

**UNIVERSIDAD NACIONAL AUTÓNOMA DE NICARAGUA
UNAN-LEÓN
DEPARTAMENTO DE MATEMÁTICA Y ESTADÍSTICA
CIENCIAS ACTUARIALES Y FINANCIERAS**



**TEMA: “DISEÑO DE UN COTIZADOR EN MICROSOFT EXCEL USANDO LA
APLICACIÓN VISUAL BASIC, PARA EL CÁLCULO DE LAS RENTAS
VITALICIAS”**

**MONOGRAFÍA PARA OPTAR AL TÍTULO DE LICENCIADO EN CIENCIAS
ACTUARIALES Y FINANCIERAS.**

ELABORADO POR:

BR. MARTHA MARÍA BUSTAMANTE MONJARREZ.
BR. ALEXIS LIZANDRO ORTIZ CASTILLO.
BR. CAROLINA ISABEL PALMA RAMOS.

TUTOR:

LIC. ROBERTO NOVOA

LEÓN, 2015



DEDICATORIA

A Dios por concederme la vida, brindarme fortaleza, sabiduría y perseverancia hasta culminar mi carrera con éxito y no haber desistido ante las adversidades.

Mis Padres que con su esfuerzo abnegado, amor incondicional y dedicación constante, me dieron lo necesario tanto espiritual como material para salir adelante y alcanzar mis metas.

Mis Hermanos por su comprensión, cariño, confianza y darme su apoyo en esos momentos difíciles.

Mis Amigos que de una u otra manera me brindaron su apoyo incondicional e hicieron olvidar esos momentos amargos de la vida.

Martha Bustamante Monjarrez.

Dios Padre Celestial por ser guía, inspiración, modelo y sobre todo por ser el ejemplo más grande de amor en este mundo.

Mi Mamita Concepción Sánchez (QEPD), persona que desde pequeño inculcó en mí amor, paz y perseverancia, ya que fue la base de mi vida para seguir adelante y luchar por mis sueños día a día.

Mis Padres por darme el ejemplo de vida a seguir, en especial a mi tía Elisa Ortiz por su ejemplo de perseverancia, amor y sobre todo por su apoyo incondicional.

Mis Hermanos por confiar y creer en mí.

Alexis Ortiz Castillo.

A Dios por darme la vida, fortaleza, sabiduría y sobretodo ser mi guía en mi vida para lograr culminar mis estudios.

A mi Madre por brindarme todo su apoyo, su amor incondicional, estar siempre presente en mi vida, luchar a diario y de esta forma darme una mejor vida.

A mi Hija, por ser mi fuente de inspiración, de lucha para seguir adelante y dar lo mejor de mí para su bienestar mismo.

A mis **Hermanos** y **Familiares**, por su apoyo y comprensión en este proceso de aprendizaje.

A mis **Amigos**, por su amistad, solidaridad y confianza a lo largo de nuestra vida universitaria.

Carolina Palma Ramos.



AGRADECIMIENTO

Agradecemos a:

Dios por bendecirnos con su infinito amor y habernos acompañado en este transcurso de nuestra vida, permitiéndonos compartir este momento de felicidad con las personas que creyeron y confiaron en nosotros.

Nuestros Padres por su esfuerzo para darnos un futuro mejor porque siempre estuvieron ahí brindándonos su apoyo incondicional y fueron ellos quienes estuvieron presentes en nuestra mente en cada paso que dimos.

Nuestro Tutor Lic. Roberto Novoa quien con sus enseñanzas y sabiduría supo guiarnos en el desarrollo de este trabajo investigativo y a el resto de profesores que con su paciencia nos instruyeron en la culminación de nuestra carrera.

Nuestros Amigos por ser con quienes compartimos estrés en esos largos días de estudios, pero también disfrutamos agradables momentos en nuestras instancias.



ÍNDICE

I.	INTRODUCCIÓN.	1
II.	OBJETIVOS.	3
III.	MARCO TEÓRICO	4
	CAPÍTULO I: ASPECTOS GENERALES DE LAS RENTAS VITALICIAS O CONTINGENTES.	4
	1.1 Definición	4
	1.2 Clasificación de las rentas vitalicias	5
	1.3 Elementos que constituye una renta vitalicia	6
	1.4 Fórmulas de las rentas vitalicias.	8
	CAPÍTULO II: ELEMENTOS DE MICROSOFT EXCEL	17
	2.1 Definición de Excel	17
	2.2 Definiciones de funciones de Excel.	18
	2.3 Descripción de algunas funciones de Microsoft Excel	20
	CAPÍTULO III: ELEMENTOS DE VISUAL BASIC.	24
	3.1 Definiciones	24
	3.2 Algunas Funciones de Visual Basic para Aplicaciones	26
IV.	DISEÑO METODOLÓGICO	35
V.	RESULTADOS.	36
	5.1 Diseño del programa en Excel.	36
	5.1.1 Generando las tablas de mortalidad.	36
	5.1.2 Elaboración del formulario base.	37
	5.1.3 Diseño y estructura del programa (macros) para el cálculo de rentas vitalicias.	41
	5.1.4 Descripción de los códigos utilizados.	53
	5.2 Resolución de ejercicios prácticos.	56
VI.	CONCLUSIONES	61
VII.	RECOMENDACIONES	62
VIII.	BIBLIOGRAFÍA	63
IX.	ANEXOS	64



I. INTRODUCCIÓN.

En Nicaragua las rentas vitalicias se usan como un fondo de ahorro, estas rentas se recomiendan a todas aquellas personas que dispongan de un ahorro y deseen rentabilizarlo sin riesgo, con el objetivo de percibirlo de forma diferida aumentando sus ingresos mensuales. Por lo tanto, es una solución idónea para personas jubiladas o cercanas a la jubilación, o para quienes tengan que hacer frente a una situación de dependencia y necesiten un complemento para hacer frente a los gastos que se ocasionen.

La renta vitalicia, en un sentido matemático, corresponde a una función que consiste en el valor presente de una unidad monetaria (constante o variable), pagadera en periodos regulares durante determinado plazo y sólo si durante dicho plazo se cumple la condición de supervivencia. Es por ello que la anualidad es la suma de valores presentes de una unidad ponderada con la probabilidad de supervivencia de personas o con la probabilidad de que una condición contingente se cumpla.

Haremos referencia también a los aspectos generales de las rentas vitalicias, tales como su definición, sus principales elementos o variables, sus diferentes clasificaciones como es de acuerdo a la naturaleza del capital, en el momento en que se realizan los pagos, en el periodo en que se inician las rentas, entre otras de igual importancia, y así mismo el proceso de valoración de las rentas, es decir, bajo qué términos matemáticos resultan las fórmulas de una renta contingente, para luego empezar a mencionar los tipos de rentas vitalicias y sus respectivas fórmulas.

Desarrollaremos los elementos de Microsoft Excel, sus definiciones y para qué sirve cada uno, así como sus diferentes funciones con el propósito de comprender y profundizar los conocimientos adquiridos de este software. Definiremos que es el programa Visual Basic, sus elementos y diferentes funciones que este nos presenta para un mejor entendimiento y para ser capaz de dar buen uso.



Tomando en cuenta la importancia del cálculo de las rentas vitalicias en nuestra área actuarial y en la misma sociedad, pretendemos diseñar en Excel un formato dirigido directamente al cálculo de los distintos tipos de rentas vitalicias el cual nos permita agilizar el trabajo a realizar. En este mismo sentido servirá como una herramienta básica no sólo para los estudiantes sino para toda aquella persona interesada en el programa.

Además considerando que la tendencia de globalización es más acentuada y como estrategia para enfrentar este nuevo escenario, hoy en día es necesario considerar la automatización de cálculo dentro del mercado laboral, porque esto permite mejorar el desempeño de las actividades realizadas, minimizando el tiempo y la posibilidad de error, lo que permitirá incrementar la productividad y la eficiencia de cualquier tipo de empresa.



II. OBJETIVOS

OBJETIVO GENERAL

- Diseñar un cotizador en Microsoft Excel usando la aplicación Visual Basic, para el cálculo de la prima de las rentas vitalicias.

OBJETIVOS ESPECÍFICOS

- Explicar los aspectos generales de las rentas vitalicias.
- Describir las diferentes funciones a utilizar en el programa Microsoft Excel.
- Plantear el algoritmo paso a paso de la elaboración del programa.
- Presentar el diseño y estructura del programa (macros) para el cálculo de las rentas vitalicias elaborado en Visual Basic.



III. MARCO TEÓRICO

CAPÍTULO I: ASPECTOS GENERALES DE LAS RENTAS VITALICIAS O CONTINGENTES.

1.1 Definición

En el ámbito financiero y comercial existen muchas operaciones en las que una serie de pagos periódicos se relaciona con su valor al comienzo o al término del plazo. Tales operaciones son conocidas como anualidades o rentas. Dentro de estas se encuentran las anualidades o rentas vitalicias, las cuales abordaremos como objeto de nuestro trabajo.

Para iniciar, las rentas vitalicias se pueden definir como un conjunto de capitales con vencimiento determinado, cuya exigencia o pago se produce si en ello se encuentra con vida una cabeza.

El concepto de anualidad vitalicia en términos matemáticos, corresponde a una función que consiste en el valor presente de una unidad monetaria (constante o variable), pagadera en periodos regulares durante determinado plazo y sólo si durante dicho plazo se cumple la condición de supervivencia. Es por ello que la anualidad es la suma de valores presentes de una unidad ponderada con la probabilidad de supervivencia de personas o con la probabilidad de que una condición contingente se cumpla. Es decir, la fecha del primer pago, la fecha del último pago, o ambas no se fijan de antemano; depende de algún hecho que se sabe que ocurrirá, pero no se sabe cuándo. Por ejemplo: Pensiones Privadas, Pensiones del Seguro Social, Pólizas de Seguros.

En rentas vitalicias, el término vitalicio en este caso no necesariamente se refiere a que dicha persona tendrá la renta hasta su muerte, sino mientras viva, que no es exactamente hasta su muerte, porque puede ser efectivamente hasta el último aniversario de su vida, como también por un plazo de “n” años.



1.2 Clasificación de las rentas vitalicias

De acuerdo a la determinación temporal

- 1) Temporales: cuando se conoce previamente el momento inicial y final de la operación.
- 2) Ilimitada: cuando no se puede definir previamente el momento final de la operación.

De acuerdo al periodo en que se inician las rentas

- 1) Inmediatas: Es el caso más común. La realización de los cobros o pagos tiene lugar en el periodo inmediatamente siguiente a la formalización del contrato.
- 2) Diferidas: La realización de los cobros o pagos se hace tiempo después de la formalización del contrato (existe un periodo de diferimiento).

De acuerdo con el momento en que realizan los pagos

- 1) Vencidas: son aquellas en que los pagos se efectúan a su vencimiento, es decir, al momento final de cada periodo.
- 2) Anticipadas: Los pagos se efectúan al principio de cada periodo.

De acuerdo a la naturaleza del Capital

- 1) Rentas Constantes: Todos sus capitales son iguales
- 2) Rentas Variables: Al menos uno de sus términos es diferente. Estas a su vez se dividen en:
 - a) Variable Aritméticamente: cuando el monto del pago se incrementa en un monto constante anual.
 - b) Variable Geométricamente: cuando el monto se incrementa en un porcentaje del pago anterior.

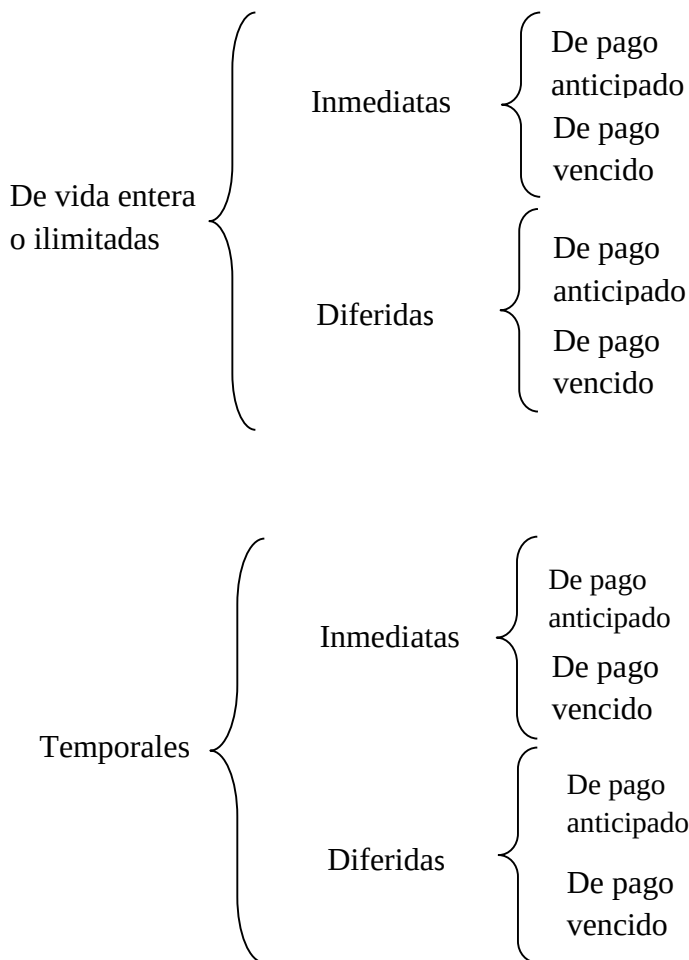
De acuerdo a su periodo de capitalización

- 1) Enteras: cuando las rentas se realizan con una periodicidad anual, en este caso se les denomina anualidades.



- 2) Fraccionadas: Cuando el periodo de capitalización es una parte (fracción) del año, en otras palabras las rentas se pagan en periodos menores a un año. Por ejemplo: meses, trimestres, semestres etc.

Para resumir de una manera más clara la clasificación de las rentas se muestra el siguiente cuadro sinóptico¹:



1.3 Elementos que constituye una renta vitalicia

Las rentas vitalicias constan principalmente de los siguientes elementos:

- ❖ **Renta (R):** Es el valor de cada pago periódico.

¹ Según el libro de Introducción al cálculo actuarial de Hugo E. Palacios, este mismo orden de clasificación siguen las rentas vitalicias tanto variables como fraccionarias.



- ❖ **Tiempo o plazo de una anualidad (n):** El intervalo de tiempo que transcurre entre el comienzo del primer período de pago y el final del último.
- ❖ **Edad de emisión (x):** es la edad a la que se emite el contrato.
- ❖ **Diferimiento (r):** es el periodo de gracia, es decir, el tiempo en el que no se paga la renta.
- ❖ **Tiempo fraccionario (m):** es el plazo menor a un año, por ejemplo: meses, trimestres, bimestres etc.
- ❖ **Razón de Crecimiento (h):** es una serie de pagos donde los valores de los pagos de cada período aumentan en una cantidad constante.
- ❖ **VAA:** se llama así al valor único o prima única que debe satisfacer hoy (a la edad x) la persona que desea percibir una renta mientras viva, sea cualquiera el tipo de renta.
- ❖ **Símbolos de conmutación:** estos no son más que artificios matemáticos que ayudan a simplificar los cálculos, sus valores son calculados con base a una determinada tabla de mortalidad y un interés técnico. Cabe señalar que para nuestra metodología de cálculo utilizaremos únicamente los siguientes conmutativos²:

D_x Numero de sobrevivientes descontados a una determinada tasa de interés anual por un tiempo equivalente a su edad. Su fórmula está dada por:

$$D_x = v^x * l_x ;$$

Donde;

$$v^x = (1 + i)^{-x}, \text{ llamado factor de actualización.}$$

l_x : Numero de sobrevivientes a la edad x.

N_x Es el conmutativo que se utiliza para representar los vivos conmutados en intervalos hasta al final de la tabla de mortalidad, siendo estos la sumatoria de los D_x . Resultando de la siguiente manera:

² Hugo E. Palacios; Introducción al cálculo actuarial. Madrid: MAPFRE; 1996.



$$N_x = D_x + D_{x+1} + D_{x+2} + D_{x+3} + \dots + D_{\omega-1} = \sum_{k=0}^{\omega-x-1} D_{x+k}$$

S_x Es el valor acumulado de los N_x , es decir, es la sumatoria de todas las N_x desde x hasta el final de la tabla. Matemáticamente se expresaría así:

$$S_x = N_x + N_{x+1} + N_{x+2} + N_{x+3} + \dots + N_{\omega-1} = \sum_{k=0}^{\omega-x-1} N_{x+k}$$

1.4 Fórmulas de las rentas vitalicias.

Para la valoración de las rentas vitalicias hay que tener en cuenta que a cada término de la renta se le asigna la probabilidad de que ocurra, que es justamente la probabilidad de que una cabeza de edad x sobreviva hasta el vencimiento del término de la renta.

