|  |  |
| --- | --- |
| CrearPolinomio | |
| Entrada: | Lista de String cargada parámetros |
| Salida: | Nuevo Polinomio almacenado en el ABBPolinomio |
| Método: | Creo nueva Lista de terminos  Defino variable signo de tipo char  Defino variable base de tipo int  Defino variable cant\_terminos con CantItemsEnLista y le resto 2  Tomo el tercer string de la lista de parametros  Mientras no se llegue al final de la lista  Tomo el primer carácter del string  Si carácter es igual a ‘-‘  Guardo en variable signo ‘-‘  Tomo resto del string y lo guardo en una nueva variable temp  Convierto el string temp a numero y lo guardo en variable base  FinSi  Sino  Guardo en variable signo ‘+’  Convierto el string a numero entero y lo guardo en variable base  FinSino  Creo Termino con variables base, signo y cant\_terminos como exponente  Agrego termino al fila nueva lista de terminos  Resto 1 a cant\_terminos  Tomo siguiente string de la lista  FinMientras  Creo polinomio con nombre y la nueva lista de términos  Agrego polinomio a ABB |
|  |  |

|  |  |
| --- | --- |
| MostrarABBPolinomio | |
| Entrada: | ABB polinomios |
| Salida: | Salida por pantalla |
| Método: | Si el ABB no es vacio  MostrarABBPolinomio ABB hijo izquierdo  Tomo Polinomio del ABB  MostrarPolinomio tomado del ABB  Desplego en pantalla salto de linea  MostrarABBPolinomio ABB hijo derecho  FinSi |

|  |  |
| --- | --- |
| MostrarPolinomio | |
| Entrada: | Polinomio |
| Salida: | Salida por pantalla |
| Método: | Muestro nombre polinomio  Muestro “ = “  Tomo lista de terminos  Tomo primer termino de la lista  Mientras que la lista no sea vasia  MuestroTermino de la lista  Despliego “ “ por pantalla  Tomo el proximo termino de la lista  FinMientras |

|  |  |
| --- | --- |
| Multiplicar | |
| Entrada: | 2 Polinomios String |
| Salida: | Un nuevo Polinomio |
| Método: | Obtengo lista terminos de 1er polinomio  Obtengo lista términos de 2do polinomio  Creo variable char Signo  Creo Lista de terminos1, Lista de términos 2  Recorro primer lista  Mientras lista1 no sea vacía obtengo termino  Recorro segunda lista  Mientras lista2 no sea vacía obtengo termino  Obtengo exponentes de ambos términos y los sumo  Obtengo bases de ambos términos y multiplico  Obtengo signos de ambos términos  Si ambos signos son ‘-’ o ambos signos son ‘+’  Asigno ‘+’ a variable Signo  Sino  Asigno ‘-‘ a variable Signo  Creo termino con dichos datos  Inserto termino en lista términos1  FinMientras  FinMientras  Recorro lista términos1  Mientras lista terminos1 no esta vacía  Si lista de términos2 esta vacía  Ingreso termino  Sino  Obtengo exponente de termino de lista1  Recorro lista de términos 2  Mientras lista no sea vacía  Obtengo exponente de término de lista2  Si ambos exponentes son iguales  Sumo bases de ambos terminos  FinSi  FinMientras  FinSi  FinMientras  Creo nuevo polinomio con String de nombre y Lista de términos2  \\Continua |

|  |  |
| --- | --- |
| MostrarTermino | |
| Entrada: | Termino |
| Salida: | Salida por pantalla |
| Método: | Si la base del termino es diferente a 0  Despliego por pantalla signo del termino  Despliego por pantalla base del termino  Despliego por pantalla “x”  Si el exponente del termino es diferente a 1 y diferente a 0  Despliego por pantalla el exponente  FinSi |

|  |  |
| --- | --- |
| Evaluar | |
| Entrada: | Polinomio Entero para evaluación |
| Salida: | Valor del polinomio dado el entero de entrada |
| Método: | Creo variable total = 0  Creo variable temp  Tomo lista de terminos  Obtengo termino  Mientras que la lista no sea nula  Si base es diferente a 0  . Obtengo exponente  . Defino temp = entrada  . Si exponente es mayor a 1  . . Defino variable i igual a exponente  . . Mientras i sea mayor a 1  . . . Temp = Temp \* entrada  . . . Resto 1 a variable i  . . FinMientras  . . Obtengo signo  . . Temp = temp \* base  . FinSi  . Sino  . . Si exponente es igual a 1  . . Multiplico entrada por base y lo guardo en Temp  . . FinSI  . . Sino  . . . Si exponente = 0  . . . Temp = a base.  . . . FinSI  . . FinSino  . FinSino  FinSI  Si signo = ‘-‘  Temp = Temp por -1  Total = Total + Temp  Tomo siguiente termino  FinMientras Devuelvo valor total por pantalla |

