1. **Problema 1**: Escriba una función llamada múltiplo que tome dos enteros y determine si el segundo es múltiplo del primero. La función deberá tomar dos argumentos enteros y devolver 1 si el segundo es un múltiplo del primero y 0 si no. Luego, utilice esta función en un programa que acepte como entrada una serie de pares de enteros.
2. **Problema 2**: Escribir un programa que lea dos números **x** y **n** y en una función, calcule la suma de la progresión geométrica:

1 + x + x^2 + x^3 + x^4 + ⋯ + x^n

1. **Problema 3**: Dado el siguiente código fuente:

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

int main () {

int N, M, cant, i, aleatorio;

printf("Digite la cantidad de números que desea generar: ");

scanf("%d",&cant);

printf("Digite los limites (primero el superior, luego el inferior): ");

scanf("%d%d",&N,&M);

srand(time(NULL)); // Inicializacion del generador

for(i = 0; i < cant; i ++) {

aleatorio = rand()%(N-M+1)+M; //Genera un numero entre M y N

printf("%d ", aleatorio);

}

printf("\n", aleatorio);

system("PAUSE");

return 0;

}

a. Compile y ejecute el código fuente anteriormente mostrado. ¿Qué es lo que hacen las funciones srand() y rand()? (Los singientes enlaces pueden serle de utilidad: [1](http://www.chuidiang.org/clinux/funciones/rand.php) y [2](http://arantxa.ii.uam.es/~swerc/ole/ejemplos/crandom.html))

b. Escriba una función que genere un número aleatorio entre a y b. **Ayuda**: Use las funciones **srand()** y **rand()** anteriormente mencionadas.

c. Realice un programa que invocando la función anteriormente creada, funcione de manera similar al programa analizado en el punto a.

1. **Problema 4**: Escriba una función que tome un valor entero y devuelva el número con sus dígitos en reversa. Por ejemplo, dado el numero 7631, la función deberá devolver 1367.
2. **Problema 5**: Una ecuación cuadrática es aquella que tiene la siguiente forma:

a\*x^2 + b\*x + c = 0

El discriminante es una expresión que se forma a partir de los coeficientes del polinomio siguiendo la siguiente fórmula:

D = b^2 - 4\*a\*c

Según el valor del discriminante se pueden dar tres posibles situaciones:

* **La ecuación cuadrática tiene como solución dos raíces reales y distintas**: Lo cual se da cuando **D > 0**.
* **La ecuación cuadrática tiene dos raíces reales iguales**: Lo cual se da cuando **D = 0**.
* **La ecuación cuadrática tiene como solución dos raíces complejas conjugadas**: Lo cual se da cuando **D < 0**

a. Realizar una función (declaración y definición) que calcule el discriminante de una ecuación cuadrática. La función se deberá llamar discriminante y deberá recibir tres argumentos reales asociados a los coeficientes del polinomio cuadrático. Como valor de retorno esta debe devolver los siguientes valores para indicar cada uno de los tres posibles casos:

* Cuando D < 0 la función deberá retornar -1.
* Cuando D = 0 la función deberá devolver 0.
* Cuando D > 0 la función deberá devolver 1.

b. Una vez realizado lo anterior realizar un test breve de la función (el cual debe ser sustentado al docente) adaptando la declaración y la definición en el siguiente archivo fuente en el cual esta es invocada en el main:

#include <stdio.h>

// Aquí van los prototipos de la funciones

int main() {

int x = 1,y = 2,z = 6,w;

w = discriminante(1,0,1);

printf("w = %d\n",w);

w = discriminante(y,z,x);

printf("w = %d\n",w);

w = discriminante(y++,--z,5);

printf("w = %d\n",w);

return 0;

}

// Aquí van las definiciones de las funciones

c. Realizar un programa que solicite por teclado los valores de los coeficientes y de acuerdo a estos, despliegue si la ecuación cuadrática tiene raíces reales y distintas, reales e iguales o complejas conjugadas. Para tal fin se debe hacer uso de la función discriminante previamente creada.

1. **Problema 3**: Para obtener las raíces de una ecuación de segundo grado utilizando la fórmula:

x1 = (-b - D^(1/2))/(2\*a)

x2 = (-b + D^(1/2))/(2\*a)

a. Realizar una función que calcule y despliegue las raíces de una ecuación cuadrática. Tenga en cuenta que el valor de estas depende de D de tal modo que:

* Si D > 0 las raíces son:

x1 = (-b - D^(1/2))/(2\*a)

x2 = (-b + D^(1/2))/(2\*a)

Y la función deberá desplegar algo como: **x1, x2**

* Si D = 0, las raíces serán:

x1 = x2 = -b/(2\*a)

Y la función deberá desplegar algo como: **x1 con multiplicidad 2**.

