. De igual forma, en la medida que el proyecto avanzó, mis habilidades para codificar y resolver problemas en lenguaje c y C++ estuvieron mejorando. Este proyecto me enseñó que un desarrollo planeado y estratégico, puede rendir frutos más allá de lo imaginado y ser de gran utilidad en mis estudios.