Si planteamos la siguiente pregunta ¿Qué cantidad C de dinero han de pagar en este momento cada una de las l_x (personas vivas a la edad x), a cambio de recibir en este momento una unidad monetaria cada una de ellas?, por tanto el valor presente de las obligaciones de ese pago que se hará a cada uno de los que estén con vida a cada año será³:

$$C = v^0 \cdot l_{x+0} + v^1 \cdot l_{x+1} + v^2 \cdot l_{x+2} + \dots + v^{\omega-x-1} \cdot l_{\omega-x-1} = \sum_{k=0}^{\omega-x-1} v^k \cdot l_{x+k}$$

Donde $v^k = \frac{1}{(1+i)^k}$; esto no es más que el valor presente de una unidad monetaria en k periodo de tiempo a una cierta tasa de interés i .

Si dividimos el valor presente de esas obligaciones entre el número de personas que se encuentran con vida al inicio del tiempo, se llega a que:

³ José Antonio Gil Fana, Antonio Heras Martínez, José Luis Vilar Zanón; Matemática de los Seguros de Vida. Madrid: MAPFRE; 1999.



$$C = v^0 \cdot \frac{l_{x+0}}{l_x} + v^1 \cdot \frac{l_{x+1}}{l_x} + v^2 \cdot \frac{l_{x+2}}{l_x} + \dots + v^{\omega-x-1} \cdot \frac{l_{\omega-x-1}}{l_x} = \sum_{k=0}^{\omega-x-1} v^k \cdot \frac{l_{x+k}}{l_x}$$

Es decir, la anualidad es el valor presente de un valor ponderado de que una persona de edad x se encuentre con vida.

Dado que el planteamiento de las fórmulas de las rentas vitalicias será a través de conmutativo tendríamos que la fórmula anterior equivale a⁴:

$$\ddot{a}_x = \sum_{k=0}^{\omega-x-1} v^k \frac{l_{x+k}}{l_x} = \sum_{k=0}^{\omega-x-1} \frac{v^{x+k} \cdot l_{x+k}}{v^x \cdot l_x} \sum_{k=0}^{\omega-x-1} \frac{D_{x+k}}{D_x} = \frac{N_x}{D_x}$$

Dónde:

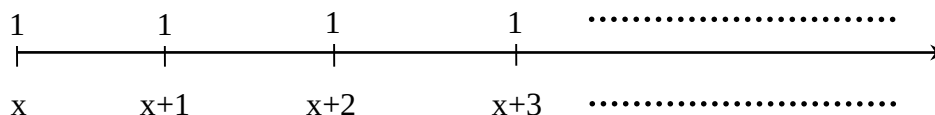
$\ddot{a}_x$ Es el símbolo actuarial que representa a una renta vitalicia ilimitada, inmediata y anticipada.

Este mismo razonamiento es el que se sigue para establecer las fórmulas de los otros tipos de rentas vitalicias que se presentan a continuación:

Rentas ilimitadas o de vida entera y constantes⁵

Como ya se señaló anteriormente las rentas ilimitadas se caracterizan porque son pagaderas sin limitación de tiempo mientras se cumpla la condición de supervivencia.

Ilimitada, inmediata y anticipada



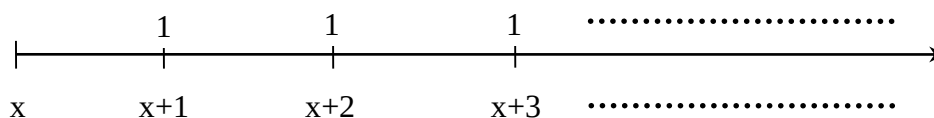
⁴ José Antonio Gil Fana, Antonio Heras Martínez, José Luis Vilar Zanón; Matemática de los Seguros de Vida. Madrid: MAPFRE; 1999.

⁵ Hugo E. Palacios; Introducción al cálculo actuarial. Madrid: MAPFRE; 1996.



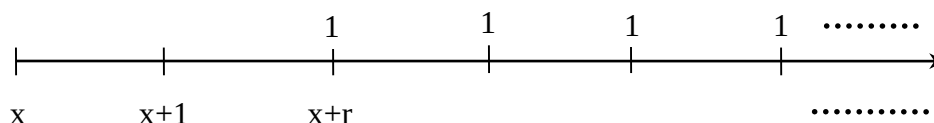
$$VAA = R * \ddot{a}_x = R * \left[\frac{N_x}{D_x} \right]$$

Ilimitada, inmediata y vencida



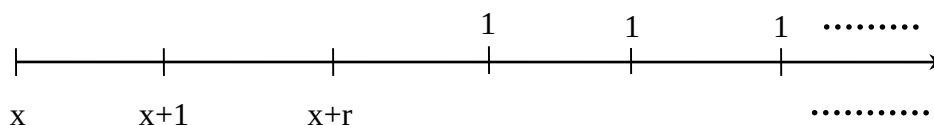
$$VAA = R * a_x = R * \left[\frac{N_{x+1}}{D_x} \right]$$

Ilimitada, diferida y anticipada



$$VAA = R * r / \ddot{a}_x = R * \left[\frac{N_{x+r}}{D_x} \right]$$

Ilimitada, diferida y vencida



$$VAA = R * r / a_x = R * \left[\frac{N_{x+r+1}}{D_x} \right]$$

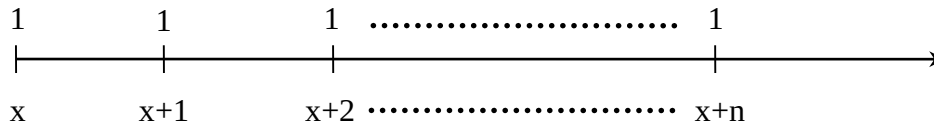
Rentas contingentes temporales y constantes⁶

A diferencia de las rentas ilimitadas, las temporales se caracterizan por ser pagaderas por un periodo determinado, siempre cumpliendo que la persona x este con vida. Siguiendo con las fórmulas tenemos:

⁶ Hugo E. Palacios; Introducción al cálculo actuarial. Madrid: MAPFRE; 1996.

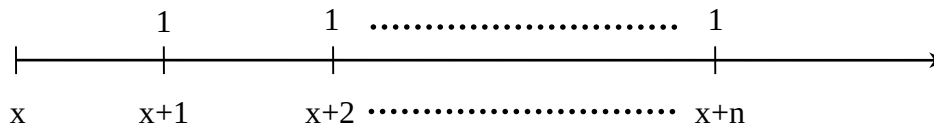


Temporal, inmediata y anticipada



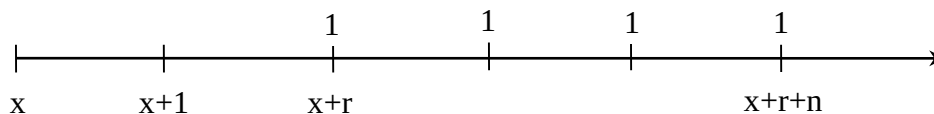
$$VAA = R * \ddot{a}_{x:n} = R * \left[\frac{N_x - N_{x+n}}{D_x} \right]$$

Temporal, inmediata y vencida



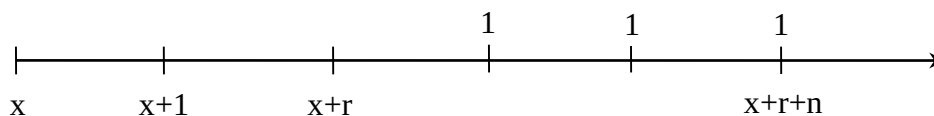
$$VAA = R * a_{x:n} = R * \frac{N_{x+1} - N_{x+n+1}}{D_x}$$

Temporal, diferida y anticipada



$$VAA = R * r/a_{x:n} = R * \left[\frac{N_{x+r} - N_{x+n+r}}{D_x} \right]$$

Temporal, diferida y vencida



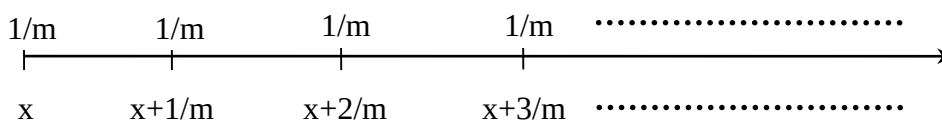
$$VAA = R * r/a_{x:n} = R * \left[\frac{N_{x+r+1} - N_{x+n+r+1}}{D_x} \right]$$



Rentas contingentes fraccionarias y constantes⁷

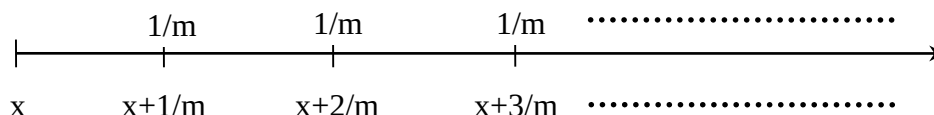
Como se mencionó anteriormente las rentas fraccionarias son aquellas en que los pagos se realizan en periodos menores a un año, dado que las tablas de mortalidad sólo presentan probabilidades anuales no se conocen las probabilidades exactas de muertes para fracciones del año, por tanto se recurren a hipótesis que conducen a expresiones aproximadas para los valores actuales actuariales, en nuestro caso solo haremos uso de la hipótesis de linealidad de D_x , obteniendo las siguientes fórmulas:

Fraccionaria, ilimitada, inmediata y anticipada



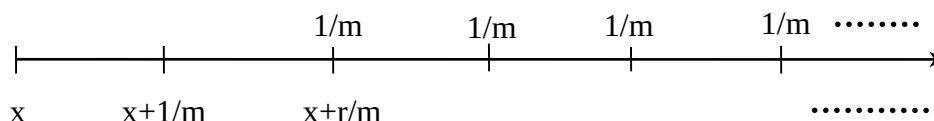
$$VAA = R(m) * \ddot{a}_x^{(m)} = R(m) * \left[\ddot{a}_x - \frac{m-1}{2m} \right] = R(m) * \left[\frac{N_x}{D_x} - \frac{m-1}{2m} \right]$$

Fraccionaria, ilimitada, inmediata y vencida



$$VAA = R(m) * a_x^{(m)} = R(m) * \left[a_x + \frac{m-1}{2m} \right] = R(m) * \left[\frac{N_{x+1}}{D_x} + \frac{m-1}{2m} \right]$$

Fraccionaria, ilimitada, diferida y anticipada



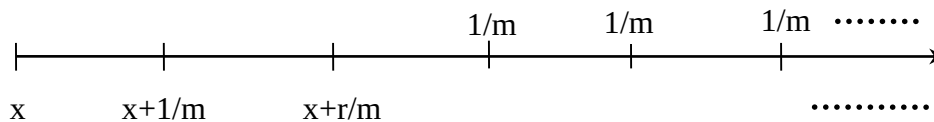
⁷ José Antonio Gil Fana, Antonio Heras Martínez, José Luis Vilar Zanón; Matemática de los Seguros de Vida. Madrid: MAPFRE; 1999.



$$VAA = R(m) * {}_r\ddot{a}_x^{(m)} = R(m) * \left[{}_r\ddot{a}_x - \frac{m-1}{2m} * {}_rE_x \right]$$

$$= R(m) * \left[\left(\frac{N_{x+r}}{D_x} \right) - \left(\frac{m-1}{2m} * \frac{D_{x+r}}{D_x} \right) \right]$$

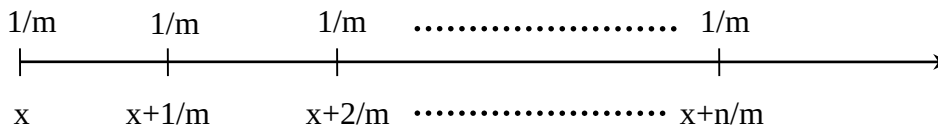
Fraccionarias, ilimitada, diferida y vencida



$$VAA = R(m) * {}_r a_x^{(m)} = R(m) \left[{}_r a_x + \frac{m-1}{2m} * {}_r E_x \right]$$

$$= R(m) \left[\left(\frac{N_{x+r+1}}{D_x} \right) + \left(\frac{m-1}{2m} * \frac{D_{x+r}}{D_x} \right) \right]$$

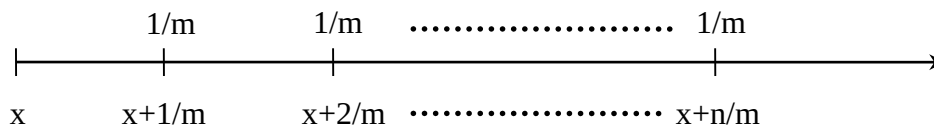
Fraccionaria, temporal, inmediata y anticipada



$$VAA = R(m) * \ddot{a}_{x:n}^{(m)} = R(m) \left[\ddot{a}_{x:n} - \frac{m-1}{2m} * (1 - {}_nE_x) \right]$$

$$= R(m) \left[\left(\frac{N_x - N_{x+n}}{D_x} \right) - \left(\frac{m-1}{2m} * \left(1 - \frac{D_{x+n}}{D_x} \right) \right) \right]$$

Fraccionaria, temporal, inmediata y vencida

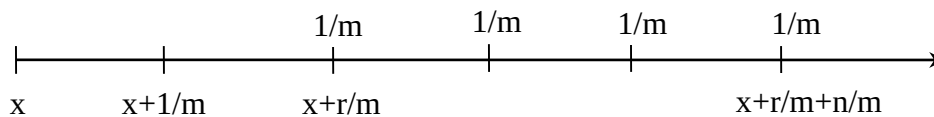




$$VAA = R(m) * a_{x:n}^{(m)} = R(m) \left[a_{x:n} + \frac{m-1}{2m} * (1 - {}_nE_x) \right]$$

$$= R(m) \left[\left(\frac{N_{x+1} - N_{x+n+1}}{D_x} \right) + \left(\frac{m-1}{2m} * \left(1 - \frac{D_{x+n}}{D_x} \right) \right) \right]$$

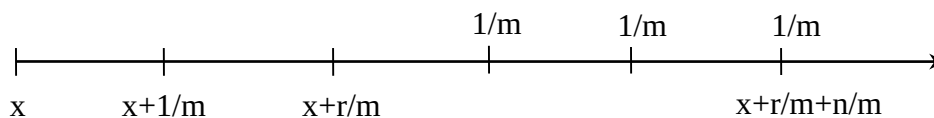
Fraccionaria, temporal, diferida y anticipada



$$VAA = R(m) * {}_{r/n}\ddot{a}_x^{(m)} = R(m) \left[{}_{r/n}\ddot{a}_x - \frac{m-1}{2m} * ({}_rE_x - {}_{r+n}E_x) \right]$$

$$= R(m) \left[\left(\frac{N_{x+r} - N_{x+r+n}}{D_x} \right) - \left(\frac{m-1}{2m} \left(\frac{D_{x+r}}{D_x} - \frac{D_{x+r+n}}{D_x} \right) \right) \right]$$

Fraccionaria, temporal, diferida y vencida



$$VAA = R(m) * {}_{r/n}a_x^{(m)} = R(m) \left[{}_{r/n}a_x + \frac{m-1}{2m} * ({}_rE_x - {}_{r+n}E_x) \right]$$

$$= R(m) \left[\left(\frac{N_{x+r+1} - N_{x+r+n+1}}{D_x} \right) + \left(\frac{m-1}{2m} \left(\frac{D_{x+r}}{D_x} - \frac{D_{x+r+n}}{D_x} \right) \right) \right]$$

Rentas vitalicias variables⁸

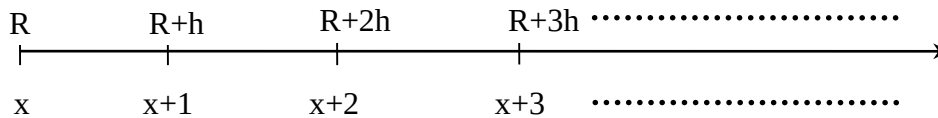
Las rentas vitalicias variables como su nombre lo indica, los pagos no son constante sino que cada término varia cada año, consideraremos en nuestro trabajo únicamente las rentas variables en progresión aritmética, tomando R como el primer término y h la razón de crecimiento. Las fórmulas serían las siguientes:

⁸ José Antonio Gil Fana, Antonio Heras Martínez, José Luis Vilar Zanón; Matemática de los Seguros de Vida. Madrid: MAPFRE; 1999.



