UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN

PREPARATORIA 8

Requisitos para presentar 4ª 5ª y/o 6ª oportunidad

Semestre Agosto – Diciembre 2018

Materia: **TECNOLOGÍA DE LA INFORMACIÓN Y DE LA COMUNICACIÓN 2**
TIC 2

Coordinador: María del Carmen Vázquez Castro.

I.- Indicaciones:

- Realizar las Actividades Integradoras de cada Etapa del libro de Texto TIC 2 y grabarlas en un CD o en una USB.
- Imprimir el Laboratorio de TIC 2 y contestarlo en su totalidad.

II. Requisitos:

- Entregar personalmente al Coordinador de TIC en un sobre cerrado y con los datos personales (nombre, turno, matrícula, grupo y materia) el Laboratorio de TIC 2 así como el CD o la USB con las Actividades Integradoras resueltas.
- Fecha de entrega: martes **16 de Octubre de 2018**
- Hora: **11:00 am a 1:00 pm.**
- Lugar: **Laboratorio 1 TIC**
- La entrega del portafolio es exclusivamente con el coordinador, el día, hora y lugar señalado.
- El CD y la USB no se regresan, se quedan en resguardo con el Coordinador.

III.- Para la evaluación sumativa se tomarán en cuenta los siguientes criterios:

El Portafolio	valor de	30%
El examen	valor de	70%
Total calificación		100%

Atentamente

Subdirección Académica

UANL UNIVERSIDAD AUTÓNOMA DE NUEVO LEÓN	**PREPARATORIA 8** **Laboratorio de Tecnología de la Información y de la Comunicación 2**		
Matrícula:	**Nombre:**	**Grupo:**	**Semestre:**

INSTRUCCIONES: Lee cuidadosamente todas y cada una de las preguntas que aparecen a continuación, contesta lo que se te pide en cada punto. Tu maestro te indicará la forma y tiempo en que entregarás este laboratorio.

ETAPA 1 PRINCIPIOS BÁSICOS DE PROGRAMACIÓN

I. COLOCA EN EL PARÉNTESIS LA LETRA QUE CORRESPONDA A CADA UNO DE LOS CONCEPTOS QUE SE PIDEN.

( ) 1.- Es la metodología o proceso para construir un programa. Durante este proceso el programa es escrito de acorde a una sintaxis de un lenguaje de programación, se prueba, se depura, se implanta y se le da mantenimiento.

( ) 2.- Es el conjunto de símbolos, vocablos y reglas (sintaxis) que controlan el comportamiento físico y lógico de una computadora. Permite al programador, entre otras cosas, especificar los datos que van a ser procesados y almacenados en ella.

( ) 3.- Es un conjunto de instrucciones, comandos, órdenes y procedimientos escritos en un lenguaje de programación para que una computadora pueda realizar una tarea determinada de manera exacta y rápida.

( ) 4.- Son las condiciones que debe tener un programa de calidad

A) EXACTO, CLARO, EFICIENTE Y PORTABLE

B) SOFTWARE DE SISTEMA Y DE APLICACIÓN

C) PROGRAMACIÓN

D) LENGUAJE DE PROGRAMACIÓN

E) PROGRAMA

II. SUBRAYA LA RESPUESTA CORRECTA

5.- Fase del proceso de programación donde el problema es definido y comprendido claramente

A) ANÁLISIS DEL PROBLEMA **B) DEFINICIÓN DEL PROBLEMA** **C) DISEÑO**
D) CODIFICACIÓN **E) DEPURACIÓN**

6.- Fase del proceso de programación donde se analiza punto a punto el problema o tarea, es pensar en las operaciones que se necesitan para resolverlo, los tipos de datos que se van a manejar y la relación con la vida cotidiana.

A) ANÁLISIS DEL PROBLEMA **B) DEFINICIÓN DEL PROBLEMA** **C) DISEÑO**
D) CODIFICACIÓN **E) DEPURACIÓN**

7.- Fase donde se diseña el algoritmo, el diagrama de lujo y se lleva a cabo la prueba de escritorio.

A) ANÁLISIS DEL PROBLEMA **B) DEFINICIÓN DEL PROBLEMA** **C) DISEÑO**
D) CODIFICACIÓN **E) DEPURACIÓN**

8.- Es la clasificación de los tipos de datos

A) LÓGICAS **B) SALIDA** **C) CONSTANTES, VARIABLES Y EXPRESIONES**
D) COMPARACIÓN **E) ARITMÉTICAS**

9.- Tipo de datos que no cambian su valor durante el desarrollo o ejecución de un programa

A) EXPRESIONES **B) VARIABLES** **C) CONSTANTES**
D) LÓGICOS **E) ARITMÉTICOS**

10.- Tipo de datos que cambian o modifican su valor durante el desarrollo o ejecución de un programa

A) EXPRESIONES **B) VARIABLES** **C) CONSTANTES**

D) LÓGICOS **E) ARITMÉTICOS**

11.- Forma en que se clasifican las Contantes y las Variables

A) NUMÉRICAS Y LÓGICAS **B) ARITMÉTICAS Y LÓGICAS** **C) NUMÉRICAS Y ALFANUMÉRICAS**

D) OPERADORES Y EXPRESIONES **E) TEXTO Y NÚMEROS**

12.- Forma en que se clasifican los tipos de operadores

A) OPERADORES, EXPRESIONES Y ALFANUMÉRICOS **B) ARITMÉTICOS, DE COMARACIÓN Y LÓGICOS** **C) LÓGICOS, ANALÓGICOS Y DE COMPARACIÓN**