* Si D < 0, las raíces serán:

x1 = (-b/(2\*a))i - ((-D)^(1/2))/(2\*a))j

x2 = (-b/(2\*a))i + ((-D)^(1/2))/(2\*a))j

Y la función deberá desplegar algo como: **R + Ij, R – Ij**.

La función anterior se deberá llamar bachiller, y esta simplemente realizara los cálculos de las raíces desplegando los resultados (no devolviéndolos porque son 2) por lo tanto el valor de retorno deberá ser tipo void. La función anterior deberá invocar la función del punto 1 para el cálculo del discriminante.

b. Invocar en el main la función varias veces de tal manera que la salida sea la mostrada en la siguiente captura de pantalla:

1 2 2

1 3 2

1 2 1

2 3 1

2.1 3.2 1.7

2 2 0.5

-1.00 + 1.00j, -1.00 - 1.00j

-2.00. -1.00

-1.00 con multiplicidad 2

-0.76 + 0.48j, -0.76 - 0.48j

-0.50 con multiplicidad 2

c. Realizar un programa que solicite al usuario los valores de los coeficientes y que haciendo uso de la función bachiller despliegue las raíces de estos coeficientes desplegados en pantalla.

6.**Problema 6**: Escriba un programa que simule el lanzamiento de una moneda. En cada línea de entrada aparece el número de lanzamientos de la moneda. En la línea de salida correspondiente debe aparecer una secuencia con el resultado de cada lanzamiento ('C' cuando fue cara y 'S' cuando fue sello) junto con el número de veces que aparece cada lado. El programa deberá usar una función que se encargue de simular el lanzamiento de la moneda una sola vez, que no tome argumentos y que retorne 0 para la cara ('C') ó 1 para el sello ('T').

Ayuda: Use las funciones srand() y rand() para simular el lanzamiento de la moneda.

Ejemplo de algunos casos de entrada:

585

Casos de salida correspondientes:

CCSSS, #de caras = 2, # de sellos = 3

SCCSCSCC, #de caras = 5, # de sellos = 3

SCSCC, # de caras = 3, # de sellos = 2

1. **Problema 7**: Las computadoras están jugando un papel creciente en la educación. Escriba un programa que ayudaría a un alumno de escuela primaria a aprender a multiplicar. Utilice rand() para producir dos enteros positivos de un dígito. A continuación debería imprimir una pregunta coma la siguiente:

¿Cuánto es 6 veces 7?

A continuación el alumno escribe la respuesta. Su programa verifica la respuesta del alumno. Si es correcta imprime **"Muy bien!"** y a continuación solicita otra multiplicación. Si la respuesta es incorrecta imprimirá **"No. Por favor intenta nuevamente"** y a continuación permitirá que el alumno vuelva a intentar con la misma pregunta de forma repetida hasta que al final la conteste correctamente. El programa debe indicarle al alumno una forma de terminar la ejecución.

1. **Problema 8**: La utilización de las computadoras en la educación se conoce como Instrucción Asistida por Computadora (CAI). Un problema que se desarrolla en los entornos CIA es la fatiga del alumno. Este problema puede ser enfrentado variando el diálogo de la computadora para retener la atención del alumno. Modifique el programa del ejercicio anterior de modo que este escoja de forma aleatoria uno de cuatro posibles mensajes tanto para respuestas correctas como para respuestas incorrectas. Los cuatro posibles mensajes en cada caso son:

Respuestas correctas:

Muy bien!

Excelente!

Buen trabajo!

Sigue haciéndolo bien!

Respuestas incorrectas:

No. Por favor trata de nuevo.

Incorrecto. Trata una vez más.

No te rindas!

No. Trata de nuevo

**Nota**: Mediante el generador de números aleatorios, seleccione un número entre 1 y 4 para desplegar un mensaje diferente para cada respuesta. Presente la respuesta mediante una estructura switch.

1. **Problema 9**: Sistemas más avanzados de CAI vigilan el rendimiento del alumno a lo largo de un periodo de tiempo. La decisión para empezar un tema nuevo se basa a menudo en el éxito del alumno en relación con temas anteriores. Modifique el programa del ejercicio anterior para contar el número de respuestas correctas e incorrectas del estudiante. Una vez el estudiante decida terminar la ejecución del programa, el programa debe calcular y mostrar el porcentaje de respuestas correctas respecto al total de preguntas que le hizo el programa. Si el porcentaje es menor a 75%, el programa deberá imprimir el mensaje "**Por favor pídele ayuda al instructor"** y termina.