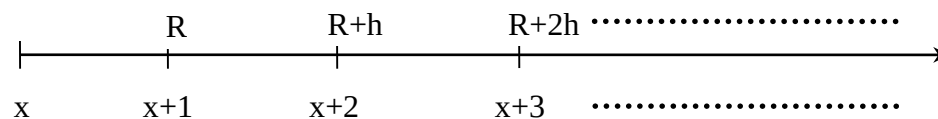
Rentas vitalicias variables crecientes en progresión aritmética

Ilimitada (VE), inmediata y anticipada



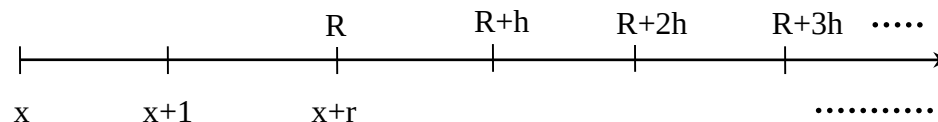
$$VAA = (V\ddot{a}C)_x = R * \left[\frac{N_x}{D_x} \right] + h * \left[\frac{S_{x+1}}{D_x} \right]$$

Ilimitada (VE), inmediata y vencida



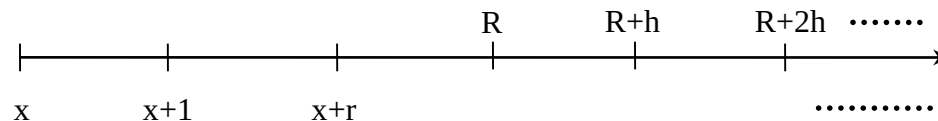
$$VAA = (VaC)_x = R * \left[\frac{N_{x+1}}{D_x} \right] + h * \left[\frac{S_{x+2}}{D_x} \right]$$

Ilimitada (VE), diferida y anticipada



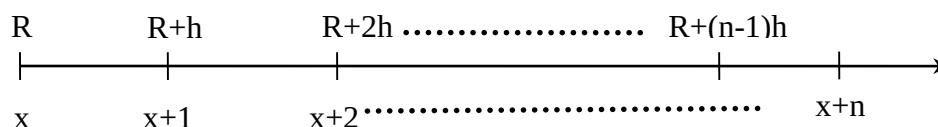
$$VAA = {}_{r/}(V\ddot{a}C)_x = R * \left[\frac{N_{x+r}}{D_x} \right] + h * \left[\frac{S_{x+r+1}}{D_x} \right]$$

Ilimitada (VE), diferida y vencida



$$VAA = {}_{r/}(VaC)_x = R * \left[\frac{N_{x+r+1}}{D_x} \right] + h * \left[\frac{S_{x+r+2}}{D_x} \right]$$

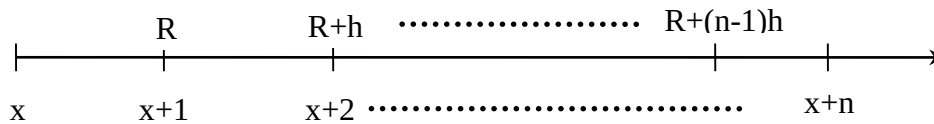
Temporal, inmediata y anticipada





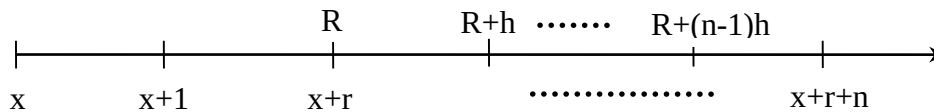
$$VAA = (V\ddot{a}C)_{x:n|} = R * \left[\frac{N_x - N_{x+n}}{D_x} \right] + h \left[\frac{S_{x+1} - S_{x+n} - (n-1)N_{x+n}}{D_x} \right]$$

Temporal, inmediata y vencida



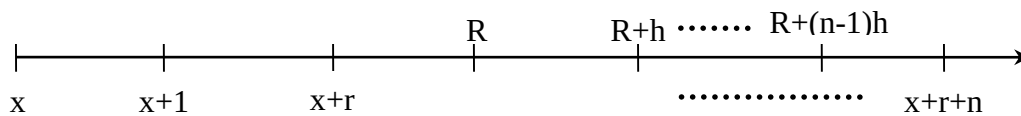
$$VAA = (VaC)_{x:n|} = R * \left[\frac{N_{x+1} - N_{x+n+1}}{D_x} \right] + h \left[\frac{S_{x+2} - S_{x+n+1} - (n-1)N_{x+n+1}}{D_x} \right]$$

Temporal, diferida y anticipada



$$VAA = {}_{r/n}(V\ddot{a}C)_x = R * \left[\frac{N_{x+r} - N_{x+r+n}}{D_x} \right] + h \left[\frac{S_{x+r+1} - S_{x+r+n} - (n-1)N_{x+r+n}}{D_x} \right]$$

Temporal, diferida y vencida



$$\begin{aligned} VAA &= {}_{r/n}(VaC)_x \\ &= R * \left[\frac{N_{x+r+1} - N_{x+r+n+1}}{D_x} \right] + h \left[\frac{S_{x+r+2} - S_{x+r+n+1} - (n-1)N_{x+r+n+1}}{D_x} \right] \end{aligned}$$



CAPÍTULO II: ELEMENTOS DE MICROSOFT EXCEL⁹

2.1 Definición de Excel

Microsoft Excel es un software para el manejo de hojas electrónicas agrupadas en *libros* para cálculos de casi cualquier índole. Entre muchas otras aplicaciones, es utilizado en el tratamiento estadístico de datos, así como para la presentación gráfica de los mismos, otro de sus usos más frecuentes es la creación y tratamiento de Bases de datos, para llevar registros extensos, en el área de contabilidad es usado por su fácil manejo de la información. La hoja electrónica Excel es ampliamente conocida, en forma generalizada, por profesionales y estudiantes en proceso de formación, pero hay una gran cantidad de usuarios que no conocen a profundidad su gran potencial y adaptabilidad a los diferentes campos del conocimiento.

Excel es empleado en muchos escenarios pero los más comunes se encuentran en Contabilidad, asesores financieros (definición de presupuestos), informes, planeación (clases, marketing, impuesto, gastos), seguimiento (listas de inventarios), uso de calendarios.

Microsoft Excel es una de las aplicaciones más utilizadas en el mundo y parte de su éxito se lo debe a la facilidad con que podemos crear fórmulas que nos permiten manipular y obtener los resultados que necesitamos de nuestra información.

Entre las tareas más básicas que se pueden realizar en este software podemos encontrar, crear un libro de cálculo, escribir datos en una hoja de cálculo, aplicar formatos a una hoja de cálculos, aplicar formatos a números y textos, gráficos en una hoja de cálculo, imprimir la información, crear una tabla de Excel, filtrar datos con autofiltros, ordenar datos usando autofiltros, utilizar formatos condicionales, utilizar validaciones de datos, usar y desarrollar fórmulas propias o ya establecidas por el programa, generar gráficos a partir de la

⁹Página web Excel Total

Excel Total. (s.f.). Recuperado el 10 de Octubre de 2014, de www.exceltotal.com



información agregada, diseñar tablas dinámicas, diseñar macros o complementos, que permitan una mejor experiencia del programa.

2.1.1 Hoja de cálculo: es un programa informático compuesto por columnas, filas y celdas, donde la intersección de las columnas y las filas son las celdas, en el interior de cada celda es el sitio donde podemos escribir cualquier tipo de información que posteriormente será tratada, siendo cada celda única en toda la hoja de cálculo.

2.2 Definiciones de funciones de Excel

Estas son una característica muy importante y básica para un programa de cálculo como Excel, a través de estas es que tomamos provecho de todo el poder de esta plataforma, estas funciones se utilizan para realizar tanto cálculos simples como avanzados, tomando los datos ingresados en las distintas celdas de las hojas de cálculo. Se pueden agrupar en categorías de acuerdo a su funcionalidad:

2.2.1 Funciones de Búsqueda y referencia: Permiten encontrar valores dentro de libros de cálculo escritos en formato de Excel de acuerdo a los criterios establecidos en la búsqueda. También nos ayudan a obtener información de referencia de las celdas.

2.2.2 Funciones Lógicas: Se utilizan en la toma de decisiones, en base al resultado de una función decidiremos si ejecutar o no cierta acción requerida.

2.2.3 Funciones de texto: Nos permite concatenar cadenas de caracteres, remover los espacios en blanco, reemplazar ciertos caracteres por otros y muchas cosas más que facilitarán manipular cadenas de texto para obtener los resultados planteados.

2.2.4 Funciones de fecha y hora: Estas son utilizadas para buscar fechas específicas, para conocer la hora actual, para encontrar la diferencia en los días laborales entre dos fechas y muchas cosas más que serán de gran utilidad al momento de estar trabajando con este tipo de datos.



2.2.5 Funciones de información: Son útiles para obtener información acerca de las hojas de trabajo y su contenido tales como su ubicación dentro de la hoja, su formato, el tipo de valor que contiene una celda o inclusive si la celda tiene el valor de error.

2.2.6 Funciones estadística: Nos permiten realizar un análisis estadístico de tus datos, se puede obtener la covarianza, desviación estándar, distribución binomial, entre otra batería de funciones por lo cual es tan utilizado por los estadísticos.

2.2.7 Funciones matemática y trigonométrica: Son utilizadas para ejecutar operaciones aritméticas como la suma y el producto de dos números. Las funciones trigonométricas de Excel permitirán obtener el seno, coseno, tangente de un ángulo especificado.

2.2.8 Funciones financieras: Las funciones financieras de Excel te ayudaran en cálculos como el de amortización, la tasa de interés anual efectivo, el interés acumulado, la tasa nominal entre otros cálculos que ayudarán en los modelos financieros.

2.2.9 Funciones de ingeniería: La utilidad de estas radica en el cálculo relacionados a los campos de ingeniería como la solución de problemas, de programación de ondas con la función bessel, cálculos como números complejos y conversiones entre diferentes sistemas de numeración como el binario, octal, decimal y hexadecimal.

2.2.10 Funciones de base de datos: Facilitan nuestro trabajo con la información que está organizada como una base de datos, es decir, que se encuentra en un rango de celdas donde la primera fila tiene los títulos de cada columna. Estas funciones nos permiten contar, sumar, multiplicar los valores de una columna que cumplen con los criterios especificados e inclusive podremos extraer un registro que cumpla con ciertas condiciones que apliquemos.

2.2.11 Funciones de cubo: nos ayudan a obtener información de un cubo olap y colocar la información directamente en una hoja de Excel. De esta manera podemos combinar el potencial de las funciones de Excel y su motor de cálculo junto con los beneficios de un repositorio de datos multidimensionales. Con las funciones de cubo podemos buscar datos



de un cubo olap como miembros, conjuntos, propiedades o valores y mezclarlos con otros cálculos y fórmulas de Excel.

2.3 Descripción de algunas funciones de Microsoft Excel¹⁰

2.3.1 Función BUSCAR

Devuelve un valor de un rango de una fila o de una columna o de una matriz. La función BUSCAR tiene dos sintaxis, vectorial y matricial. La forma vectorial de BUSCAR busca un valor en un rango de una fila o de una columna (vector) y devuelve un valor desde la misma posición en un segundo rango de una fila o de una columna. La forma matricial de BUSCAR, busca el valor especificado en la primera fila o en la primera columna de la matriz y devuelve el valor de la misma posición en la última fila o columna de la matriz.

Sintaxis 1: forma vectorial

BUSCAR (*valor buscado; vector de comparación; vector resultado*)

Valor _buscado: es un valor que BUSCAR busca en la matriz.

Valor _buscado puede ser un número, texto, un valor lógico o un nombre o referencia a un valor.

Vector de comparación: es un rango que sólo contiene una columna o una fila. Los valores en el *vector de comparación* pueden ser texto, números o valores lógicos.

Nota: los valores en el *vector de comparación* deberá colocarse en orden ascendente: ...;-2;-1; 0; 1; 2;...; A-Z; FALSO; VERDADERO; de lo contrario BUSCAR puede dar un valor incorrecto. El texto en mayúscula y en minúscula es equivalente.

¹⁰ Página web (MICROSOFT OFFICE)

MICROSOFT OFFICE. (s.f.). Recuperado el 12 de Octubre de 2014, de www.office.microsoft.com



Vector resultado: es un rango que solo contiene una columna o una fila debe ser del mismo tamaño que el *vector de comparación*.

Si BUSCAR no puede encontrar el *valor buscado*, utilizara el mayor valor del *Vector de comparación* que sea menor o igual al *valor buscado*.

Si valor buscado es menor que el valor más bajo del *Vector de comparación* BUSCAR devuelve el valor de error #N/A.

Sintaxis 2: forma matricial

BUSCAR (*valor buscado*; *matriz*)

Valor buscado: es un valor que BUSCAR busca en la *matriz*. *Valor buscado* puede ser un número, texto, un valor lógico o un nombre o referencia a un valor.

Si BUSCAR no puede encontrar el *valor buscado*, utiliza el mayor valor de la matriz que sea menor o igual al *valor buscado*.

Si el *valor _buscado* es menor que el valor más bajo de la primera fila o columna (dependiendo de las dimensiones de la *matriz*), BUSCAR devolverá el valor de error #N/A.

Matriz: es un rango de celdas que contiene el texto, los números o los valores lógicos que desea comparar con *valor buscado*.

La forma matricial de BUSCAR es muy parecida a la de la función BUSCARV busca en la primera columna y BUSCAR busca de acuerdo a las dimensiones de la matriz.

Si la matriz cubre un área que es más ancha que alta (más columnas que filas), BUSCAR buscara valor buscado en la primera fila. Si la matriz es cuadrada o más alta que alta (más filas que columnas), BUSCAR buscara en la primera columna.

2.3.2 Función SUMA

Suma todos los números de un rango.

Sintaxis: SUMA (número1; número2;...)



Número1; número2;... son de 1 a 255 argumentos cuyo valor total o suma desea obtener.

Cuando se suma una matriz la forma es SUMA (A1:A125), esta función sumara todos los valores que se encuentra en este rango.

Observaciones

Se tienen en cuenta los números, valores lógicos y representaciones textuales de números que se escriban directamente en la lista de argumentos.

Si un argumento es una matriz o una referencia, sólo se considerarán los números de esa matriz o referencia. Se pasarán por alto las celdas vacías, los valores lógicos o el texto contenidos en la matriz o en la referencia.

Los argumentos que sean valores de error o texto que no se pueda traducir a números provocan errores.

2.3.3 Función PRODUCT

Multiplica todos los números proporcionados como argumentos.

Sintaxis

PRODUCTO (número1, [número2], ...)

- número1 (*obligatorio*): El primer número a multiplicar.
- número2 (*obligatorio*): El segundo número a multiplicar y hasta un máximo de 255.

2.3.4 Función POTENCIA

Eleva un número a una potencia especificada.

Sintaxis

POTENCIA (número, potencia)

- número (*obligatorio*): El número que se elevará a una potencia.
- potencia (*obligatorio*): El exponente al que se elevará el número

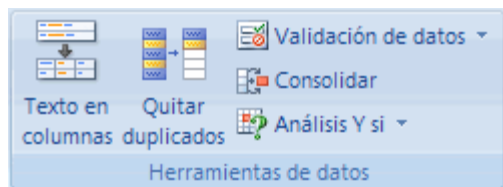


2.3.5 Lista desplegable

Para facilitar la entrada de datos, o para limitar las entradas a determinados elementos, se puede crear una lista desplegable de entradas válidas que se compila a partir de las celdas de otra parte de la hoja de cálculo. Cuando se crea una lista desplegable para una celda, se muestra una flecha en dicha celda. Para escribir información en esta celda, haga clic en la flecha y, a continuación, en la entrada que desee.

Crear una lista desplegable:

1. Seleccione la celda donde desee crear la lista desplegable.
2. En el grupo Herramientas de datos de la ficha Datos, haga clic en Validación de datos.



3. Se muestra el cuadro de diálogo Validación de datos.
4. Haga clic en la ficha Configuración.
5. En el cuadro Permitir, haga clic en Lista.
6. Para especificar la ubicación de la lista de entradas válidas, siga uno de los procedimientos siguientes:
 - Si la lista está en la hoja de cálculo actual, escriba una referencia a la lista en el cuadro Origen.
 - Si la lista está en otra hoja de cálculo, escriba el nombre definido para la lista en el cuadro Origen.



CAPÍTULO III: ELEMENTOS DE VISUAL BASIC

3.1 Definiciones

3.1.1 Visual Basic: Microsoft Visual Basic es un lenguaje de programación desarrollado por el alemán Alan Cooper, su primera versión fue presentada en 1991, con la intención de simplificar la programación utilizando un ambiente de desarrollo completamente gráfico que facilitará la creación de interfaces gráficas y en cierta medida, también la programación de la misma.

Visual Basic consiste en un editor de código, un depurador, un compilador y un constructor de interfaz gráfica.

La razón por la que visual Basic es una herramienta poderosa para Excel, se debe a que en Excel en muchas ocasiones para evitarnos estar en un constante ciclo repetitivo del tratamiento de los datos, necesitamos acelerar este proceso, o en otro aspecto cuando necesitamos realizar alguna tarea que no esté incluida en la interfaz.

Visual Basic para aplicaciones funciona mediante el diseño y ejecución de macros, estos mismo son los que nos agilizan, facilitan, y mejoran la experiencia en el tratamiento de datos que ofrece Excel.

3.1.2 Macros: Es un conjunto de instrucciones que sirven para automatizar los distintos procesos que se pudieran utilizar en Excel, es decir automatizar tareas individuales o en conjunto, con el simple accionar de un botón.

Algunos ejemplos de las diferentes tareas sencillas que se podrían realizar son, darle formato a los textos del documento, generar distintos formatos para el tratamiento de las celdas, generar fórmulas pre escritas personales, llamar celdas, escribir o ingresar textos en distintas celdas, presentar gráficos, esconder columnas, etc...

Excel presenta dos formas para generar una macros, el método de grabación, y el diseño del código de la macros.