D) TEXTO, NÚMERO Y ALFANUMÉRICO **E) NUMÉRICOS Y ALFANUMÉRICOS**

13.- Del ejemplo siguiente determina a qué tipo de dato pertenece DESCUENTO:

DESCUENTO = (PRECIO*30)/100

A) CONSTANTE NUMÉRICA **B) CONSTANTE ALFANUMÉRICA** **C) VARIABLE NUMÉRICA**

D) EXPRESIÓN **E) VARIABLE ALFANUMÉRICA**

14.- Forma en que se clasifican las Contantes y las Variables

A) NUMÉRICAS Y LÓGICAS **B) ARITMÉTICAS Y LÓGICAS** **C) NUMÉRICAS Y ALFANUMÉRICAS**

D) OPERADORES Y EXPRESIONES **E) TEXTO Y NÚMEROS**

15.- Fase de la metodología de la programación que consiste en desarrollar el procedimiento para resolver el problema.

A) ANÁLISIS DEL PROBLEMA **B) DEFINICIÓN DEL PROBLEMA** **C) DISEÑO**

D) MANTENIMIENTO **E) DEPURACIÓN**

III. RELACIONA AMBAS COLUMNAS

( ) 16.- Conjunto de pasos ordenados de forma lógica que describen la solución de un problema o tarea determinada

( ) 17.- De acuerdo a como son descritos los algoritmos, estos pueden ser de diversos tipos

( ) 18.- Son características de los algoritmos

A) SECUENCIALES, CONDICIONALES Y CÍCLICOS

B) ALGORITMO

C) TENER PRINCIPIO Y FIN; CANTIDAD FINITA DE PASOS; PASOS ORDENADOS LÓGICAMENTE Y LOS PASOS BUSCAN SOLUCIONAR EL PROBLEMA DE MANERA LÓGICA

D) PRUEBA DE ESCRITORIO

IV. RELACIONA AMBAS COLUMNAS

( ) 19.- Es el tipo de algoritmo donde los pasos se ejecutan o describen de principio a fin sin saltos o brincos

( ) 20.- Es el tipo de algoritmo donde se tiene la necesidad de tomar decisiones en alguno de los pasos debido a condiciones que nos llevan a responder verdadero o falso

( ) 21.- Es el tipo de algoritmo donde existen acciones o pasos que se repiten mientras o hasta que una condición se cumpla

A) CONDICIONALES

B) SECUENCIALES

C) CÍCLICOS

D) DE COMPARACIÓN

V. SUBRAYA LA RESPUESTA CORRECTA

22.- Identifica a qué tipo de algoritmo pertenece la siguiente secuencia de pasos:

1. Inicio
2. Pedir la cantidad de horas trabajadas
3. Preguntar si horas trabajadas exceden de 48, entonces
4. No: Calcular sueldo, ir al paso 7
5. Si: Calcular las horas extras
6. Calcular sueldo
7. Imprimir el sueldo a pagar
8. Fin

A) Secuencial

B) Condicional

C) Cíclico

D) Bloques

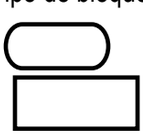
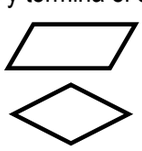
E) Circular

23.- Es la representación gráfica de un algoritmo, es una de las herramientas de programación que permite visualizar de manera general el desarrollo de la solución de un problema

A) DIAGRAMA DE FLUJO **B) ALGORITMO GRÁFICO** **C) FLUJO DE DATOS**

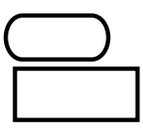
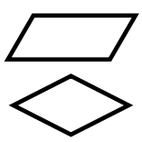
D) FLUJO DE INFORMACIÓN **E) DIAGRAMA DE BLOQUES**

24.- Tipo de bloque que indica en donde comienza y termina el diagrama de flujo

A)  **B)**  **C)** 

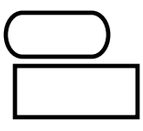
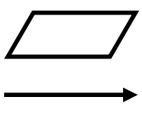
D) **E)**

25.- Tipo de bloque que se utiliza para indicar operaciones aritméticas o transferencias de datos dentro del proceso

A)  **B)**  **C)** 

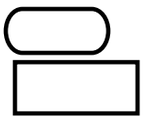
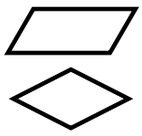
D) **E)**

26.- Tipo de bloque que indica el flujo o dirección del proceso, es la conexión de un bloque a otro

A)  **B)**  **C)** 

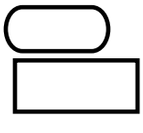
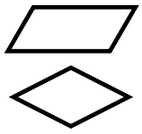
D) **E)**

27.- Tipo de bloque que se utiliza para representar una verificación o comparación lógica dentro del proceso

A)  **B)**  **C)** 

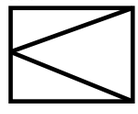
D) **E)**

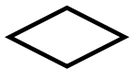
28.- Tipo de bloque que se utiliza para representar una entrada o lectura de datos