|  |  |  |
| --- | --- | --- |
| EsRaiz | | |
| Entrada: | Polinomio  Entero para calcular | |
| Salida: | TRUE, FALSE | |
| Método: | Creo variable Boolean en FALSE  Si EvaluarPolinomio de valor ingresado es igual a 0  Guardo variable en TRUE  Retorno variable Boolean | |
| Sumar | | | |
| Entrada: | | Dos polinomios a ser sumados  Polinomio vacio por referencia | |
| Salida: | |  | |
| Método: | | Obtengo términos de primer polinomio  Obtengo términos de segundo polinomio  Creo variable Booleana en FALSE  Creo variable j  Obtengo cantidad de elementos de cada lista de términos.  Si lista1 es más grande o son iguales  Recorro primer lista  Mientras lista1 no sea vacía  Obtengo exponente de término de primer lista  Si signo es igual ‘-1’  Tomo base y multiplico por -1  FinSi  Recorro segunda lista desde j  Mientras lista2 no sea vacía y variable Booleana sea FALSE  Obtengo exponente de término de segunda lista  Si ambos exponentes son iguales  Seteo variable en TRUE  Si signo es igual a ‘-’  Tomo base y multiplico por -1  FinSi  Creo nuevo termino con base igual a suma de ambas bases y exponente  Agrego término a lista de términos  Guardo última posición recorrida en j  Sino  Creo nuevo termino con base igual a base y exponente  Agrego término a lista de términos  FinSi  FinMientras  FinMientras  Sino  Realizo el mismo procedimiento recorriendo primero lista2 y luego lista1.  FinSi  Agrego lista de términos a Polinomio vacío | |

|  |  |
| --- | --- |
| GuardarPolinomio | |
| Entrada: | Polinomio String con el nombre del archivo |
| Salida: |  |
| Método: | Defino f como archivo en modo ‘wb’  Tomo nombre del polinomio  GuardoString en el archivo  Tomo Lista de Terminos  Tomo primer valor de la lista  Mientras que la lista no se nula  GuardarTermino en archivo f  Tomo siguiente valor de la lista  FinMientras |

|  |  |
| --- | --- |
| GuardarString | |
| Entrada: | String Referencia a archivo |
| Salida: | String en el archivo |
| Método: | Tomo primer caracter  Mientas que carácter sea diferenta a fin de linea  Escribo carácter en archivo  Tomo siguiente caracter  FinMientras  Escribo carácter fin de linea en archivo |

|  |  |
| --- | --- |
| GuardarTermino | |
| Entrada: | Termino Referencia a archivo |
| Salida: | Termino en el archivo |
| Método: | Tomo signo del termino y la guardo en el archivo  Tomo base del termino y la guardo en el archivo  Tomo exponente del termino y lo guardo en el archivo |

|  |  |
| --- | --- |
| IngresarComando | |
| Entrada: | Caracteres ingresados por teclado |
| Salida: | - |
| Método: | Creo un string nulo (strcrear) Cargo lo ingresado en un string dinámico (scan) Parsear el string ingresado |

|  |  |
| --- | --- |
| Parsear | |
| Entrada: | String de comandos |
| Salida: | Lista de Strings |
| Método: | Creo nuevo string  Recorro String de comandos  Mientras caracter sea distinto a fin de línea  Si carácter es distinto a espacio  Cargo String con carácter  Sino  Agrego caracter fin de línea  Guardo nuevo String en lista de String  Creo nuevo string  FinMientras |

|  |  |
| --- | --- |
| ValidarComando | |
| Entrada: | String |
| Salida: | TRUE, FALSE |
| Método: | Guardo valor FALSE  Tomo primer enumerado.  Mientras haya enumerados de comandos Y valor guardado sea FALSE  Si el String de entrada es igual al enumerado de comandos  Guardo valor TRUE  Sino  Tomo el proximo enumerado  FinMientras  SI el valor guardado es TRUE  Devuelvo TRUE  Sino  Devuelvo FALSE |