El método de grabación, es el más intuitivo dado que simplemente debes acceder al botón grabar macros, y luego empezar a realizar las tareas que futuramente realizaras en repetidas ocasiones. Visual Basic para aplicaciones lo que realiza al grabar una macros es GRABAR todas las acciones deseadas, movimientos del mouse, escritura, diseño de fórmulas, etc...

Es decir, que al estar grabando, lo que realmente hace es generar en segundo plano los códigos escritos al realizar nuestras acciones, gracias a esto evitamos la parte confusa de diseñar propiamente el código.



El segundo método que es el diseño del código de la macros, es simplemente escribir en forma de código todo lo que queremos que la macros realice, este método permite agregar más funciones que el anterior, pero es un poco más complicado, por la amplia cantidad de funciones que se deben de tomar en cuenta para poder diseñar y hacer funcionar la macros con todas las ordenes que se le asignen.

Entonces una de las ventajas de utilizar macros es que podemos realizar programas pequeños a la medida de nuestras necesidades. Además al reducir las horas de tratamiento, formato que se perdían en el trabajo de nuestro Excel, ya no se perderán, lo cual aumentará teóricamente la eficiencia y eficacia de nuestros trabajos.

3.1.3 Proyecto: Es una colección de archivos relacionados o programas, que se integran para crear en su conjunto una aplicación completa. El proyecto genera el programa final.

3.1.4 Objeto: Un objeto es una entidad que tiene asociado un conjunto de métodos, eventos y propiedades.

Un objeto es una combinación de código y datos que puede tratarse como una unidad. Un objeto puede ser una porción de una aplicación, como un control o un formulario. Una aplicación entera también puede ser un objeto.

Las propiedades se pueden considerar como atributos de un objeto, los métodos como sus acciones y los eventos como sus respuestas.

3.1.4.1 Propiedades: Son los atributos de un objeto que definen las características del objeto, como tamaño, color o ubicación en pantalla, o un aspecto de su comportamiento, por ejemplo, si está habilitado o visible. Para cambiar las características de un objeto, se cambian los valores de las propiedades correspondientes.

3.1.4.2 Método: Los métodos son funciones internas de un determinado objeto que permite realizar funciones sobre él o sobre otro objeto. Se le conoce como aquello que el objeto puede hacer, es decir, es una acción que un objeto puede realizar.

3.1.4.3 Evento: Un evento es una acción que un objeto reconoce, se dice también que es un proceso que ocurre en un momento no determinado causando una respuesta por parte de un objeto. Los eventos se producen como consecuencia de una acción del usuario o de código del programa, o bien pueden ser desencadenados por el sistema.

3.1.5 Módulo: Un módulo es un archivo de Visual Basic donde escribimos la parte del código de nuestro programa, es decir, es un conjunto de declaraciones y procedimientos de Visual Basic que se almacenan en una sola unidad.

3.1.6 Procedimientos: Un procedimiento está formado por un conjunto de sentencias que permite resolver un problema. Un módulo está formado por uno o más procedimientos. Un



procedimiento se declara a través de la sentencia **Sub** y puede ser **Privado**, **Público** o **Estático**.

3.1.6.1 Procedimiento Privado: Un procedimiento privado sólo es accesible por otros procedimientos dentro del mismo módulo. Su sintaxis es:

Private Sub Procedimiento (Argumento1, Argumento2,.....)

[Sentencias]

End Sub

3.1.6.2 Procedimiento Público: Un procedimiento público es accesible por todos los procedimientos de todos los módulos de VBA de un proyecto, su sintaxis es:

Public Sub Procedimiento (Argumento1, Argumento2,.....)

[Sentencias]

End Sub

3.1.6.3 Procedimiento Estático: Para que las variables de un procedimiento se conserven una vez terminada su ejecución, éste debe definirse como Estático:

Static Sub Procedimiento (Argumento1, Argumento2,.....)

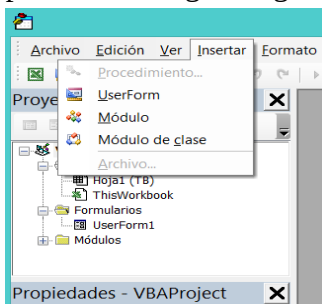
[Sentencias]

End Sub

La sentencia **Sub** y **End Sub** son obligatorias al definir cualquier procedimiento. Los argumentos y las sentencias Private, Public y Static son opcionales. Es importante mencionar que al definir un procedimiento sin ninguna de las sentencias anteriores, por defecto éste se define como Público.

3.2 Algunas Funciones de Visual Basic para Aplicaciones

Visual Basic para empezar presenta tres tipos de módulos: estándar, de clase y userform, podemos elegir según sea nuestro interés para empezar a generar una macro.





3.2.1 Módulo estándar: Un módulo que contiene solamente declaraciones y definiciones de procedimiento, tipo y datos. Las declaraciones y definiciones a nivel de módulo de un módulo estándar son Public de manera predeterminada. Un módulo estándar se denomina módulo de código en versiones anteriores de Visual Basic.

3.2.2 Módulo de clase: En estos módulos se pueden crear clases personalizadas con las cuales se puede representar en cierta forma objetos, aunque Excel VBA no es un lenguaje orientado a objetos, se puede aprovechar algunas ventajas que da el uso de clases.

En VBA una clase es algo tan sencillo como un procedimiento, que contiene datos para manejarlos, y con el que se puede crear objetos.

Excel cuenta con muchos objetos disponibles, aunque puede llegar el momento en que sea más práctico para el usuario crear su propio objeto. Estos se crean usando un módulo de clase asignándoles propiedades y métodos personalizados.

3.2.3 UserForm: Un objeto userform es una ventana o cuadro de dialogo que conforma una parte del interfaz de una aplicación.

La colección userforms es una colección cuyos elementos representan cada userform cargado en la aplicación. La userforms tiene una propiedad count, una propiedad ítem y un método add. Count especifica el número de elementos en la colección; Item (el miembro predeterminado) especifica un miembro de la colección específico y add coloca un nuevo elemento userform en la colección.

Sintaxis

UserForms [.Item] (índice)

El marcador de posición índice representa un entero entre 0 y UserForms.Count – 1. Item es el miembro predeterminado de la colección UserForms y no se necesita especificar.

Puede utilizar la colección UserForms para pasar a través de todos los formularios de usuario cargados en una aplicación. Se identifica con una variable intrínseca global llamada UserForms. Puede transferir UserForms (índice) a una función cuyo argumento está especificado como una clase UserForm.

Los formularios de usuario tienen propiedades que determinan la apariencia, como posición, tamaño y color; y aspectos de su comportamiento.

Los formularios de usuario también pueden responder a eventos iniciados por un usuario o activados por el sistema. Por ejemplo, puede escribir código en el procedimiento de evento



Initialize del UserForm para inicializar variables a nivel de módulo antes de mostrar el UserForm.

Además de propiedades y eventos, puede utilizar métodos para manipular formularios de usuario utilizando código. Por ejemplo, puede utilizar el método Move para cambiar la ubicación y tamaño de un UserForm.

Cuando diseña formularios de usuario, establezca la propiedad BorderStyle para definir bordes, y establezca la propiedad Caption para colocar texto en la barra de título. En código, puede utilizar los métodos Hide y Show para hacer invisible o visible un UserForm en tiempo de ejecución.

UserForm es un tipo de datos Object. Puede declarar variables como de tipo UserForm antes de colocarlas en una instancia de un tipo de UserForm declarado en tiempo de diseño. De modo parecido, puede transferir un argumento a un procedimiento como de tipo UserForm. Puede crear instancias múltiples de formularios de usuario en código utilizando la palabra clave New en instrucciones Dim, Set y Static.

Los formularios disponibles en VBA son conocidos como Userforms, estos formularios pueden ser empleados desde cualquier hoja de Excel, simplemente invocándolos desde un botón.

Cuando se desea ejecutar un macro, al ingresar la opción “Macros” desde la ficha de programador, se muestra una relación de macros que pueden ser empleadas, desde dicha pantalla no existe más interacción relevante que solo seleccionar cuál de las macros emplear, en cambio desde un Userform se pueden crear objetos como cajas de texto, botones de opción, listas, etc., con las cuales se puede interactuar antes de proceder a ejecutar alguna macro.

El contenido disponible en un Userform puede ser empleado para interactuar con algún contenido en cualquiera de las hojas que encuentren en nuestro archivo Excel, o solo interactuar entre los datos disponibles desde el mismo Userform.

Propiedades UserForm

Esta ventana de VBA lo que nos permite es darle formato a nuestra ventana de userform, nos muestra todas las propiedades modificables de la ventana y de los ítems que agreguemos a ella, estas propiedades pueden variar en dependencia de que tengamos seleccionado en la ventana de userform, ajustándose y mostrando las propiedades específicas para ese ítem.



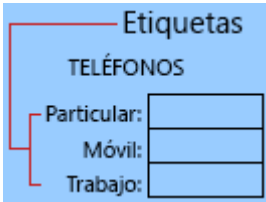
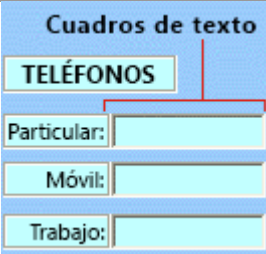
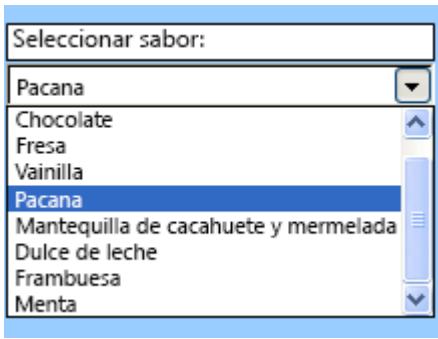
Principales propiedades del userform:

- **Name:** El nombre con el que vamos a identificar el userform que estamos diseñando. Tiene que ser único y no puede tener espacios ni otros caracteres (no alfanuméricos).
- **BackColor:** Establece el color de fondo, para mostrar el texto y los gráficos del Objeto.
- **BorderColor:** Establece el color del borde del cuadro del userform, o cualquier otro ítem dentro de él.
- **BorderStyle:** Establece el estilo del borde del cuadro UserForm. Solo presenta dos opciones.
- **Caption:** Establece el texto mostrado como título.
- **Enable:** Devuelve un valor determinado valor falso o verdadero
- **Font:** Establece el tamaño y el tipo de letra.
- **Picture:** Establece una imagen dentro del cuadro userform
- **StartPosition:** Designa la posición en la que aparecerá el cuadro de diálogo
- **SpecialEffect:** Para asignar diferentes diseños predeterminados para cada opción en la que se puede utilizar esta propiedad.
- **PictureAlignment:** Alinea la imagen que se está insertando en el userform.
- **Text:** Es el texto, o caracteres que queremos que aparezcan dentro del ítem seleccionado.
- **Multiline:** los valores posibles son verdaderos y falsos, indicando si queremos que el control permita al usuario escribir más de una línea o no.
- **Cancel:** Verdadero o falso, esto nos permite en el botón de comando asignar si queremos que la ventana que nos abra este botón se cierre presionando el botón escape o que no suceda esto.
- **ControlSource:** Identifica la ubicación de los datos, esta acepta intervalos de hojas de cálculo de Microsoft Excel.
- Es decir identifica una celda o un rango, no contiene los datos almacenados en la celda o rango. Si puede cambiar la propiedad value del control, el cambio se refleja automáticamente en la celda o el campo vinculado. De modo similar si el valor de la celda cambia manualmente o por algún evento, en el userform entonces también cambiará el valor que presenta.
- **Rowsource:** especifica el origen que proporciona una lista para el control ComboBox o ListBox.
- **ListRows:** especifica el número máximo de las filas que se muestran en la lista.
- Esto no quiere decir que no mostrará toda la lista, si el número de elementos dentro de la lista es superior al establecido, mostrará una barra de desplazamiento para poder mostrar el resto de elementos.

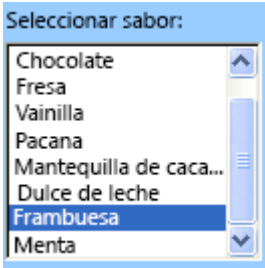
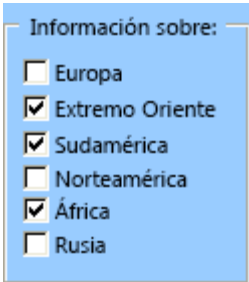
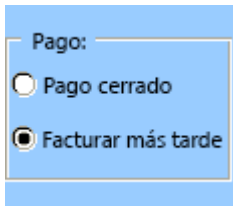
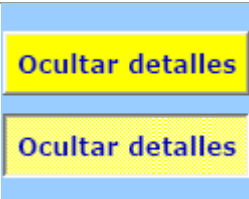


- **Autosize:** esta propiedad muestra dos valores falso, y verdadero, donde falso anula el autosize del objeto que tenemos seleccionado y verdadero lo activa, permitiendo que se ajuste automáticamente a lo que pretendemos mostrar en este objeto.

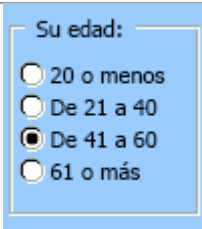
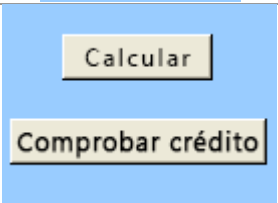
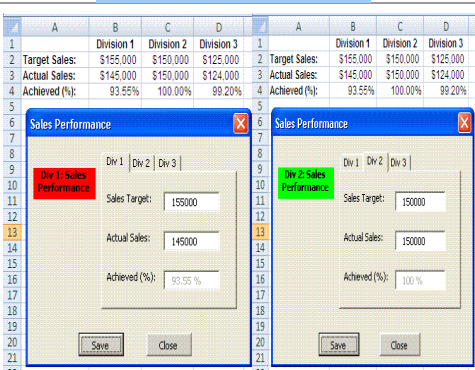
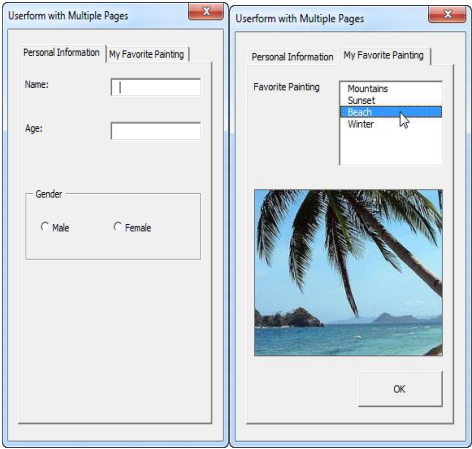
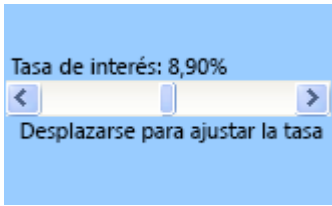
Además del conjunto de propiedades, los formularios en VBA constan con un cuadro de herramientas que es donde se encuentran todos los controles que pueden añadirse para la funcionalidad y personalización de la ventana del formulario. Los principales controles que se encuentran son los siguientes:

Nombre del Control	Ejemplo	Descripción
 Etiqueta		Describe la finalidad de una celda o un cuadro de texto, o muestra texto descriptivo (como títulos, leyendas, imágenes) o breves instrucciones.
 Cuadro de texto		Permite ver, escribir o editar texto o datos encadenados a una celda, en un cuadro rectangular. Un cuadro de texto también puede ser un campo de texto estático que presenta información de solo lectura.
 Cuadro combinado		Combina un cuadro de texto con un cuadro de lista para crear un cuadro de lista desplegable. Un cuadro combinado es más compacto que un cuadro de lista pero requiere que el usuario haga clic en la flecha de abajo para mostrar una lista de elementos. Use un cuadro combinado para permitir que un usuario escriba una entrada o elija solamente un elemento de la lista. El control muestra el valor actual en el cuadro de texto, sin importar el modo en que dicho valor se haya proporcionado.

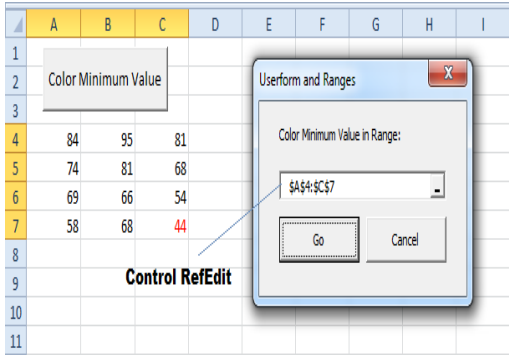


 Cuadro de lista		Presenta una lista de elementos donde el usuario puede seleccionar uno (o múltiples elementos). Estos controles son muy flexibles. Por ejemplo, se puede especificar un rango de hoja de cálculo que contenga elementos de un cuadro de lista, y este rango puede constar de múltiples columnas.
 Casilla de verificación		Activa o desactiva un valor que representa una selección inequívoca entre opuestos, es decir, cuando se selecciona una casilla de verificación posee un valor verdadero, en caso contrario es falso. Puede seleccionar más de una casilla de verificación en una hoja de cálculo o en un cuadro de grupo. Una casilla de verificación puede tener uno de tres estados: activada, desactivada y mixta, lo que significa una combinación de los estados activada y desactivada (como en una selección múltiple).
 Botón de opción		Permite una única elección dentro de un conjunto limitado de opciones que se excluyen mutuamente; un botón de opción generalmente está contenido en un cuadro de grupo o un marco. Un botón de opción puede tener uno de tres estados: activado, desactivado y mixto, este último significa la combinación de los dos primeros.
 Botón de alternar		Indica un estado, como Sí/No, o un modo, como Activado/Desactivado. El botón alterna entre un estado habilitado o deshabilitado cuando se hace clic en él.