A)  **B)**  **C)** 

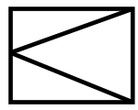
D) **E)**

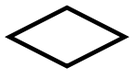
29.- Tipo de bloque que indica la conexión en donde se debe de continuar el flujo de los datos

A)  **B)**  **C)** 

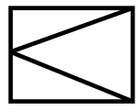
D)  **E)** 

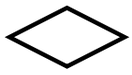
30.- Tipo de bloque que indica el resultado final o parcial del proceso, también se utiliza para dar mensajes o comentarios

A)  **B)**  **C)** 

D)  **E)** 

31.- Tipo de bloque que indica operaciones repetitivas dentro del diagrama de fluijo

A)  **B)**  **C)** 

D)  **E)** 

32.- Los diagramas de flujo como los algoritmos se pueden definir de tres tipos:

A) MATEMÁTICOS, CONDICIONALES Y REPETITIVOS **B) CICLICOS, RECURRENTES Y SECUENCIALES** **C) REPETITIVO, CÍCLICO Y SECUENCIAL**

D) FLUJO CONTINUO, CÍCLICO Y SECUENCIAL **E) SECUENCIALES, CONDICIONALES Y CÍCLICOS**

33.- Consiste en dar valor a las variables en el diagrama de flujo y comprobar si el resultado obtenido es el esperado.

A) VARIABLES DE PRUEBA **B) PRUEBA DE ESCRITORIO** **C) DISEÑO DE ESCRITORIO**

D) MANTENIMIENTO **E) DEPURACIÓN**

VI. RELACIONA AMBAS COLUMNAS

( ) 34.- Es la traducción del algoritmo a un lenguaje de programación de acuerdo a las reglas gramaticales del mismo.

( ) 35.- Fase donde se verifica que el programa haya sido codificado correctamente. El programa se deja sin errores de sintaxis y de lógica.

( ) 36.- Fase donde el programa es instalado en el sistema, se prueba y se corre y se registra su efectividad.

( ) 37.- Fase donde el programa es revisado periódicamente para quitar o agregar operaciones que lo hagan más efectivo.

A) MANTENIMIENTO

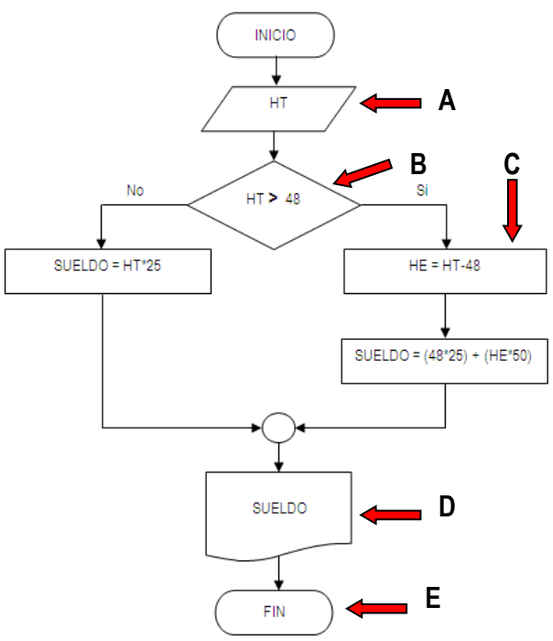
B) DEPURACIÓN

C) CODIFICACIÓN

D) IMPLEMENTACIÓN

E) PRUEBA DE ESCRITORIO

VII. DEL DIAGRAMA DE FLUJO SIGUIENTE IDENTIFICA LOS SÍMBOLOS QUE SE TE PIDEN Y RELACIONALOS CON SU RESPECTIVA FUNCIÓN.



( ) 38.- Se utiliza para indicar operaciones aritméticas o transferencia de datos.

( ) 39.- Se utiliza para representar una verificación o comparación lógica.

( ) 40.- Representa una operación de entrada o lectura de datos.

( ) 41.- Indica el resultado final o parcial del programa y también es utilizado para dar mensajes o comentarios.

VIII. LEE DETENIDAMENTE LAS SIGUIENTES PREGUNTAS Y ANOTA EN EL PARÉNTESIS LA RESPUESTA CORRECTA.

42.- Del programa anterior si las Horas Trabajadas en una semana por un trabajador serían de 48 horas ¿Cuál sería su sueldo semanal (Bloque D)?

A) 2400 **B) 2650** **C) 2150**

D) 1200 **E) 1700**

43.- Del programa anterior si las Horas Trabajadas en una semana por un trabajador serían de 58 horas ¿Cuál sería su sueldo semanal (Bloque D)?