|  |  |
| --- | --- |
| EsValidoNombre | |
| Entrada: | String cargado |
| Salida: | TRUE, FALSE |
| Método: | Recorro String  Mientras carácter sea distinto a carácter fin de línea y variable sea TRUE  Si carácter >= ‘a’ o caracter <= ‘z’  o carácter >= ‘A’ o caracter <= ‘Z’  o carácter >= ‘0’ o caracter <= ‘9’ entonces  Seteo variable en TRUE  Sino  Seteo variable en FALSE  FinMientras |

|  |  |
| --- | --- |
| EsValidoNumero | |
| Entrada: | String cargado |
| Salida: | TRUE, FALSE |
| Método: | Recorro String  Mientras carácter sea distinto a carácter fin de línea y variable sea TRUE  Si carácter es ‘-’ entonces tomo el siguiente caracter  Si carácter >= ‘0’ y carácter <= ‘9’ entonces  Seteo variable en TRUE  Sino  Seteo variable en FALSE  Sino  Si carácter >= ‘0’ y carácter <= ‘9’ entonces  Seteo variable en TRUE  Sino  Seteo variable en FALSE  FinMientras |

|  |  |
| --- | --- |
| ExistePolinomio | |
| Entrada: | Árbol de búsqueda binario cargado  Nombre de polinomio |
| Salida: | TRUE, FALSE |
| Método: | //Falta hacer |

|  |  |
| --- | --- |
| ConvertirStringATermino | |
| Entrada: | String |
| Salida: | Termino |
| Método: | Defino nuevo termino vacio.  Recorro string  Mientras carácter sea distinto a fin de línea  Si carácter es ‘-‘  Guardo signo en Signo  Sino  Guardo signo ‘+’ en Signo  ConvertirCharANumero  Guardo valor en base  FinSi  FinMientras |

|  |  |
| --- | --- |
| CantItemsEnLista | |
| Entrada: | Lista String cargada |
| Salida: | Int |
| Método: | Declaro variable cant  Recorro lista  Mientras haya ítems en lista  Sumo uno a variable cant  FinMientras |

|  |  |
| --- | --- |
| ConvertirCharANumero | |
| Entrada: | String cargado |
| Salida: | Int |
| Método: | Creo variables cantItems, contador, resultado, multiplicador en 1, i  Recorro String  Mientras carácter sea distinto a carácter fin de línea  Sumo uno a variable cantItems  FinMientras  Asigno valor de cantItems a variable i  Para cada elemento de String mientras contador sea menor a i  Asigno a resultado, (valor de resultado mas valor de carácter en posicion i menos 1) – 48 por multiplicador  Asigno a multiplicador valor de multiplicador, multiplicado por 10  Resto uno a cantItems  FinParaCada  Retorno variable resultado |

MÓDULO MAIN

Defino texto\_entrada y guardo lo ingresado por teclado hasta pulsar Enter

Parsear texto\_entrada a ListaStrings

Defino variable comando de tipo string con el valor del primer elemento de ListaString

Si ValidarComando de variable comando Devuelve TRUE

. Según DarComando de variable comando sea:

. . crear: Tomo el segundo valor de ListaString

. . Si ValidarNombre devuelve TRUE

. . . Si ExistePolinomio devuelve FALSE

. . . . Defino variable cant con CantItemsEnLista y le resto 2

. . . . Tomo tercer valor de la lista

. . . . Guardo valor TRUE

. . . . Mientras que haya elementos en la lista Y valor sea TRUE

. . . . . Si ValidarNumero devuelve FALSE

. . . . . Guardo Valor FALSE

. . . . . FinSi

. . . . . Tomo siguiente valor de la lista

. . . . FinMientras

. . . . Si Valor es TRUE

. . . . . Tomo tercer valor de la lista

. . . . . Convierto CharAInt

. . . . . Si cant es mayor a 1 y valor diferente de 0

. . . . . CrearPolinomio

. . . . . FinSi

. . . . . Sino

. . . . . MostrarError Primer coeficiente 0

. . . . . FinSino

. . . . FinSi

. . . . Sino

. . . . MostrarError Coeficiente invalido

. . . . FinSino

. . . FinSi

. . . Sino

. . . MostrarError polinomio ya existe

. . . FinSino

. . FinSi

. . Sino

. . MostrarError Nombre polinomio Invalido

. . FinSino

. . break;

. . Modificar:

. . ETC:

. FinSegun

. . FinSi

Sino MostrarError de comando no valido

FinSino