 Cuadro de grupo o marco		Agrupar controles relacionados en una unidad visual en un rectángulo con una etiqueta opcional. Generalmente, se agrupan botones de opción, casillas de verificación o contenido estrechamente relacionado.
 Botón		Ejecuta una macro que realiza una acción cuando un usuario hace clic en él. Los botones también se conocen como botones de comando.
 Barra de tabulacionnes		Es una colección de pestañas en la que cada pestaña contiene el mismo conjunto de controles, y el contenido de los controles cambia cuando se selecciona una pestaña diferente. En otras palabras este control permite visualizar diferentes contenidos en cada ficha, para el mismo conjunto de controles.
 Página múltiple		Este control comprende de uno o más objetos de página, que contienen cada una su propio conjunto de controles. Todos los controles que se agregan a una página, están contenidos en ella y se convierten en una parte de esa página, es decir, que cada página puede contener un conjunto de controles diferentes.
 Barra de desplazamiento		Se desplaza por un intervalo de valores cuando el usuario hace clic en las flechas de desplazamiento o arrastra el cuadro de desplazamiento. Además, se puede mover por una página (en un intervalo preestablecido) de valores haciendo



		clic en el área entre el cuadro de desplazamiento y cualquiera de las flechas de desplazamiento. Generalmente, el usuario también puede escribir un valor de texto directamente en un cuadro de texto o una celda asociado.
 Control de número		Aumenta o disminuye un valor, como un incremento numérico, una hora o una fecha. Para incrementar el valor, es necesario hacer clic en la flecha arriba; para disminuirlo, se debe hacer clic en la flecha abajo. Generalmente, el usuario también puede escribir un valor de texto directamente en un cuadro de texto o una celda asociado.
 Imagen		Se utiliza para mostrar una imagen en el formulario de usuario.
 RefEdit		Permite al usuario seleccionar un rango de hoja de cálculo de una caja, o para introducir el rango en el mismo. Se comporta de manera similar a las cajas de referencia de Excel incorporadas. El uso más común de este control es para seleccionar una celda o un rango de celdas para dar formato a ellos u obtener sus valores.

Los eventos son acciones que se pueden realizar en cualquier control: click, doble click, movimiento del ratón. A estos eventos se les puede asociar código para que se ejecute al producir el evento. Los eventos ya están definidos, son bastantes y cada control cuenta con los suyos propios, aunque son muy parecidos. Lo único que tendremos que hacer es asociar el código necesario al evento que necesitemos tratar.

Entre los principales eventos que podemos mencionar se encuentran:

- **Click:** ocurre cuando se presiona y suelta un botón del mouse sobre un objeto.



- **DblClick:** ocurre cuando se presiona y suelta dos veces un botón del mouse sobre un objeto.
- **DragDrop:** ocurre al arrastrar y soltar un determinado objeto con el mouse.
- **DragOver:** ocurre si una operación de arrastrar y soltar está en curso.
- **GotFocus:** ocurre cuando un objeto recibe el control o foco, ya sea mediante una acción del usuario como hacer click en un objeto ventana, o cambiando el foco de objeto desde el programa, mediante el método SetFocus.
- **LostFocus:** contrario al anterior, este evento ocurre cuando el objeto pierde el enfoque, sea mediante acción del usuario o efectuado desde la aplicación.
- **KeyDown:** ocurre cuando el usuario mantiene presionada una tecla.
- **KeyUp:** ocurre cuando el usuario deja de presionar una tecla. Este evento sucede precisamente al terminar el evento KeyDown.
- **KeyPress:** ocurre como cuando se presiona y suelta una tecla. En este evento el único argumento KeyAscii es necesario para conocer cuál es el código ASCII de la tecla pulsada
- **MouseDown:** ocurre cuando el usuario presiona un botón del mouse.
- **MouseUp:** se produce cuando el usuario suelta el botón del mouse.
- **MouseMove:** este evento ocurre mientras el usuario mueve o desplaza el puntero del mouse sobre un objeto.
- **Change:** al cambiar el contenido del control.
- **BeforeUpdate:** tiene lugar antes de que se actualicen los datos modificados en un control o registro.
- **AfterUpdate:** Ocurre tras modificar los datos de un control.
- **Initialize:** Para inicializar las variables definidas a nivel de módulo.



IV. DISEÑO METODOLÓGICO

Tipo de estudio: El tipo de estudio según el nivel de medición de los datos obtenidos es de carácter cuantitativo y descriptivo.

Descriptivo: Ya que en el presente trabajo abarcamos de forma descriptiva todo lo relacionado al programa visual Basic, sus funciones con sus diversas características, así como también las rentas vitalicias, su concepto, clasificaciones y características.

Cuantitativa: Debido a que utilizaremos técnicas matemáticas científicas para los cálculos de las rentas, las cuales serán descritas paso a paso.

Obtención de la Información: Recolectamos información de las fuentes primarias como son los libros de cálculo actuarial y matemática financiera para basar nuestros cálculos en los procesos ahí descritos. En las fuentes secundarias fueron el material obtenido de internet, como medio de apoyo al momento de trabajar con el programa de visual Basic y las funciones de Excel 2010.

Procesamiento de la información: Para el desarrollo de nuestra investigación utilizamos el Software MS Excel 2010, así como también la aplicación Visual Basic dentro del mismo para el avance de nuestro trabajo.

Tipos de variables:

1. Variables independientes

- ❖ **Renta (R):** Es el valor de cada pago periódico.
- ❖ **Tiempo o plazo de una anualidad (n):** El intervalo de tiempo que transcurre entre el comienzo del primer período de pago y el final del último.
- ❖ **Edad de emisión (x):** es la edad a la que se emite el contrato.
- ❖ **Diferimiento (r):** es el periodo de gracia, es decir, el tiempo en el que no se paga la renta.
- ❖ **frecuencia o periodo fraccionario (m):** es el plazo menor a un año, por ejemplo: meses, trimestres, bimestres etc.
- ❖ **Razón de Crecimiento (h):** serie de pagos donde sus valores de cada período aumentan en una cantidad constante.

2. Variable Dependiente

- ❖ **Prima Neta Única o VAA (valor actual actuarial):** es el importe que se debe pagar en una sola exhibición.



V. RESULTADOS

5.1 Diseño del programa en Excel.

5.1.1 Generando las tablas de mortalidad.

Utilizaremos las probabilidades de muerte dadas por las tablas CSO41, CSO58, GAM-71, CSO80 y CET58, y en base a esto crearemos nuestra tabla de mortalidad dinámica, siguiendo el protocolo que mostrará los valores de x , l_x , d_x , q_x , C_x , D_x , M_x , N_x , R_x , S_x de los cuales utilizaremos x , D_x , N_x y S_x para los diferentes cálculos que realizaremos para nuestro programa.

TABLA DE MORTALIDAD DINAMICA										
		Tabla		CSO 58						
		i		0.025						
		v		0.98						
x	l_x	d_x	q_x	C_x	D_x	M_x	N_x	R_x	S_x	x
0	10000,000.00	70,800.00	0.00708	69,073.17	10000,000.00	2076,827.70	324850,064.25	118975,370.96	8440862,424.93	0
1	9929,200.00	17,475.39	0.00176	16,633.33	9687,024.39	2007,754.53	314850,064.25	116898,543.26	8116012,360.68	1

Lo primero que haremos es diseñar la tabla con sus respectivas variables además de establecer las celdas en las que se insertarán el interés técnico (i), la lista desplegable de que probabilidades de muerte se usarán y la celda en que se insertará la fórmula del factor de actualización actuarial.

Fórmula de actualización actuarial			
=1/(1+\$AG\$5)			
T	U	V	W
		i	0.025
		v	=1/(1+\$AG\$5)

Este factor de actualización lo aplicaremos al interés técnico, este mismo después será utilizado para generar distintos conmutativos de nuestra tabla dinámica. Uno entre (uno más el interés).

Calculamos las defunciones multiplicando las probabilidades de muerte por nuestra l_x inicial, que es nuestra población y así sucesivamente para cada una de las edades.



Asignamos q_x (probabilidad de muerte) utilizando la función Si, condicionando a que si la tabla elegida en la lista desplegable coincide con alguna de las disponibles, muestre el valor correspondiente empezando por la probabilidad de muerte a la edad cero, hasta llegar a 99.

SUMA X ✓ f _x =SI(\$AG\$4=\$AP\$3,AP7,SI(\$AG\$4=\$AQ\$3,AQ7,SI(\$AG\$4=\$AR\$3,AR7,SI(\$AG\$4=\$AS\$3,AS7,SI(\$AG\$4=\$AT\$3,AT7))))											
	Q	R	S	T	U	V	W	X	Y	Z	AA
6	x	l_x	d_x	q_x	C_x	D_x	M_x	N_x	R_x	S_x	x
7	0	10000,000.00	70,800.00	0.00708	69,073.17	10000,000.00	2076,827.70	324850,064.25	118975,370.96	8440862,424.93	0
8	1	9929,200.00	17,475.39	=SI(\$AG\$4=\$AP\$3,AP7,SI(\$AG\$4=\$AQ\$3,AQ7,SI(\$AG\$4=\$AR\$3,AR7,SI(\$AG\$4=\$AS\$3,AS7,SI(\$AG\$4=\$AT\$3,AT7))))							
9	2	9911,724.61	15,065.82								

Nuestro primer conmutativo C_x es la multiplicación de d_x por el factor de actualización elevado a la $x+1$.

El segundo conmutativo es D_x , en este conmutativo se multiplican los l_x por el factor de actualización elevado a la x , mientras que el resto de conmutativos son sumatorias simples.

M_x es calculado como la sumatoria de los C_x .

N_x es la sumatoria de los valores del conmutativo D_x .

R_x será la suma entonces de los M_x calculados anteriormente.

Y por último nuestro S_x sería la sumatoria de los N_x .

5.1.2 Elaboración del formulario base.

Construida nuestra tabla de mortalidad procedemos a construir una tabla en donde insertaremos las distintas variables involucradas en el cálculo de Rentas vitalicias, en este caso de la prima neta única, para rentas ilimitadas, temporales, y rentas fraccionarias ilimitadas y temporales así como las rentas contingentes variables en progresión aritmética.



COTIZADOR DE PRIMA NETA UNICA	
Tabla de Mortalidad	CSO 41
Interes Técnico (i)	1.00%
Plan	R. Ilimitada Vencida Inmediata
Renta	
Edad de Emisión (x)	
Tiempo(n)	
Diferimiento(r)	
Periodo Fraccionario (m)	
Razon de crecimiento (h)	
Prima Neta Unica	-

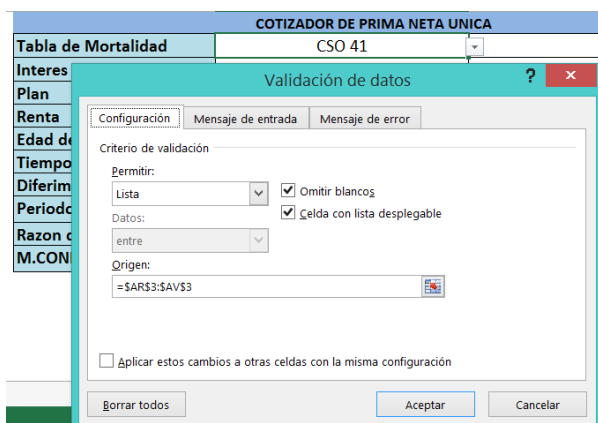
En esta tabla podemos observar 9 celdas, Tabla de mortalidad, Interes Técnico, Plan de renta, Renta, Edad de emisión, Tiempo, Diferimiento, Periodo fraccionario y razón de crecimiento, presentamos este ordenamiento con el fin de facilitarnos el ingreso de los datos que utilizaremos para los cálculos y manejar fluidamente la información. También agregamos una décima celda que es en la cual mostraremos el valor de la prima neta única.

Las celdas contigua al Interés Técnico, Renta, Edad de emisión, Tiempo, Diferimiento, Periodo fraccionario, son celdas simples en las cual es necesario el ingreso de algun valor numérico que permita realizar los cálculos necesarios para encontrar el valor de la Prima neta única.

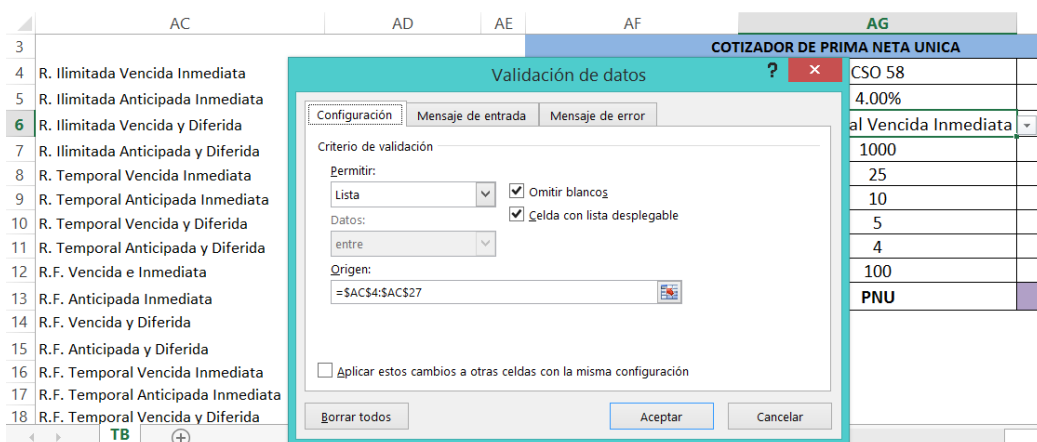
R. Ilimitada Vencida inmediata	10.931,89
R. Ilimitada anticipada inmediata	11.411,89
R. Ilimitada Vencida y Diferida	8.719,70
R. Ilimitada anticipada y diferida	9.137,95
R. Temporal vencida inmediata	7.184,57
R. Temporal anticipada inmediata	7.404,22
R. Temporal vencida y diferida	6.130,48
R. Temporal anticipada y diferida	6.336,10
R.F. Vencida e inmediata	44.447,58
R.F. anticipada Inmediata	44.927,58
R.F. Vencida y Diferida	35.506,17
R.F. anticipada y diferida	35.924,43
R.F. Temporal vencida Inmediata	29.067,76
R.F. Temporal anticipada inmedia	29.287,41
R.F. Temporal vencida y diferida	24.830,36
R.F. Temporal anticipada y diferid	25.035,97



Para facilitarnos el cálculo de las primas con respecto a los diferentes planes, decidimos también ingresar en diferentes celdas cada uno de los planes que se presentarán en nuestro trabajo, y realizar los cálculos individualmente, tomando en cuenta el diseño anterior de la presentación de las distintas variables. Esto también nos permite asignar rápidamente una lista desplegable de los diferentes planes.



En la celda contigua a “tabla de mortalidad” se ingresó una lista desplegable haciendo uso de la validación de datos, con origen de datos en el rango AR3:AV3, en el cual se muestran CSO41, CSO58, GAM-71, CSO80 y CET58.



Esta es la segunda lista desplegable que utilizaremos en la cual se mostrarán los distintos planes de rentas contingentes antes mencionados, el rango de donde tomaremos los datos es AC4:AC27.



=AG7*(BUSCAR((AG8+1),Q7:Q106,X7:X106))/(BUSCAR(AG8,Q7:Q106,V7:V106))	
AC	AD
BUSCAR(valor_buscado, vector_de_comparación, [vector_resu BUSCAR(valor_buscado, matriz)	
COTIZADOR DE PRIMA NETA UNICA	
R. Ilimitada Vencida inmediata	=AG7*(BUSCA
R. Ilimitada anticipada inmediata	11,411.89
R. Ilimitada Vencida y Diferida	8,719.70
R. Ilimitada anticipada y diferida	9,137.95
R. Temporal vencida inmediata	7,184.57
Tabla de Mortalidad	CSO 58
Interes Técnico (i)	2.50%
Plan	R. Temporal vencida y diferida
Renta	480
Edad de Emisión (x)	35

Como se puede observar en esta tabla además de el marco diseñado inicialmente, se establecieron los planes de renta en celdas anteriores, la finalidad de esto es poder calcular las rentas una a una en fórmulas independientes, dado la longitud de las fórmulas de cálculo, y la complejidad de ingresar estas fórmulas en una sola celda, decidimos que esta forma permitiría mayor orden, y flujo de la información, así también en caso de que algún error se presentara, poder determinarlo y darle una solución rápida.