A) 1700 **B) 1200** **C) 2400**

D) 2650 **E) 2150**

44.- Del programa anterior determina a qué tipo de datos pertenece SUELDO

A) VARIABLE NUMÉRICA **B) VARIABLE ALFANUMÉRICA** **C) EXPRESIÓN**

D) CONSTANTE NUMÉRICA **E) CONSTANTE ALFANUMÉRICA**

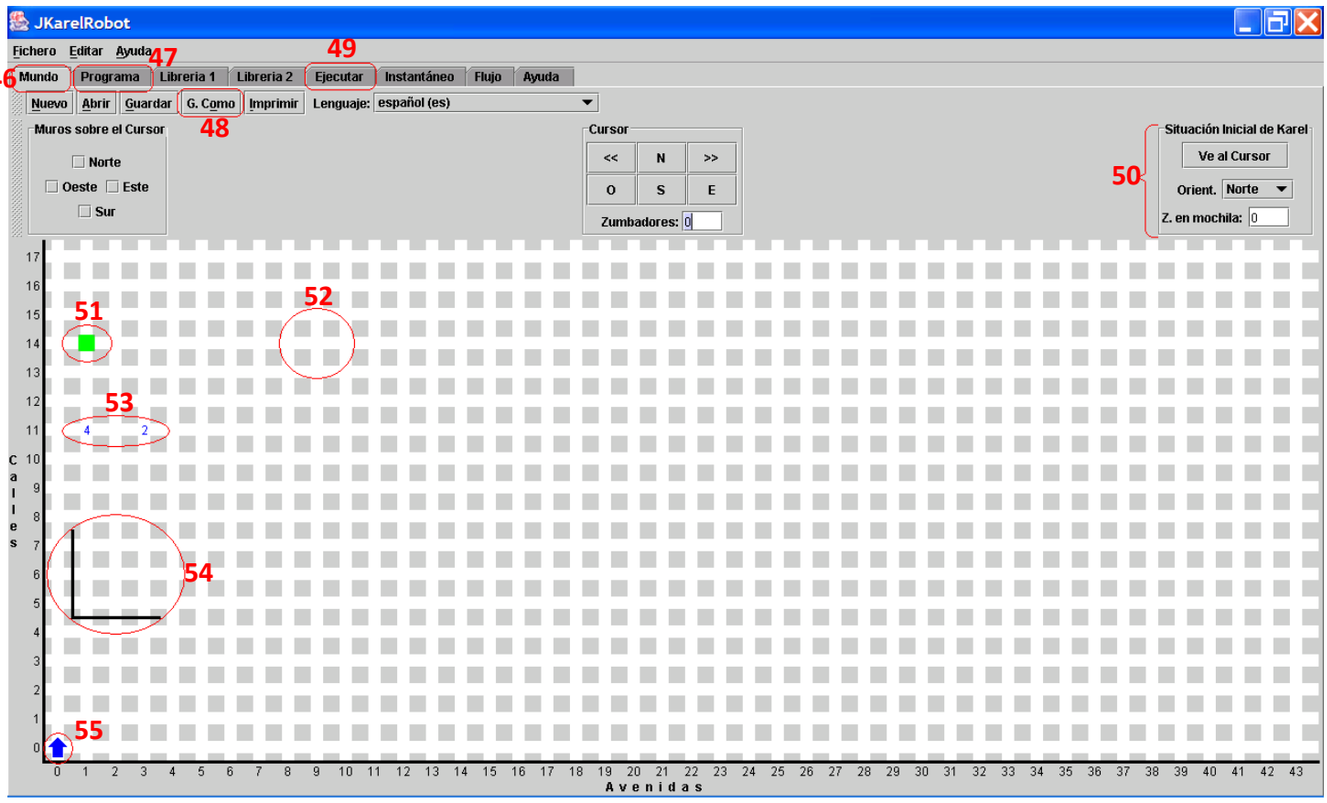
45.- Del programa anterior (en el boque **C**) determina, ¿Qué tipos de operadores se están empleando?

