#include <stdio.h>

#include <stdlib.h>

#include<string.h>

int main()

{

    /\*excerciseOfWhile();

    calculatorOfExponents();

    averageGrade();

    validGrade();

    printOdd();

    calculatorOfExponents2();

    printTen();

    printPyramid();

    printPyramid2(); \*/

    millionQuestion();

    return 0;

}

/\*\*Este Procedimiento Imprime la Cantidad de Asteriscos que el Usuario Requiera\*\*/

int excerciseOfWhile()

{

    int iAsterisk, iCounter = 0;

    printf("Ingresa la cantidad de asteriscos que requieres imprimir: ");

    scanf("%i", &iAsterisk);

    printf("Impresion de asteriscos: \n");

    while (iCounter < iAsterisk)

    {

        printf("\*");

        iCounter = iCounter + 1;

    }

    printf("\n \n");

    return 0;

}

/\*\*Este Procedimiento se Encarga de Resolver una Simple Operacion Exponencial, Mediante el Uso de un Ciclo WHILE\*\*/

int calculatorOfExponents()

{

    int iBase, iExp, iExp2, iResult;

    printf("Ingresa la base (entero) de la operacion exponencial: ");

    scanf("%i", &iBase);

    printf("Ingresa el exponente (entero) de la operacion exponencial: ");

    scanf("%i", &iExp);

    if (iExp > 0)

        iResult = iBase;

    else

        iResult = 1;

    iExp2 = iExp;

    while(iExp > 1)

    {

        iResult = iResult \* iBase;

        iExp = iExp - 1;

    }

    printf("%i ^ %i = %i", iBase, iExp2, iResult);

    printf("\n \n");

    return 0;

}

/\*\*Este Procedimiento se Encarga de Calcular el Promedio de las Calificaciones Ingresadas\*\*/

#include<stdio.h>

int averageGrade()

{

    float fGrade = 0, fAverage = 0;

    int iCounter = -1;

    printf("Comienza a ingresar calificaciones. Cuando ingreses una calificacion fuera del rango 0-100, tu promedio sera calculado sin tomar en cuenta esta ultima calificacion. \n \n");

    while (fGrade >= 0 && fGrade <= 100)

    {

        iCounter++;

        fAverage = fAverage + fGrade;

        printf("\tIngresa tu calificacion: ");

        scanf("%f", &fGrade);

        fflush(stdin);

    }

    printf("\nTu promedio es ");

    if (iCounter == 0)

        printf("0");

    else

        printf("%0.2f", fAverage / iCounter);

    printf("\n \n");

    return 0;

}

/\*\*Este Procedimiento Recibe una Calificacion e Indica Si Es Valida o No\*\*/

int validGrade()

{

    float fGrade;

    do {

            printf("Ingresa una calificacion: ");

            scanf("%f", &fGrade);

    } while (!(fGrade >= 0 && fGrade <= 100));

    printf("La calificacion %0.2f es valida. FIN. ", fGrade);

    printf("\n \n");

    return 0;

}

/\*\*Este Procedimiento Explora el Uso de los Ciclos de For\*\*/

int printOdd()

{

    int i;

    printf("Impresion de Numeros Impares del 1 al 100. \n \n-");

    for(i = 1; i <= 100; i = i + 2)

    {

        printf("%i-", i);

    }

    printf("\n \n");

    return 0;

}

/\*\*Este Procedimiento se Encarga de Resolver una Simple Operacion Exponencial, Mediante el Uso de un Ciclo FOR\*\*/

int calculatorOfExponents2()

{

    int iNum, iExp, iResult, f;

    printf("Ingresa la base de la operacion exponencial: ");

    scanf("%i", &iNum);

    printf("Ingresa el exponente de la operacion exponencial: ");

    scanf("%i", &iExp);

    iResult = 1;

    for (f = 1; f <= iExp; f++)

    {

        iResult = iResult \* iNum;

    }

    printf("Resultado: %i ^ %i = %i", iNum, iExp, iResult);

    printf("\n \n");

    return 0;

}

/\*\*Este Procedimiento Imprime 10 Veces Algun String\*\*/

int printTen()

{

    char cString[25];

    int iCounter = 1;

    printf("Ingresa una oracion para que se repita 10 veces: ");

    scanf("%[^\n]s", &cString);

    do{

            printf("\n %s", cString);

            iCounter++;

    }while(iCounter < 11);

    printf("\n \n");

    return 0;

}

/\*\*Este Procedimiento Imprime Una Piramide de Numeros, de Manera Secuencial\*\*/

int printPyramid()

{

    float fNumber;

    int iCounter = 1, f;

    printf("Ingresa un numero: ");

    scanf("%f", &fNumber);

    printf("Impresion de la Piramide Correspondiente a %0.0f \n \n", fNumber);

    if (fNumber - (int)fNumber < 0.50)

        fNumber = fNumber - (fNumber - (int)fNumber);

    else

        fNumber = fNumber - (fNumber - (int)fNumber) + 1;

    while (iCounter <= fNumber)

    {

        for (f = 1; f <= iCounter; f++)

        {

            printf("%i ", f);

        }

        iCounter++;

        printf("\n");

    }

    printf("\n \n");

    return 0;

}

/\*\*Este Procedimiento se Encarga de Imprimir una Piramide con los Componentes de un Numero\*\*/

int printPyramid2()

{

    float fNumber;

    int iCounter = 1, f;

    printf("Ingresa un numero: ");

    scanf("%f", &fNumber);

    if (fNumber - (int) fNumber < 0.5)

        fNumber = fNumber - (fNumber - (int) fNumber);

    else

        fNumber = fNumber - (fNumber - (int) fNumber) + 1;

    printf("Impresion de la Piramide Correspondiente a %0.0f \n \n", fNumber);

    while (iCounter <= fNumber)

    {

        for (f = 1; f <= iCounter; f++)

            printf("%i ", iCounter);

        printf("\n");

        iCounter++;

    }

    printf("\n \n");

    return 0;

}

/\*\*Este Procedimiento Determina Si es Mejor $1,000,000 o tener $0.10 Duplicado Diaramente por 30 Dias\*\*/

int millionQuestion()

{

    int iCounter = 1;

    float fResult = 0.1, fTotal;

    printf("Que es mejor tener, $1,000,000 en este instante, o $0.10 duplicados diariamente por 30 dias? \n \n \n");

    printf("\tPrimera Opcion\n\n\t+$1,000,000\n--------------------------------\n\t=$1,000,000\n\n\n\n");

    printf("\tSegunda Opcion\n\n");

    do{

        printf("\t+$%0.2f\n", fResult);

        fTotal = fTotal + fResult;

        fResult = fResult \* 2;

        iCounter++;

    }while (iCounter < 31);

    printf("--------------------------------\n\t$=%0.2f\n\n", fTotal);

    printf("\n \n");

    return 0;

}