=AG7*(BUSCAR(AG8,Q7:Q106,X7:X106))/(BUSCAR(AG8,Q7:Q106,V7:V106))	
AC	AD
COTIZADOR DE PRIMA NETA UNICA	
R. Ilimitada Vencida inmediata	10,931.89
R. Ilimitada anticipada inmediata	X106))/(BUSCA
R. Ilimitada Vencida y Diferida	8,719.70
R. Ilimitada anticipada y diferida	9,137.95
R. Temporal vencida inmediata	7,184.57
Tabla de Mortalidad	CSO 58
Interes Técnico (i)	2.50%
Plan	R. Temporal vencida y diferida
Renta	480
Edad de Emisión (x)	35

La fórmula que se utilizó para calcular esta prima, en este caso, AG7 es el valor de la renta el cual multiplicará el resultado de la división siguiente, Buscar AG8 (35 en la columna de edad(x) y que muestre el valor de la fila 35 de la columna $N_{(x)}$ dato que esta dividido por el valor que encuentre Buscar AG8 en la columna de edad(x) y que muestre el valor de la fila 35 de la columna $D_{(x)}$,

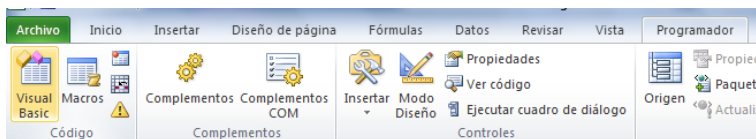
De esta manera también se ingresaron el resto de fórmulas para los distintos planes de rentas, con la idea de automatizar nuestras tablas, el formulario de ingreso y a la vez nuestro programa final.

=SI(AG6=AC4,AD4,SI(AG6=AC5,AD5,SI(AG6=AC6,AD6,SI(AG6=AC7,AD7,SI(AG6=AC8,AD8,SI(AG6=AC9,AD9,SI(AG6=AC10,AD10,SI(AG6=AC11,AD11,SI(AG6=AC12,AD12,SI(AG6=AC13,AD13,SI(AG6=AC14,AD14,SI(AG6=AC15,AD15,SI(AG6=AC16,AD16,SI(AG6=AC17,AD17,SI(AG6=AC18,AD18,SI(AG6=AC19,AD19,SI(AG6=AC20,AD20,SI(AG6=AC21,AD21,SI(AG6=AC22,AD22,SI(AG6=AC23,AD23,SI(AG6=AC24,AD24,SI(AG6=AC25,AD25,SI(AG6=AC26,AD26,SI(AG6=AC27,AD27,"-"))))))))))))))))))))									
AF	AG	AH	AI	AJ	AK	AL	AM	AN	
lo Fraccionario (m)	4			9%					

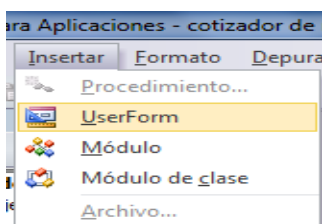
Lo que realizamos para asignar el valor de la prima neta única a nuestra celda fue asignar la fórmula si, en la cual señalamos que si el contenido de la celda AG6 (Plan de renta) es igual a alguna de las celdas en el rango AC4:AC27, asigne el valor correspondiente, es decir, por ejemplo que si AG6 es igual a AC4 se asigne el valor de AD4, celda en la cual está calculado el valor de la prima neta única según su fórmula y valores dados.

5.1.3 Diseño y estructura del programa (macros) para el cálculo de rentas vitalicias.

Una vez que tenemos hecho nuestro formulario base en Excel, para aplicarle un efecto visual y una apariencia más profesional pero intuitiva a nuestro programa, decidimos hacer uso de los formularios en VBA y de esta manera aplicar la programación de macros que puede ser muy útil para el ámbito profesional de un actuario.

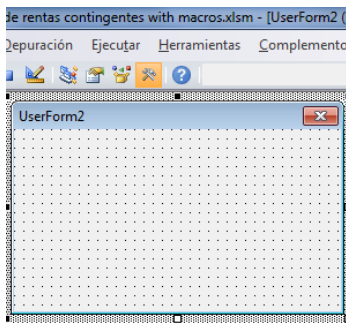


Para empezar debemos entrar en el panel del programador o desarrollador de Microsoft Excel, este panel muestra todas las herramientas que utilizaremos para el diseño del programa, entre ellas el acceso a un desarrollador llamado Visual Basic para aplicaciones, las listas de macros guardadas, complementos, insertar botones, diseño, propiedades, estructura del código utilizado, etc...

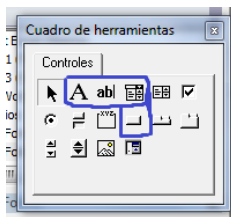




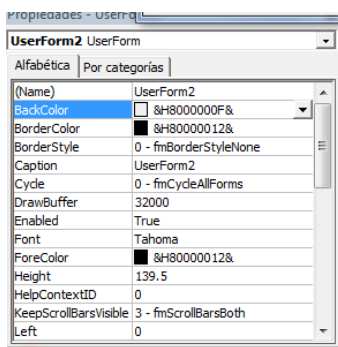
Una vez dentro del Visual Basic para aplicaciones en la barra de herramientas abriremos el desplegable insertar y elegiremos un UserForm, con esto podremos diseñar libremente nuestro programa en forma de formulario.



Una vez abierto el userform, nos aparecerá una ventana como si abrieramos un programa o aplicación, listo para que nosotros diseñemos la estructura del programa o en otras palabras como vamos a introducir la información y a obtenerla.

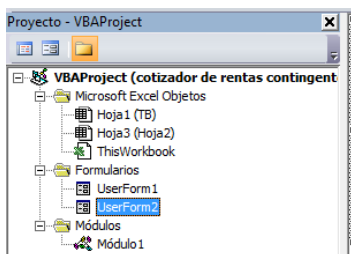


De este cuadro de herramientas, utilizaremos las seleccionadas, estas son de izquierda a derecha, etiqueta, cuadro de texto, combobox (lista desplegable), y el último en la parte inferior es el botón de comando.

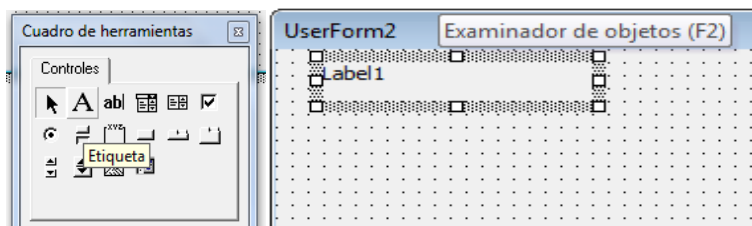




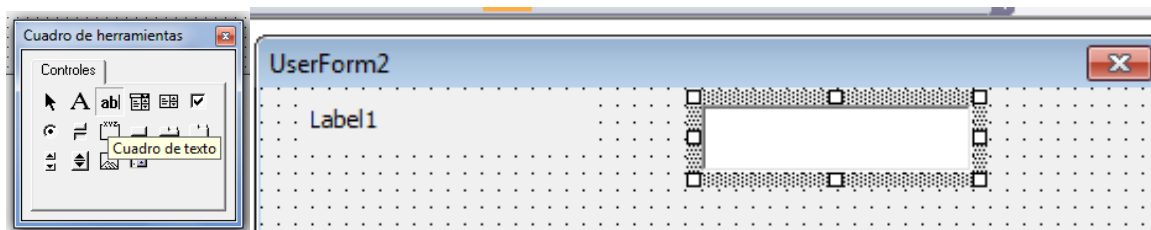
Este software de diseño de programación nos facilita también una ventana de propiedades para el userform, así como para cada uno de los elementos que insertemos en nuestro userform, propiedades como el tamaño de letra, tipo de letra, color de fuente, referencias de celda para lectura o escritura, aplicación de funciones en caso de suceder algún evento entre otros.



Dentro de la ventana proyecto podemos observar que los ítems (objetos) del libro de Excel, los userform (formularios) que tenemos creados en nuestro proyecto y también muestra la existencia de módulos que son ordenes de códigos para ejecutar distintas actividades.

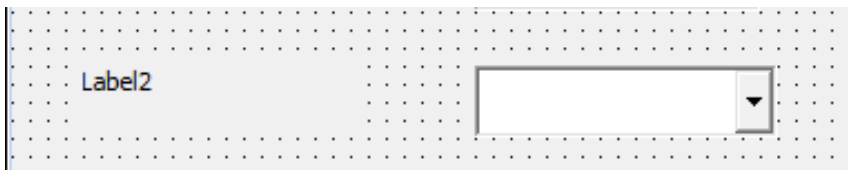


Para crear nuestro programa utilizaremos 9 etiquetas como la que mostramos en la imagen anterior, las cuales nombraremos según el formato que ya tenemos diseñado en la tabla de Excel.

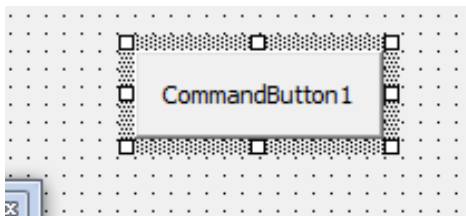




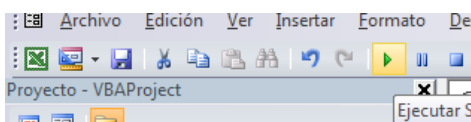
Utilizaremos también 7 cuadros de textos como el que mostramos arriba, en estos cuadros de texto se asignaran las celdas en las cuales el programa (macros) debe de escribir los valores que introduzcamos.



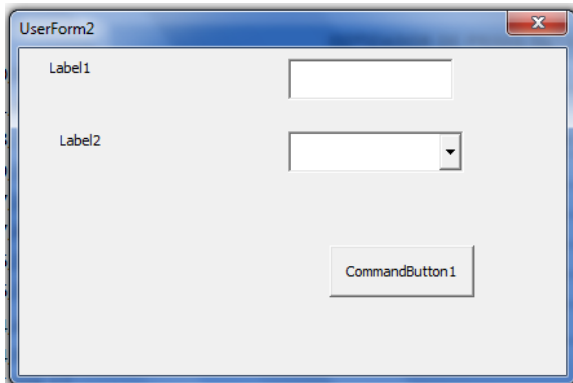
Además de etiquetas y cuadros de texto también utilizaremos 3 listas desplegables, en una de ellas asignaremos las distintas tablas de mortalidad disponibles, en otra utilizaremos los distintos planes de rentas contingentes y una última donde aparecerán los intereses técnicos disponibles.



Por ultimo dentro de nuestro formulario utilizaremos 2 botón de comando, este tipo de botón llama a realizar una función, abrir programas, direccionar a algún sitio, entre muchas otras funciones, en nuestro caso, uno llamará a nuestra celda de Prima neta única y nos mostrará el valor resultado del cálculo del plan de renta contingente seleccionado y en función de todas las variables utilizadas, así como de la tabla de mortalidad asignada. El segundo botón tendrá la función de limpiar los datos, esto con el fin de darle mayor funcionalidad a nuestro programa.

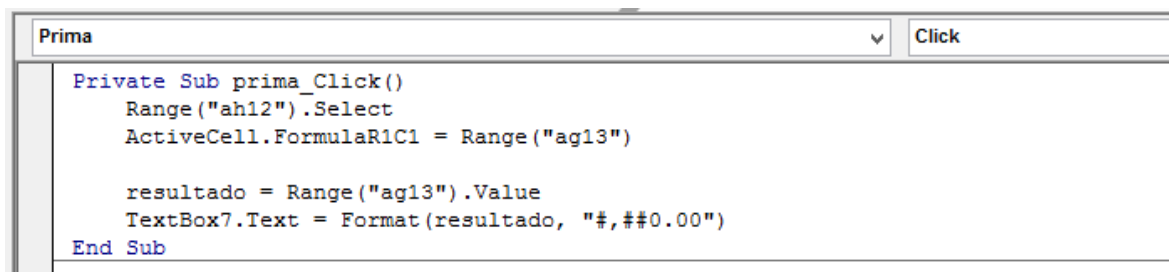


Vamos a correr nuestra macro, para ver como va ir quedando, nos tendrá que aparecer un formulario como el siguiente:



Tenemos entonces que hasta el momento para efectos de mostrar como hacer nuestro formulario, hemos ingresado dos etiquetas, un cuadro de texto y una lista desplegable, además de haber agregado 1 botón de comando. Recordemos que estos procedimientos de ingreso de datos o valores, son similares entre sí, y por lo mismo sólo presentamos a manera de ejemplo como se configuran sus propiedades.

Abriremos nuestro botón de comando que nos dará la prima dándole doble clic, nos aparecerá una ventana similar a la que mostramos, ésta ya viene preparada para que el botón funcione con el “evento Clic” es decir que sólo se active cuando demos un clic al botón, además de esto nos presenta un cuadro de edición, este es específicamente para el ingreso del código de programación que será necesario para que el botón realice alguna función.



En nuestro caso, queremos que el botón llame la celda de prima neta única de nuestra tabla de cálculo en Excel a la celda “ah12” por lo tanto debemos ingresar el siguiente código: Range("ah12").Select, con esto estamos seleccionando la celda “ah12”, el otro código sería ActiveCell.FormulaR1C1 = Range("ag13") , este indica que en la celda activa, es decir, la celda “ah12” que hemos seleccionado nos escriba la fórmula de la celda “ag13” que es donde



se encuentra nuestra prima neta única. Esto con el fin que al darle clic al botón la prima sea mostrada o que copie su valor en una celda diferente, ya que al momento de accionar nuestro botón de limpiar nos borre el valor de esta celda ah12 y no la celda ag13 donde se encuentra nuestra fórmula para el cálculo de la prima. Por otra parte el otro código de este botón servirá para darle formato al resultado y que se nos muestre un separador de mil y en cuatro decimales. El diseño del botón se muestra en la imagen de abajo:



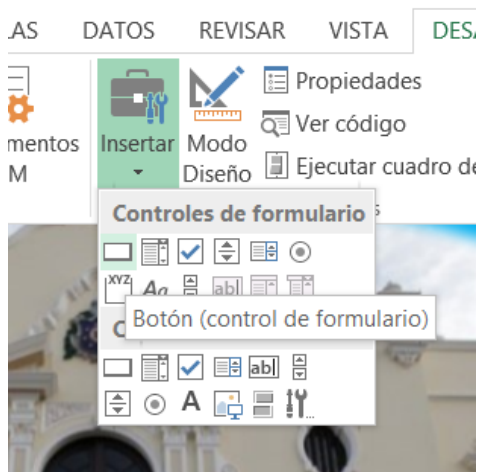
Nuestro segundo botón de comando llamado limpiar tendrá el siguiente código:

```
Private Sub limpiar_Click()  
    Range("ag7:ah12").Select  
    Selection.ClearContents  
  
End Sub
```

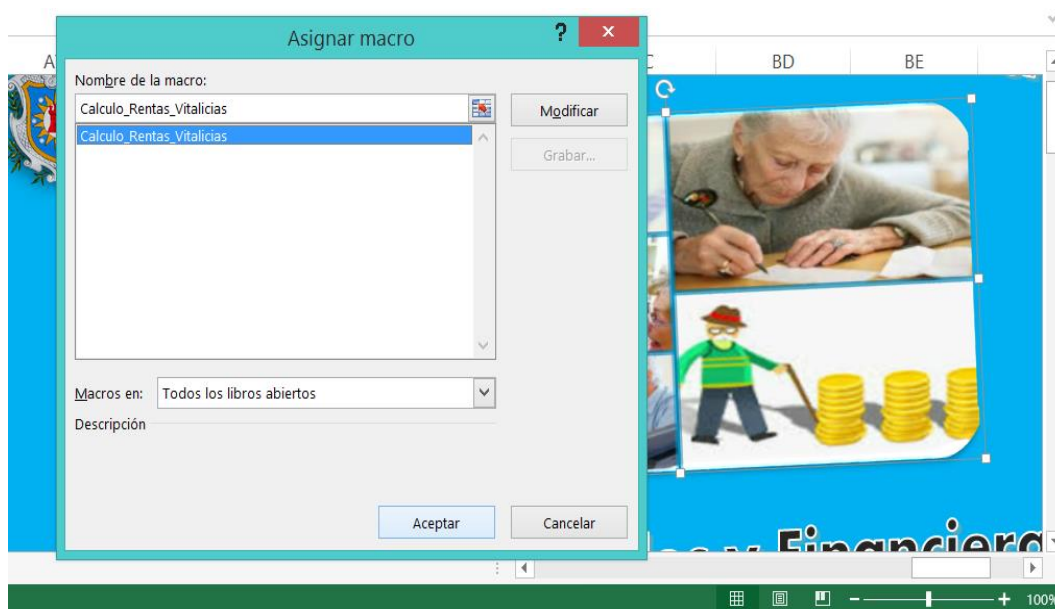
Con este código le estamos diciendo al programa que el RANGO “AG7:AH12” (rango donde se ingresaron los datos de las variables) una vez seleccionado limpie sus valores, todo esto con la finalidad de evitar lo tedioso de eliminar dato por dato y así dar lugar a un nuevo cálculo. El diseño del botón quedaría de la siguiente manera:



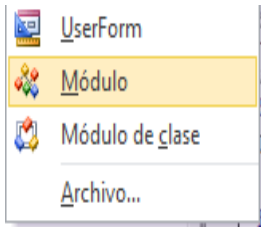
Ahora tenemos que insertar un botón en nuestra hoja de cálculo, con el fin de poder llamar a nuestro userform, se puede realizar en el módulo programador, seleccionamos insertar y escogemos un botón simple y nos mostrará el siguiente recuadro.