A) ARITMÉTICOS **B) DE COMPARACIÓN** **C) DISYUNCIÓN**

D) LÓGICOS **E) CONJUNCIÓN**

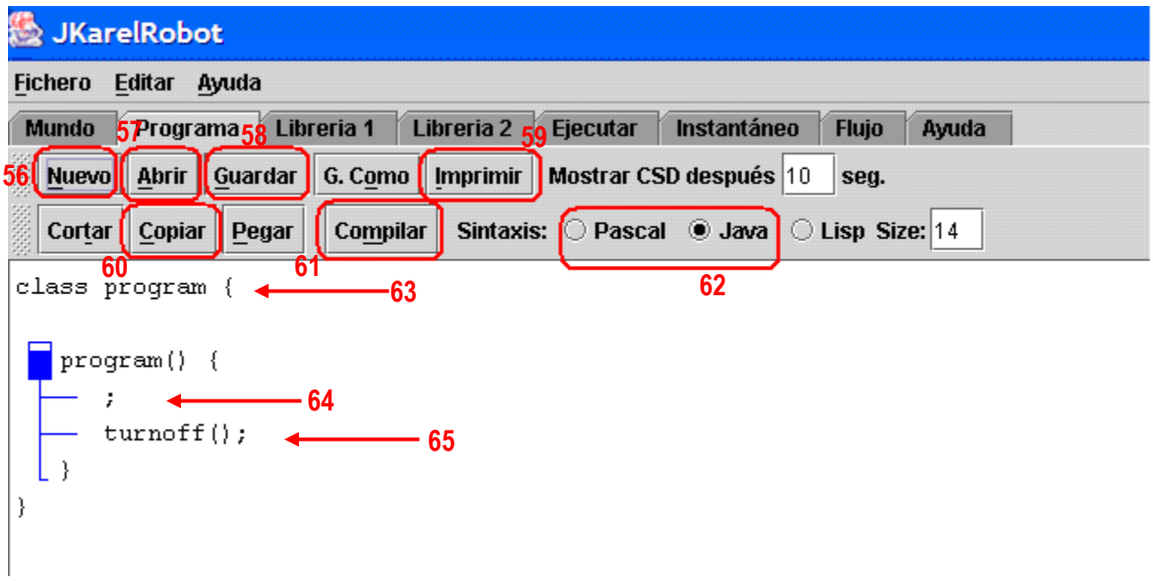
ETAPA 2 AMBIENTE GRÁFICO DEL ROBOT KAREL

IX. DESCRIBE BREVEMENTE EL NOMBRE Y LA FUNCIÓN DE LOS ELEMENTOS SEÑALADOS EN LA PANTALLA DE KAREL



	ELEMENTO	DESCRIPCIÓN
46.-		
47.-		
48.-		
49.-		
50.-		
51.-		
52.-		
53.-		
54.-		
55.-		

X. A PARTIR DE LA SIGUIENTE ILUSTRACIÓN DESCRIBE BREVEMENTE LA FUNCIÓN QUE REALIZA CADA BOTÓN Y SECCIÓN INDICADA.



56.-	
57.-	
58.-	
59.-	
60.-	
61.-	
62.-	
63.-	
64.-	
65.-	

XI. SUBRAYA LA RESPUESTA CORRECTA

66.- Sección de la pantalla de Robot Karel que está dividido en calles y avenidas, además de pequeños cuadros que representan las manzanas de la ciudad.

A) SECCIÓN EJECUTAR **B) SECCIÓN MUNDO** **C) SECCIÓN PROGRAMA**
D) SECCIÓN MOCHILA **E) SECCIÓN CÓDIGO**

67.- Está representado por la flecha azul que se encuentra en la parte inferior izquierda del mundo

A) LA MOCHILA **B) LOS SENSORES** **C) ROBOT KAREL**
D) LAS BARDAS **E) EL CURSOR**

68.- Sirven de referencia para saber la ubicación de Karel, cruzan al Mundo de forma vertical y horizontal

A) LA MOCHILA **B) LOS SENSORES** **C) EL CURSOR**

D) LAS BARDAS **E) CALLES Y AVENIDAS**

69.- Está representado por un recuadro verde que indica la ubicación inicial de trabajo del Robot Karel, en cualquier momento se puede cambiar la ubicación

A) LA MOCHILA **B) LOS SENSORES** **C) TROMPOS O ZUMBADORES**

D) EL CURSOR **E) LAS CALLES Y AVENIDAS**

70.- Es un elemento que contiene el Mundo, para realizar ciertas tareas en conjunto con las calles y avenidas se activan o desactivan dando clic sobre la mitad de una calle o avenida.

A) LOS MUROS O BARDAS **B) LOS SENSORES** **C) LAS CALLES Y AVENIDAS**

D) LA MOCHILA **E) TROMPOS O ZUMBADORES**

71.- Están representados por números arábigos del 1 al 9999999 que podemos colocar en cualquier esquina del Mundo

A) LOS MUROS O BARDAS **B) LOS SENSORES** **C) LAS CALLES Y AVENIDAS**

D) TROMPOS O ZUMBADORES **E) EL CURSOR**

72.- Es un accesorio que contiene Karel para realizar tareas específicas en conjunto con los trompos o zumbadores. Está representado por una caja de texto en donde se registran y pueden ver los trompos o zumbadores

A) LA MOCHILA **B) LOS SENSORES** **C) MUROS O BARDAS**

D) EL CURSOR **E) TROMPOS O ZUMBADORES**

XII. RELACIONA AMBAS COLUMNAS



A) **B)** **C)** **D)** **E)**

( ) 73.- Botón que sirve para generar o iniciar un Mundo.

( ) 74.- Botón que se utiliza para grabar los cambios realizados en el Mundo.

( ) 75.- Botón que se utiliza para grabar por primera vez el Mundo.

( ) 76.- Botón que se utiliza para abrir un Mundo que se encuentra grabado.

XIII. SUBRAYA LA RESPUESTA CORRECTA

77.- Es la Sección o pestaña donde se escriben las instrucciones que se ejecutaran paso a paso. Se pueden escribir en dos tipos de lenguaje.

A) SECCIÓN EJECUTAR **B) SECCIÓN MUNDO** **C) SECCIÓN PROGRAMA**

D) SECCIÓN COMPILAR **E) SECCIÓN IMPRIMIR**

78.- Tipos de lenguajes que se utilizan para escribir las instrucciones en Robot Karel

A) LENGUAJE C **B) LENGUAJE MÁQUINA** **C) PASCAL Y C**

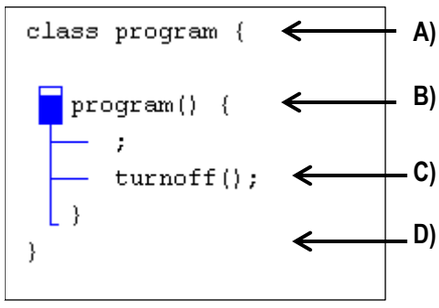
D) PASCAL Y JAVA **E) HTML**

XIV. RELACIONA AMBAS COLUMNAS

( ) 79.- Es la zona para escribir las instrucciones del programa.

( ) 80.- Es la instrucción inicial de todo programa

( ) 81.-. Es la instrucción para finalizar el programa



XV. RELACIONA AMBAS COLUMNAS



B) **B)** **C)** **D)** **E)**

( ) 82.- Se utiliza para revisar línea por línea con el propósito de detectar algún error.