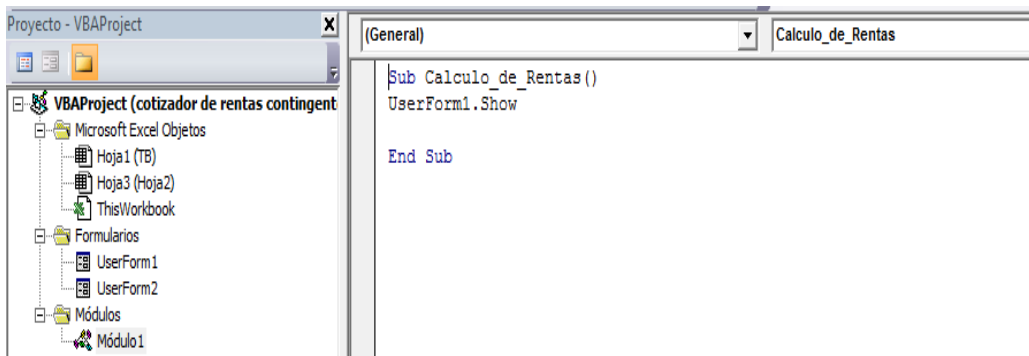
Para una presentación más fresca y atractiva de nuestro programa adaptaremos el llamado de nuestro formulario a una imagen que hemos editado, para ello damos clic derecho sobre la imagen y seleccionamos asignar macro, finalmente seleccionamos el nombre de nuestra macro y clic en aceptar, como se muestra en la imagen de abajo:



En este cuadro podremos agregar manualmente un módulo que llame al formulario deseado, o solo seleccionar una macros ya establecida y que al presionar el botón se active la macros.

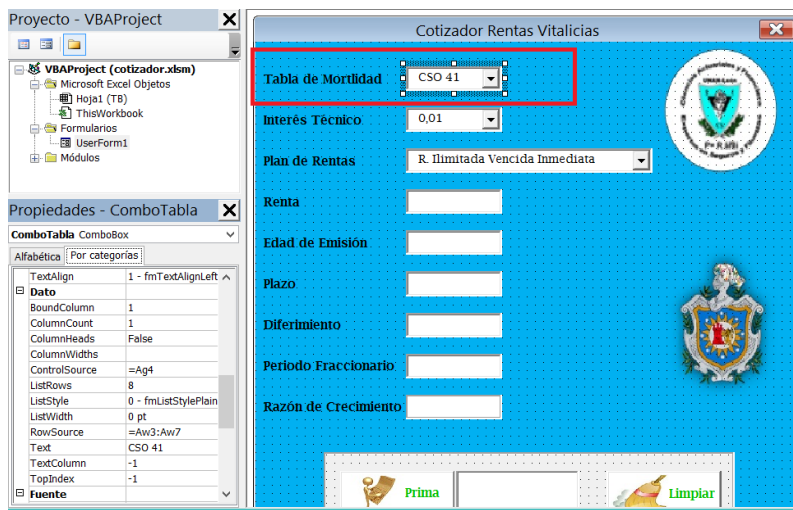


Ahora insertaremos un módulo manualmente, el cual nos servirá para asignar a un botón en nuestra hoja de Excel la llamada a nuestro formulario o userform listo para usarse. (Recordemos que sólo mostramos los pasos básicos para diseñar y estructurar un userform similar al nuestro)

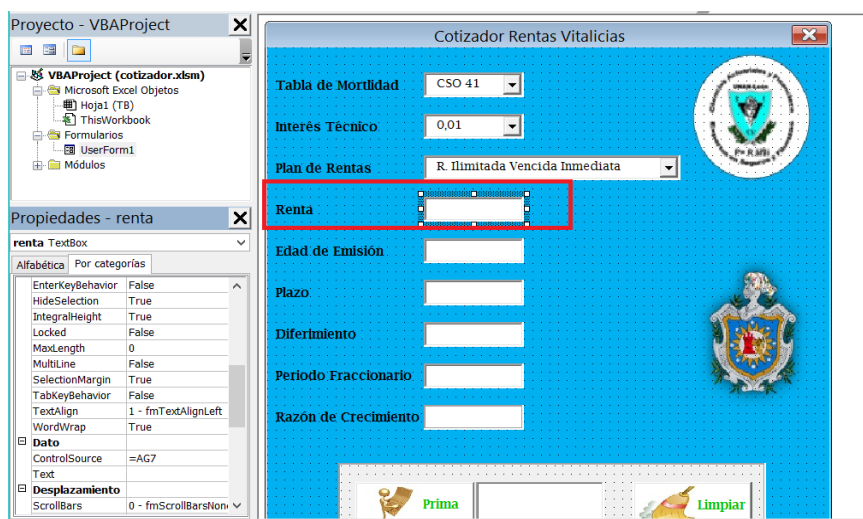


Este módulo no es nada más que la instrucción que hacemos de mostrar el userform1 al darle clic sobre el botón que asignemos en nuestra hoja de Excel. Esto lo hacemos a través del comando “*Userform1.Show*”, listo ahora que demos clic sobre la imagen que hemos designado como nuestro botón en Excel el formulario aparecerá en pantalla.

Una vez agregadas todas las etiquetas, cuadros de textos, listas desplegables y botones, estamos listos para asignarles un valor a cada uno, re direccionar a la celda en la que queremos que se introduzcan los valores que agreguemos a nuestro formulario en funcionamiento, esto como otras opciones de diseño lo hacemos desde la ventana de propiedades (aparece al lado derecho debajo del proyecto) de cada uno de los ítems que hemos agregado, es decir seleccionamos el ítem para que nos muestre la ventana de propiedades acorde a este, y para facilitarnos el manejo de esta la ordenaremos por categoría.

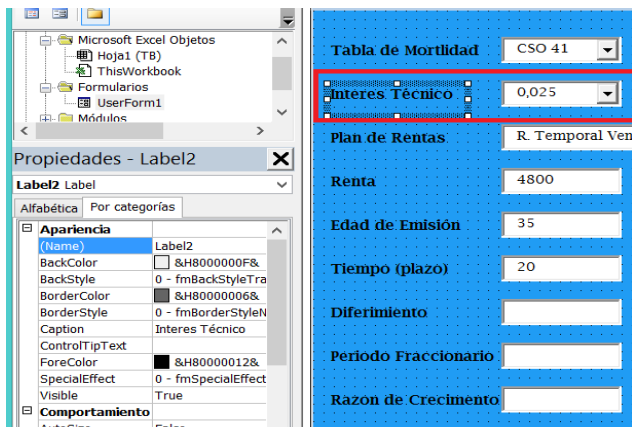


En la imagen anterior podemos ver que tenemos seleccionada la lista desplegable de tablas de mortalidad, para asignar entonces el rango en donde se encuentra esta lista y que se muestre en nuestro formulario vamos a ubicarnos en la ventanilla de propiedades, bajamos hasta la división de datos y buscamos “RowSource” aquí insertaremos o pegaremos el rango en que se encuentran los valores o datos de nuestra lista desplegable, en nuestro caso el rango que utilizamos es AW3:AW7, además de esto también necesitamos asignar la celda a la que se copiará la selección que hagamos de la tabla que usaremos, en esta misma parte de datos, nos ubicamos en “ControlSource” y asignamos la celda AG4 ya establecida en nuestro formato anterior en la hoja de cálculo.





En los cuadros de texto simples se trabaja de manera similar, con la diferencia que no existe una lista previa de opciones por lo tanto el valor a asignar es libre, es decir que solo encontraremos la opción “ControlSource” en esta insertaremos en el caso de la renta la celda AG7. Entonces estos dos pasos para asignar celdas los repetiremos para cada uno de los cuadros de textos y listas desplegables que hubiésemos insertado en nuestro formulario.



Para que las etiquetas muestren el nombre que deseamos por ejemplo Interés Técnico se puede realizar de distintas maneras, las más simples son doble clic (lento) sobre la etiqueta y asignar el nombre deseado. Y la segunda opción en la ventana de propiedades buscaremos la opción “Caption” e introducimos ahí el nombre deseado, esto hace que en nuestro formulario se asigne el nombre que estamos colocando a esta etiqueta.





Una vez realizado todos los cambios mencionados para todos y cada uno de los objetos anteriores para probar el programa ejecutaremos la macros esperando obtener un formulario similar al que estamos mostrando arriba, la única variante que debería de verse es el cambio de fondo que le hemos asignado nosotros, de igual manera lo aplicamos haciendo uso del cuadro de propiedades, pero esta vez seleccionamos el formulario y en sus propiedades buscamos el nombre picture que es la propiedad que nos permite colocar un fondo a nuestro formulario.

Corriendo ya nuestra macros probamos el botón “PRIMA” que nos debe de mostrar el valor de la prima neta única en la caja de texto que se encuentra a la derecha del botón, acorde a todas las variables que solicite el plan que hemos ingresado, así también el resultado debe mostrarse con separador de mil y únicamente con dos decimales.

Así también si probamos nuestro botón de limpiar, debe limpiar los campos de renta, edad de emisión, tiempo, diferimiento, periodo fraccionario, razón de crecimiento y prima. Tal y como se ve en la imagen de abajo:



Al final de todo lo antes explicado, obtendremos nuestro programa de Cálculo de rentas contingentes algo similar al que estamos presentándoles en estas imágenes. Como explicamos antes las variaciones posibles son las imágenes que se utilizaron para darle una apariencia fresca a nuestro trabajo.





5.1.4 Descripción de los códigos utilizados.

Anteriormente se explicó sobre la elaboración del formulario paso a paso, sin embargo se omitió la explicación de algunos códigos utilizados para validar la introducción de los datos, ya que para algunas variables era necesario, y de esta manera presentar un programa bastante completo en cuanto a la eficiencia y coherencia de los cálculos.

El primer conjunto de código utilizado fue en el evento “Initialize” del UserForm, es decir, lo que debe hacer el programa al momento de iniciarse o lo que es lo mismo que acciones deberían cumplir algunas variables en el formulario al momento que este se inicie.

```
UserForm Initialize
Private Sub UserForm_Initialize()
    Range("ag7:ah12").Select
    Selection.ClearContents

    ComboTabla.ListIndex = 0
    ComboInteres.ListIndex = 0
    ComboPlan.ListIndex = 0

    Me.plazo.Enabled = False
    Me.diferimiento.Enabled = False
    Me.periodo.Enabled = False
    Me.razon.Enabled = False

    renta.SetFocus
End Sub
```

Lo primero que se indicó al programa es que borre todos los campos donde se introducen los datos, con la intención que cada vez que iniciemos nuestro formulario los campos aparezcan en blanco y no con datos de algún calculo que hayamos realizado anteriormente, para ello usamos la línea de código:

```
Range ("ag7:ah12").Select
Selection.ClearContents
```

En segundo lugar que las listas desplegables donde se encuentran las tablas de mortalidad, el interés y los planes de rentas, se inicien seleccionando el primer elemento de su lista para ello se utilizó la instrucción ListIndex y tiene asociado el número cero, porque este indica la primera opción de una lista desplegable:

```
ComboTabla.ListIndex = 0
ComboInteres.ListIndex = 0
ComboPlan.ListIndex = 0
```

Dado que el primer plan de renta seleccionado será R. Ilimitada Vencida Inmediata (primero de nuestra lista) entonces vamos a desactivar las variables que no se necesitan para este plan



por tanto hacemos uso de la propiedad Enabled que significa habilitado, por lo que dejamos en False y así dejar deshabilitadas las variables que no usaremos para el primer plan:

Me.plazo.Enabled = False

Me.diferimiento.Enabled = False

Me.periodo.Enabled = False

Me.razon.Enabled = False

El último código de este conjunto fue renta.SetFocus, este nos indica que la variable renta será el cuadro de texto que aparecerá con el enfoque, es decir, el cursor estará en ese cuadro de texto al abrirse la ventana del formulario.

```
Private Sub ComboTabla_KeyPress(ByVal KeyAscii As MSForms.ReturnInteger)
KeyAscii = 0
End Sub
Private Sub ComboInteres_KeyPress(ByVal KeyAscii As MSForms.ReturnInteger)
KeyAscii = 0
End Sub
Private Sub ComboPlan_KeyPress(ByVal KeyAscii As MSForms.ReturnInteger)
KeyAscii = 0
End Sub
```

El siguiente conjunto de código que utilizamos es usando el evento KeyPress en los cuadros de lista desplegable, la instrucción que colocamos (KeyAscii = 0) esta sencilla línea nos indica que al presionar cualquier tecla, no nos alterará nada de nuestras listas, es decir, no permitirá que se escriba ningún tipo de caracteres en estas listas desplegables, debido a que KeyAscii nos indica el código de la tecla pulsada y al tener valor cero presionar cualquier tecla es nulo.

El parámetro KeyAscii como ya hemos dicho nos indica el código de la tecla pulsada, este código nos lo da como valor numérico, no como una cadena, es decir que si pulsamos la tecla de la pleca o barra (/) valdrá 47, ya que ese es el valor ASCII de ese carácter.

```
Private Sub renta_KeyPress(ByVal KeyAscii As MSForms.ReturnInteger)
If KeyAscii < 46 Or KeyAscii > 57 Or KeyAscii = 47 Then
KeyAscii = 0
End If
End Sub
```

En esta ocasión en el cuadro de texto llamado renta, que es donde se introducirá el valor de la renta, hemos validado la introducción del dato, es decir, con el evento KeyPress esta vez estamos indicando que si se presionan las teclas menor a 46 o mayor a 57 o exactamente la



numero 47 entonces que no deje escribir el valor de esas teclas, en otras palabras no deseamos que nos permita escribir letras ni ningún símbolo, sino que solo números y el punto, ya que los números en ASCII van del 48 al 57 y el punto es el 46.

```
Private Sub edad_KeyPress(ByVal KeyAscii As MSForms.ReturnInteger)
If KeyAscii < 48 Or KeyAscii > 57 Then
KeyAscii = 0
End If
End Sub
```

Nuestra siguiente variable es la edad la cual validamos que solo acepte que se escriban números enteros, esto lo logramos de igual forma utilizando el evento KeyPress definiendo que solo se puedan pulsar las teclas de los números, en el código ASCII estos van del 48 (0) al 57 (9).

```
Private Sub plazo_KeyPress(ByVal KeyAscii As MSForms.ReturnInteger)
If KeyAscii < 48 Or KeyAscii > 57 Then
KeyAscii = 0
End If
End Sub

Private Sub diferimiento_KeyPress(ByVal KeyAscii As MSForms.ReturnInteger)
If KeyAscii < 48 Or KeyAscii > 57 Then
KeyAscii = 0
End If
End Sub

Private Sub periodo_KeyPress(ByVal KeyAscii As MSForms.ReturnInteger)
If KeyAscii < 48 Or KeyAscii > 57 Then
KeyAscii = 0
End If
End Sub

Private Sub razon_KeyPress(ByVal KeyAscii As MSForms.ReturnInteger)
If KeyAscii < 46 Or KeyAscii > 57 Or KeyAscii = 47 Then
KeyAscii = 0
End If
End Sub
```

Para no caer en repetir explicaciones de códigos cabe señalar que en las demás variables como el plazo, diferimiento, periodo fraccionario y razón de crecimiento, se hizo uso del código del evento KeyPress, siempre siguiendo la intención que en los cuadros de textos nos deje escribir únicamente números enteros, exceptuando la variable de razón de crecimiento que si aceptará valores con decimales tal y como se validó la variable renta.



```
ComboPlan          Change
Private Sub ComboPlan_Change()
    plan = Me.ComboPlan.ListIndex

    Select Case plan

        Case "0", "1"
            Me.renta.Enabled = True
            Me.edad.Enabled = True

            Me.plazo.Value = 0
            Me.diferimiento.Value = 0
            Me.periodo.Value = 0
            Me.razon.Value = 0

            Me.plazo.Enabled = False
            Me.diferimiento.Enabled = False
            Me.periodo.Enabled = False
            Me.razon.Enabled = False

        Case "2", "3"
            Me.renta.Enabled = True
            Me.edad.Enabled = True
            Me.diferimiento.Enabled = True

            Me.plazo.Value = 0
            Me.periodo.Value = 0
            Me.razon.Value = 0

            Me.plazo.Enabled = False
            Me.periodo.Enabled = False
            Me.razon.Enabled = False
```

Una vez ya validadas todas las variables, procedemos a una nueva instrucción de códigos; debido a la cantidad de planes de rentas y que no todos hacen uso de todas las variables, definiremos en el programa que al momento de seleccionar un tipo de plan se activen únicamente las casillas de las variables para dicho plan correspondiente. Para ello usamos la instrucción `Select Case` que nos permite ejecutar una o más sentencias según el valor que tenga determinada variable.

En el primer caso seleccionamos de la lista desplegable los dos primeros tipos de rentas, estos son las rentas ilimitadas anticipada o vencida, lo cual sus únicas variables activas deben ser la renta y la edad, a como se ve en las dos primeras líneas del código, dejamos en true la propiedad `enabled` (activadas) y el resto de variables que no se necesitan se dejan con valor cero y desactivadas para que no se pueda escribir nada en esas variables. Y así sucesivamente se validaron los demás tipos de rentas.

5.2 Resolución de ejercicios prácticos.

Realizamos este programa con la finalidad de facilitar una herramienta de estudio y ayuda para el cálculo de las rentas vitalicias, un material de ayuda, agilización, y apoyo para los estudiantes de la carrera Ciencias Actuariales y Financieras, y a todos aquellos interesados en estos cálculos, o inclusive en la realización de una macros en formato formulario simple.



Este trabajo esta propuesto para dar a entender que las ciencias actuariales se vinculan fácilmente con la programación de software y diseño de los mismos para elaborar herramientas que nos faciliten los cálculos y trabajos a realizar.

Para probar nuestro programa (macros), pasaremos a realizar 4 ejemplos del cálculo de las primas netas únicas de rentas vitalicias, plantearemos 4 ejemplos específicos para 4 distintos planes de rentas, para rentas ilimitadas, temporales, fraccionarias y variables.

Ejercicios resueltos.

- 1) Si se concede a Martha de 35 años de edad, una renta ordinaria contingente a 20 años de \$4,800 anuales, ¿Cuál es su VAA? Mediante la tabla CSO 41 al 2.5%.

Datos:

Renta. Temporal inmediata y vencida

X: 35

n: 20

R: 4,800 u.m

VAA=?

$$VAA = R a_{x:n|}$$
$$VAA = 4800 a_{35:20|} = 4800 * \left[\frac{N_{36} - N_{56}}{D_{35}} \right]$$
$$VAA = 4800 * \left[\frac{79,449,125.86 - 25,030,062.06}{37,337,05.46} \right]$$
$$VAA = \$69,960.3943$$



- 2) La Señora Carolina compra a la edad de 52 años, una renta vitalicia de \$2,500 cuyo primer pago debería hacerse cuando cumpla 60 años, cual es el VAA? Utilizar la tabla CSO 41 al 2.5%.

DATOS

Renta ilimitada, diferida y anticipada

X: 52 años

R: \$2,500

VAA: ?

x+r: 60

r: 8

$$VAA = R {}_r\ddot{a}_x = 2500 {}_8\ddot{a}_{52} = 2500 \left[\frac{N_{60}}{D_{52}} \right]$$
$$VAA = 2500 \left[\frac{18,234,901.00}{2,139,055.73} \right]$$
$$VAA = \$21,311.8582$$

Cotizador de Rentas Vitalicias

Tabla de Mortalidad: CSO 41

Interés Técnico: 0,025

Plan de Rentas: R. Ilimitada Anticipada y Diferida

Renta: 2500

Edad de Emisión: 52

Plazo: 0

Diferimiento: 8

Periodo Fraccionario: 0

Razón de Crecimiento: 0

Prima: 21,311.86

Logos: UNAN-León, APFRE, SEGUROS AMERICA



- 3) Hallar la prima neta única de una anualidad vitalicia prepagable de \$1,000 mensuales, para una persona de 30 años de edad. Usando la tabla CSO 58 al 4.5%.

Datos

R. Fraccionaria anticipada inmediata e ilimitada

R= 1,000

X= 30

m=12

VAA=?

$$VAA = R(m)[\ddot{a}_x^m] = R(m) \left[\ddot{a}_x - \frac{m-1}{2m} \right]$$

$$VAA = 1000(12) \left[\ddot{a}_{30} - \frac{12-1}{2(12)} \right]$$

$$VAA = 12000 \left[\frac{N_{30}}{D_{30}} - \frac{11}{24} \right]$$

$$VAA = 12000 \left[\frac{474,691.32}{25,312.55} - \frac{11}{24} \right]$$

$$VAA = \$219,538.4034$$



- 4) Determinar valor actual actuarial de una renta vitalicia, pagadera a Alexis de 35 años, además se conoce que el primer término es de \$6,500, y cada año incrementarán en \$1,500. Utilizar la tabla CSO 58 al 4.5%.

Datos

R.V. inmediata, anticipada e ilimitada

X=35

R=6500

h=1500

VAA=?

$$\begin{aligned} VAA &= (V\ddot{a}C)_{35} = R \left(\frac{N_x}{D_x} \right) + h \left(\frac{S_{x+1}}{D_x} \right) \\ VAA &= 6500 \left(\frac{N_{35}}{D_{35}} \right) + 1500 \left(\frac{S_{36}}{D_{35}} \right) \\ VAA &= 6500 \left(\frac{359,054.44}{20,083.79} \right) + 1500 \left(\frac{4,833,734.19}{20,083.79} \right) \\ VAA &= \$477,223.36 \end{aligned}$$

El software 'Cotizador de Rentas Vitalicias' presenta una interfaz con los siguientes campos de entrada y resultados:

- Tabla de Mortalidad: CSO 58
- Interés Técnico: 0,045
- Plan de Rentas: R.V.A. Ilimitada Anticipada Inmediat
- Renta: 6500
- Edad de Emisión: 35
- Plazo: 0
- Diferimiento: 0
- Periodo Fraccionario: 0
- Razón de Crecimiento: 1500

El resultado calculado se muestra en el campo 'Prima' con el valor 477,223.36. El botón 'Limpia' permite reiniciar los cálculos.



VI. CONCLUSIONES

Al finalizar nuestro trabajo investigativo, concluimos que:

La renta vitalicia es una renta líquida que se percibirá, si y solo si, la persona permanezca con vida. Es una manera segura, rentable y muy atractiva de convertir en un flujo de ingresos una cantidad que tengamos ahorrada. Así mismo estas se clasifican en ilimitadas, temporales, fraccionarias, y variables, que a su vez pueden ser de forma anticipada o vencida.

En Nicaragua no es muy conocida la renta vitalicia, por la poca existencia de profesionales en la materia, sin embargo, este tipo de inversión es la más rentable para aquellas personas que después de una prolongada etapa de trabajo lo que desea es descansar, aprovechar mejor su tiempo, compartirlo con su familia y sobre todo disponer de una pensión que otorgue estabilidad financiera. Por tanto para las aseguradoras de nuestro país es muy viable tomar visión en este tipo de seguro.

Excel es una de las herramientas más eficientes, de fácil y común acceso, porque nos permite realizar diferentes cálculos, actividades, y análisis en las distintas áreas. Por ejemplo la utilización de tablas de mortalidad dinámicas, y la aplicación de estas mismas en el formulario nos facilitaron el tratamiento de los datos, utilizando diferentes probabilidades de muerte y la facilidad de cambiar las tablas con pocos clics. Por tanto gracias a esta plataforma de Excel podemos realizar en pocos minutos cálculos complejos, a través de simples conjuntos de fórmulas automáticas.

El Visual Basic como herramienta de Excel es una aplicación muy eficaz y eficiente al procesar, programar, y desarrollar información al momento de trabajar con macros, automatizando las funciones que debe realizar un determinado programa, gracias a esta herramienta construimos una interfaz de cálculos de primas que al introducir datos necesarios nos dan sus resultados.



VII. RECOMENDACIONES

Recomendamos a:

Estudiantes: Es importante señalar que abordamos casi la totalidad de las rentas vitalicias posibles, pero a pesar de todo debido a la longitud de este tema dejamos fuera del programa las rentas vitalicias variables en progresión geométrica, por lo cual recomendamos se tomen en cuenta para brindar una mayor solidez a estos cálculos. Además también señalamos que nuestro programa es meramente para el cálculo de la prima neta única, por lo tanto no toma en consideración el cálculo despejado de alguna de las otras variables disponibles, es decir que no presenta método automático alguno para el cálculo de las distintas variables. Este aspecto se podría tomar en cuenta en el diseño, agregando formularios distintos o anexos al programa inicial que faciliten los cálculos de las distintas variables.

A su vez incentivamos, también a los estudiantes egresados y profesionales a la realización de nuevas investigaciones acerca de las rentas vitalicias o contingentes, ampliando o mejorando investigaciones anteriores para la actualización de estas, aportando así un mayor enriquecimiento de información para todo aquel interesado en este tipo de rentas.

Nuestra alma mater: En cuanto a nuestra alma mater UNAN-León la utilización de este trabajo como material bibliográfico será un apoyo a las generaciones futuras, teniendo en cuenta la problemática que presentan los estudiantes al momento de buscar información relacionada a la ciencia actuarial. Por tanto recomendamos la reproducción de este tipo de investigaciones que ayudarán a la formación y crecimiento académico de la propia universidad, porque una universidad sin estudios investigativos no es universidad.



VIII. BIBLIOGRAFÍA

Referencias Bibliográficas:

1. Noel Reyes Alvarado; *Matemática Financiera*. UNAN Managua: Edición de la facultad de Ciencias Económicas; 2000.
2. Nieto de Alba Ubaldo, *Matemática Actuarial*. Madrid: MAPFRE; 1993.
3. José Antonio Gil Fana, Antonio Heras Martínez, José Luis Vilar Zanón; *Matemática de los Seguros de Vida*. Madrid: MAPFRE; 1999.
4. Hugo E. Palacios; *Introducción al cálculo actuarial*. Madrid: MAPFRE; 1996.

Referencias Electrónicas:

1. Eloy Pozo Carrero, Javier Zúñiga Rodríguez; *Análisis y formulación de las operaciones financieras*. (20/09/2014). [Libro en línea]. Colección de e-books de google. Recuperado de:
http://books.google.com.ni/books?id=RFksOsSpJLwC&pg=PA114&lpg=PA114&dq=rentas+contingentes&source=bl&ots=ZUEv7doam1&sig=L8y9_dsj-IwMbSvl69oaFnZru5s&hl=es-419&sa=X&ei=kIjRU9S7GKrgsASfkYGICw&redir_esc=y#v=twopage&q=rentas%20contingentes&f=false
2. Excel Total. (10/10/2014). Recuperado de:
www.exceltotal.com
3. MICROSOFT OFFICE. (12/10/2014). Recuperado de:
www.office.microsoft.com



IX. ANEXOS

Tasas de mortalidad					
	CSO 41	CSO 58	GAM-71	CSO 80	CET 58
	General	General	Hombres	General	General
0	0.02258	0.00708	0.00167	0.00418	0.00783
1	0.00577	0.00176	0.00043	0.00107	0.00251
2	0.00414	0.00152	0.00040	0.00099	0.00227
3	0.00338	0.00146	0.00039	0.00098	0.00221
4	0.00299	0.00140	0.00038	0.00095	0.00215
5	0.00276	0.00135	0.00046	0.00090	0.0021
6	0.00261	0.00130	0.00042	0.00086	0.00205
7	0.00247	0.00126	0.00040	0.00080	0.00201
8	0.00231	0.00123	0.00039	0.00076	0.00198
9	0.00212	0.00121	0.00039	0.00074	0.00196
10	0.00197	0.00121	0.00039	0.00073	0.00196
11	0.00191	0.00123	0.00040	0.00077	0.00198
12	0.00192	0.00126	0.00040	0.00085	0.00201
13	0.00198	0.00132	0.00041	0.00099	0.00207
14	0.00207	0.00139	0.00042	0.00115	0.00214
15	0.00215	0.00146	0.00043	0.00133	0.00221
16	0.00219	0.00154	0.00044	0.00151	0.00229
17	0.00225	0.00162	0.00046	0.00167	0.00237
18	0.00230	0.00169	0.00047	0.00178	0.00244
19	0.00237	0.00174	0.00049	0.00186	0.00249



20	0.00243	0.00179	0.00050	0.0019	0.00254
21	0.00251	0.00183	0.00052	0.00191	0.00258
22	0.00259	0.00186	0.00054	0.00189	0.00261
23	0.00268	0.00189	0.00057	0.00186	0.00264
24	0.00277	0.00191	0.00059	0.00182	0.00266
25	0.00288	0.00193	0.00062	0.00177	0.00268
26	0.00299	0.00196	0.00065	0.00173	0.00271
27	0.00311	0.00199	0.00068	0.00171	0.00274
28	0.00325	0.00203	0.00072	0.0017	0.00278
29	0.00340	0.00208	0.00076	0.00171	0.00283
30	0.00356	0.00213	0.00081	0.00173	0.00288
31	0.00373	0.00219	0.00086	0.00178	0.00294
32	0.00392	0.00225	0.00092	0.00183	0.003
33	0.00412	0.00232	0.00098	0.00191	0.00307
34	0.00435	0.00240	0.00105	0.002	0.00315
35	0.00459	0.00251	0.00112	0.00211	0.00326
36	0.00486	0.00264	0.00120	0.00224	0.00343
37	0.00515	0.00280	0.00130	0.0024	0.00364
38	0.00546	0.00301	0.00140	0.00258	0.00391
39	0.00581	0.00325	0.00151	0.00279	0.00423
40	0.00618	0.00353	0.00163	0.00302	0.00459
41	0.00659	0.00384	0.00179	0.00329	0.00499
42	0.00703	0.00417	0.00200	0.00356	0.00542



43	0.00751	0.00453	0.00226	0.00387	0.00589
44	0.00804	0.00492	0.00257	0.00419	0.0064
45	0.00861	0.00535	0.00292	0.00455	0.00696
46	0.00923	0.00583	0.00332	0.00492	0.00758
47	0.00991	0.00636	0.00375	0.00532	0.00827
48	0.01064	0.00695	0.00423	0.00574	0.00904
49	0.01145	0.00760	0.00474	0.00621	0.00988
50	0.01232	0.00832	0.00529	0.00671	0.01082
51	0.01327	0.00911	0.00587	0.0073	0.01182
52	0.01430	0.00996	0.00648	0.00796	0.01295
53	0.01543	0.01089	0.00713	0.00871	0.01416
54	0.01665	0.01190	0.00781	0.00956	0.01547
55	0.01798	0.01300	0.00852	0.01047	0.0169
56	0.01943	0.01421	0.00926	0.01146	0.01847
57	0.02100	0.01554	0.01004	0.01249	0.0202
58	0.02271	0.01700	0.01089	0.01359	0.0221
59	0.02457	0.01859	0.01192	0.01477	0.02417
60	0.02659	0.02034	0.01312	0.01608	0.02644
61	0.02878	0.02224	0.01444	0.01754	0.02891
62	0.03118	0.02431	0.01586	0.01919	0.0316
63	0.03376	0.02657	0.01741	0.02106	0.03454
64	0.03658	0.02904	0.01918	0.02314	0.03775
65	0.03964	0.03175	0.02126	0.02542	0.04128



66	0.04296	0.03474	0.02364	0.02785	0.04516
67	0.04656	0.03804	0.02632	0.03044	0.04945
68	0.05046	0.04168	0.02919	0.03319	0.05418
69	0.05470	0.04561	0.03243	0.03617	0.05929
70	0.05930	0.04979	0.03611	0.03951	0.06473
71	0.06427	0.05415	0.04001	0.0433	0.0704
72	0.06966	0.05865	0.04383	0.04765	0.07625
73	0.07550	0.06326	0.04749	0.05264	0.08224
74	0.08181	0.06812	0.05122	0.05819	0.08856
75	0.08864	0.07337	0.05529	0.06419	0.09538
76	0.09602	0.07918	0.06007	0.07053	0.10293
77	0.10399	0.08570	0.06592	0.07712	0.11141
78	0.11259	0.09306	0.07260	0.0839	0.12098
79	0.12186	0.10119	0.07969	0.09105	0.13155
80	0.13185	0.10998	0.08743	0.09884	0.14297
81	0.14260	0.11935	0.09544	0.10748	0.15516
82	0.15416	0.12917	0.10369	0.11725	0.16792
83	0.16657	0.13938	0.11230	0.12826	0.18119
84	0.17988	0.15001	0.12112	0.14025	0.19501
85	0.19413	0.16114	0.13010	0.15295	0.20948
86	0.20937	0.17282	0.13931	0.16609	0.22467
87	0.22563	0.18513	0.14871	0.17955	0.24067
88	0.24300	0.19825	0.15849	0.19327	0.25773



89	0.26144	0.21246	0.16871	0.20729	0.2762
90	0.28099	0.22814	0.17945	0.22177	0.29658
91	0.30173	0.24577	0.19092	0.23698	0.3195
92	0.32364	0.26593	0.20126	0.25345	0.34571
93	0.34666	0.28930	0.21299	0.27211	0.37609
94	0.37100	0.31666	0.22653	0.2959	0.41166
95	0.39621	0.35124	0.24116	0.32996	0.45661
96	0.44719	0.40056	0.25620	0.38455	0.52073
97	0.54826	0.48842	0.27248	0.4802	0.63495
98	0.72467	0.66815	0.29016	0.65798	0.8686
99	1.00000	1.00000	0.30912	1.00000	1