( ) 83.- Se utiliza para sacar una copia de una serie de instrucciones y duplicarlas en otra parte del programa.

( ) 84.- Se encarga de mover o duplicar a otra parte del programa la serie de instrucciones que fueron seleccionadas al cortar o copiar

( ) 85.- Se utilizar para mover una serie de instrucciones a otra parte del programa

XVI. SUBRAYA LA RESPUESTA CORRECTA

86.- Es la Sección o pestaña donde se muestra del lado izquierdo de la pantalla el programa que estamos editando y del lado derecho l mundo que hemos diseñado previamente.

A) SECCIÓN EJECUTAR **B) SECCIÓN MUNDO** **C) SECCIÓN PROGRAMA**
D) SECCIÓN COMPILAR **E) SECCIÓN IMPRIMIR**

87.- Es el botón de la Sección Ejecutar para que aparezca el programa y el mundo en su configuración inicial.

A) ADELANTE **B) SIGUIENTE INSTR** **C) PASOS MIN**
D) INICIALIZAR **E) CORRER**

88 - Es el botón de la Sección Ejecutar que permite visualizar en forma simultánea la ejecución de cada paso en el programa

A) ADELANTE **B) SIGUIENTE INSTR** **C) PASOS MIN**
D) INICIALIZAR **E) CORRER**

89.- Son los pasos que llevarías a cabo para Ejecutar un programa, una vez que hayas elaborado el Mundo de Karel, capturado y compilado el programa.

A) SECCIÓN EJECUTAR: INICIALIZAR Y CORRER
B) SECCIÓN MUNDO: INICIALIZAR Y CORRER
C) SECCIÓN PROGRAMA: INICIALIZAR Y CORRER
D) SECCIÓN EJECUTAR: CORRER E INICIALIZAR
E) SECCIÓN MUNDO: CORRER E INICIALIZAR

90.- Son instrucciones que se emplean para detectar las condiciones del entorno del Robot Karel y van incluidas dentro de un comando. Permiten detectar diversas condiciones dentro de una instrucción if o while, ellos contestaran falso o verdadero según corresponda.

A) BOTONES **B) SENSORES** **C) CURSORES**
D) BARDAS **E) ZUMBADORES**

91.- Sensor que indica ¿Está despejado al frente?

A) frontIsClear ( ) **B) frontIsBlocked ( )** **C) facingEast ( )**
D) rigthIsBlocked ( ) **E) facingNorth ( )**

92.- Sensor que indica ¿Estoy junto a un trompo?

A) anyBeepersInBeeperBag ( ) **B) noBeepersInBeeperBag ( )** **C) nextToABeeper ( )**
D) rigthIsBlocked ( ) **E) notNextToABeeper ( )**

93.- Sensor que indica ¿Estoy en dirección Este?

A) frontIsClear ( ) **B) frontIsBlocked ( )** **C) facingEast ( )**
D) rigthIsBlocked ( ) **E) facingNorth ( )**

94.- Sensor que indica ¿Está bloqueado al frente?

A) frontIsClear ( ) **B) frontIsBlocked ( )** **C) facingEast ( )**
D) facingNorth ( ) **E) rigthIsBlocked ( )**

95.- Sensor que indica ¿No estoy junto a un trompo?
A) notNextToABeeper( ) **B) frontIsBlocked ( )** **C) facingEast ( )**
D) facingNorth ( ) **E) rigthIsBlocked ( )**

96.- Sensor que indica ¿Estoy en dirección Oeste?
A) facingNorth ( ) **B) leftIsBlocked ( )** **C) facingSouth ( )**
D) facingWest ( ) **E) facingEast ( )**

97.- Sensor que indica ¿Está despejado a mi izquierda?
A) rightIsClear ( ) **B) leftIsBlocked ( )** **C) leftIsClear ( )**
D) facingWest ( ) **E) rightIsBlocked ( )**

98.- Sensor que indica ¿Hay algún trompo en mi mochila de trompos?
A) noBeepersInBeeperBag ( ) **B) nextToABeeper ( )** **C) notNextToABeeper ( )**
D) facingNorth ( ) **E) anyBeepersInBeeperBag ( )**

99.- Sensor que indica ¿No estoy en dirección Norte?
A) rightIsClear ( ) **B) leftIsBlocked ( )** **C) leftIsClear ( )**
D) NotfacingNort ( ) **E) rightIsBlocked ( )**

100.- Sensor que indica ¿Está bloqueado a mi izquierda?
A) frontIsClear ( ) **B) frontIsBlocked ( )** **C) facingEast ( )**
D) rigthIsBlocked ( ) **E) lefthIsBlocked ( )**

101.- Sensor que indica ¿No hay trompo en mi mochila de trompos?
A) frontIsClear ( ) **B) noBeepersInBeeperBag ( )** **C) facingEast ( )**
D) rigthIsBlocked ( ) **E) lefthIsBlocked ( )**

102.- Sensor que indica ¿No estoy en dirección Sur?
A) rightIsClear ( ) **B) leftIsBlocked ( )** **C) leftIsClear ( )**
D) rightIsBlocked ( ) **E) NotfacingSouth ( )**

103.- Sensor que indica ¿Está despejado a mi derecha?
A) leftIsClear ( ) **B) leftIsBlocked ( )** **C) facingSouth ( )**
D) facingWest ( ) **E) rightIsClear ( )**

104.- Sensor que indica ¿Estoy en dirección Norte?
A) rightIsClear ( ) **B) facingNort ( )** **C) leftIsClear ( )**
D) NotfacingNort ( ) **E) rightIsBlocked ( )**

105.- Sensor que indica ¿No estoy en dirección Este?
A) NotfacingEast ( ) **B) facingEast ( )** **C) leftIsClear ( )**
D) rightIsBlocked ( ) **E) NotfacingSouth ( )**

106.- Sensor que indica ¿Está bloqueado a mi derecha?
A) frontIsClear ( ) **B) frontIsBlocked ( )** **C) facingEast ( )**
D) rigthIsBlocked ( ) **E) facingNorth ( )**

107.- Sensor que indica ¿Estoy en dirección Sur?
A) rightIsClear ( ) **B) facingSouth ( )** **C) NotfacingSouth ( )**
D) rightIsBlocked ( ) **E) leftIsBlocked ( )**

108.- Sensor que indica ¿No estoy en dirección Oeste?
A) facingEast ( ) **B) facingSouth ( )** **C) notFacingEast ( )**
D) facingWest ( ) **E) notFacingWest ( )**

ETAPA 3 PROGRAMACIÓN BÁSICA EN ROBOT KAREL

XVII. SUBRAYA LA RESPUESTA CORRECTA

109.- Existe una gran variedad de comandos que se emplean en la programación del Robot Karel, del siguiente listado selecciona aquellos que correspondan al grupo de comandos básicos.

1.- move ( )	2.- turnleft ( )	3.- Program ( )	4.- putbeeper ( )
5.- turnoff ( )	6.- pickbeeper ( )	7.- Class program {	8.- iterate ( )

A) 1,4,5,7,8
B) 2,3,4,7,8
C) 3,4,5,6,7
D) 4,5,6,7,8
E) 1,2,4,5,6

110.- Es la forma correcta en que se escriben las instrucciones, es bien importante porque un carácter extra o de menos como una coma, un punto o un espacio marcará error en el programa al momento de estarlo capturando
A) SINTAXIS **B) SEMÁNTICA** **C) COMANDO**
D) SENTENCIA **E) GRAMÁTICA**

111.- Comando que se emplea para avanzar al Robot Karel una cuadra sobre la calle o avenida en la dirección en que se encuentre orientado
A) PUTBEEPER ( ) **B) TURNOFF ( )** **C) MOVE ( )**
D) TURNLEFT ( ) **E) PICKBEEPER ( )**

112.- Comando que se emplea para hacer girar al Robot Karel 90 grados a su izquierda
A) PICKBEEPER ( ) **B) TURNOFF ( )** **C) PUTBEEPER ( )**
D) TURNLEFT ( ) **E) MOVE ( )**

113.- Comando que se emplea para que el Robot Karel deje un trompo o zumbador en la esquina donde se encuentre parado.
A) TURNOFF ( ) **B) PUTBEEPER ( )** **C) MOVE ( )**
D) PICKBEEPER ( ) **E) TURNLEFT ( )**

114.- ¿Cuántos trompos debe tener Karel en su mochila para poder utilizar la instrucción putbeeper() y no marque error el programa?
A) AL MENOS UNO **B) NINGUNO** **C) DIEZ**
D) CERO **E) AL MENOS DOS**

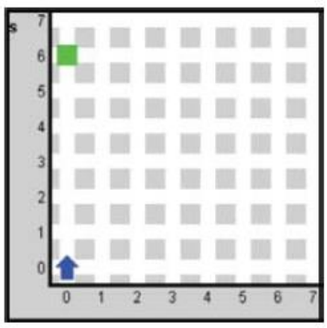
115.- Comando que se emplea para que el Robot Karel levante un trompo de la esquina en donde se encuentre parado.
A) PICKBEEPER ( ) **B) TURNOFF ( )** **C) PUTBEEPER ( )**
D) TURNLEFT ( ) **E) MOVE ( )**

116.- ¿Cuáles son los requisitos para poder utilizar la instrucción pickbeeper ( ) y no marque error el programa?
A) KAREL DEBE ESTAR EN UNA ESQUINA DONDE HAYA AL MENOS UN TROMPO O ZUMBADOR
B) KAREL DEBE ESTAR EN UNA ESQUINA DONDE HAYA AL MENOS DIEZ TROMPOS O ZUMBADORES
C) KAREL DEBE ESTAR EN UNA AVENIDA
D) KAREL DEBE ESTAR EN UNA ESQUINA AL NORTE
E) KAREL DEBE TENER DIEZ ZUMBADORES EN SU MOCHILA

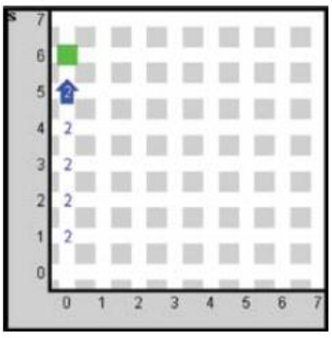
117.- Comando que se emplea para que el Robot Karel deje de realizar cualquier tarea.
A) PUTBEEPER ( ) **B) PICKBEEPER ( )** **C) TURNLEFT ( )**
D) TURNOFF ( ) **E) MOVE ( )**

118.- Completa el siguiente programa con los comandos básicos que hacen falta para que karel cumpla su recorrido

MUNDO INICIAL:



MUNDO FINAL:



PROGRAMA:

```
class program {

  program() {
                    (  )
    putbeeper();
    putbeeper();
    move();
    putbeeper();
                    (  )
    move();
    putbeeper();
    putbeeper();
    move();
    putbeeper();
    putbeeper();
    move();
    putbeeper();
    putbeeper();
                    (  )
  }
}
```

A) turnleft ( );
B) move ( );
C) turnoff ( );
D) pickbeeper ( );
E) putbeeper ( );

ETAPA 4 PROGRAMACIÓN AVANZADA EN ROBOT KAREL

XVIII. SUBRAYA LA RESPUESTA CORRECTA

119.- Comando que se emplea para ordenarle al Robot Karel que repita una o más veces una serie de instrucciones u órdenes escritas entre dos llaves

A) ITERATE (N) **B) WHILE (CONDICION)** **C) MODULOS**
D) IF (CONDICION) **E) IF (CONDICION) / ELSE**

120.- Sentencia de control que se emplea para que el Robot Karel evalúe una condición con ayuda de sus sensores (realiza una tarea específica si se cumple con una condición determinada)

A) MODULOS **B) WHILE (CONDICION)** **C) ITERATE (N)**
D) IF (CONDICION) **E) IF (CONDICION) / ELSE**

121.- Estatuto de control condicional compuesto que se emplea para que el Robot Karel con ayuda de sus sensores realice una tarea, si la condición determinada es verdadera procesa las ordenes X y en caso de ser falsa la condición cumple con las ordenes Y.

A) WHILE (CONDICION) **B) IF (CONDICION) / ELSE** **C) MODULOS**
D) IF (CONDICION) **E) ITERATE (N)**

122.- Sentencia de control que se emplea en los programas del Robot Karel para realizar una tarea, mientras se cumple con una condición determinada.

A) IF (CONDICION) **B) IF (CONDICION) / ELSE** **C) MODULOS**
D) WHILE (CONDICION) **E) ITERATE (N)**

123.- Conjunto de instrucciones u órdenes que se emplean para resolver una tarea simple, concreta y especifica como parte de la solución completa de un programa.

A) ITERATE (N) **B) FUNCIONES ó MODULOS** **C) IF (CONDICION) / ELSE**
D) IF (CONDICION) **E) WHILE (CONDICION)**

124.- Es la palabra que se utiliza en el programa para avisarle al sistema (compilador) que va a definirse una nueva función o módulo.

A) Class **B) Program** **C) Function**
D) void **E) llamar**

125.- Es la sección en el programa donde se define la función o el módulo para avisarle al sistema (compilador) que va a definirse una nueva función o módulo.

A) PROGRAM ( ) { **B) TURNOFF ( )** **C) CLASS PROGRAM {**
D) FUNCION **E) VOID**

126.- Permiten subdividir la solución de un problema en en pequeñas tareas y escribir un módulo para solucionar cada subproblema o tareas previamente identificados.

A) IF (CONDICION) **B) ITERATE (N)** **C) MÓDULOS COMPUESTOS**
D) FUNCIONES O MÓDULOS MULTIPLES **E) PARÁMETROS**

127.- Es colocar una variable dentro del paréntesis con la finalidad de realizar la subtarea tantas veces como valga el valor de la variable dentro del programa. Es el valor numérico constante o nombre de una variable que se escribe entre paréntesis al definir una nueva función invocada para llamar a un módulo o función.

A) FUNCIÓN MULTIMODAL **B) MÓDULOS** **C) PARÁMETRO**
D) VOID **E) CLASS PROGRAM**

128.- Tipo de función que se invoca a si misma desde algún punto de su código dentro de la misma función.

A) FUNCIÓN MUITIFUNCIONAL **B) MÓDULO CONDICIONAL** **C) FUNCIÓN PARAMETRAL**
D) FUNCIÓN VOID **E) FUNCIÓN RECURSIVA**

129.- Función que permite incrementar (le suma uno) en uno el valor del parámetro al momento de llamar a otra función o nuevo módulo.

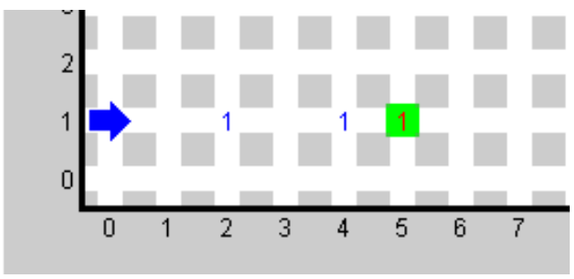
A) SUCC( ) **B) VOID** **C) TURNOFF**
D) PRED( ) **E) ITERATE**

130.- Función que permite decrementar (le resta uno) en uno el valor del parámetro al momento de llamar a otra función o nuevo módulo.

A) VOID **B) SUCC( )** **C) PRED( )**
D) ITERATE **E) TURNOFF**

131.- En el siguiente programa, completa los estatutos y comandos que faltan en el código para que Karel lleve a cabo su tarea de manera correcta.

MUNDO INICIAL



MUNDO FINAL




```
program() {
    (  )      (6) {
    (  )(nextToABeeper()) {
        (  )      ;
    }
    (  )      ;
}
;
turnoff();
}
}
```

A) while
B) if
C) iterate
D) move ( )
E) pickbeeper ( )