**Base de conocimiento TRAMO 1 - Cobol**

**Base de conocimiento**

Tramo 1 COBOL

El siguiente documento esuna ayudaparaquepuedas tener tupropia base de conocimiento resumida delprimer tramo de este curso.

A modo de autoevaluaciónteproponemos vayas completando cada secciónconloquehayas comprendido de cada ítem, te animamos aqueutilices los materiales de consultaque les fuerondados, tambiénpodes buscar en internet opreguntar losprofes.

Que el conocimiento fluya!!!

**Pensamiento lógico**

*El pensamiento lógico no es algo que se pueda enseñar de forma lineal, es algo que se adquiere de manera paulatina a medida que nos enriquecemos de conocimiento y nos desafiamos a resolver cuestiones de los más simples escenarios cotidianos a enigmas complejos. Cuanto más activemos el pensamiento lateral en lo cotidiano vamos a desarrollar un pensamiento lógico creativo, simple y práctico.*

**Responder las siguientes preguntas:**

● Para plantear la solución de un problema utilizando algoritmo se sugiere seguir los siguientes pasos. Escribir para cada bloque lo que considere que debe resolverse en cada uno:

○ Determinar cuál es el objetivo:

cual es la necesidad que se requiere resolver

○ Identificar Entradas y Salidas:

que datos o información tengo y que información tengo que devolver

○ Describir los pasos (instrucciones):

como voy a organizar y procesar la informacion para llegar a la solucion



**Base de conocimiento TRAMO 1 - Cobol**

● Para volcar un planteo de una solución a la programación estructurada, se propone organizar la información de la siguiente manera: INICIO - PROCESO - FIN.

Escribir para cada bloque lo que considere que debe resolverse en cada uno:

○ INICIO:

declarar variables

○ PROCESO:

procesar datos

○ FIN :

muestro resultados

● Condiciones lógicas: Reemplazar cada condicional con un caso de ejemplo para las siguientes premisas:

**A. ejemplo conjunción**

**SI nota > 7 y asistencia >= 70 entonces**

**alumno aprobado**

**Sino**

**alumno desaprobado**

**Fin-si**

**B. ejemplo disyunción**

**SI nota < 7 ó asistencia <= 70 entonces**

**alumno desaprobado**

**Sino**

**alumno aprobado**

**Fin-si**

|  |  |  |  |
| --- | --- | --- | --- |
| Resultado con Y, **conjunción** | | | |
| nota |  | asistencia | **resultado** |
| v | **y** | v | reprobado |
| v | **y** | f | false |
| f | **y** | v | falso |
| f | **y** | f | falso |

|  |  |  |  |
| --- | --- | --- | --- |
| Resultado con O, **disyunción** | | | |
| nota |  | asistencia | **resultado** |
| v | **o** | v | reprobado |
| v | **o** | f | reprobado |
| f | **o** | v | desaprobado |
| f | **o** | f | aprobado |

**Mainframe**

*En su esencia, mainframes son computadoras de alto rendimiento con grandes cantidades de memoria y procesadores que procesan miles de millones de cálculos y transacciones simples en tiempo real. El mainframe es fundamental para las bases*



**Base de conocimiento TRAMO 1 - Cobol**

*de datos comerciales, los servidores de transacciones y las aplicaciones que requieren resiliencia, seguridad y agilidad.*

**Responder las siguientes preguntas:**

● ¿Qué recuerdo sobre el tema mainframe?:

○ ¿Qué sistema operativo utiliza?:

“Z os” - AS400 y usa procesadores redundantes que se hacen cargo de tareas de otros procesadores

○ ¿Qué lenguajes interpreta?:

Cobol, Assembler, RPG, Hexadecimal, PL1,

○ ¿Cómo interactúa en la vida cotidiana con los demás sistemas?:

lenguajes de capas de arquitectura que conversan con los lenguajes de cada capa de presentación, ANDROID, IOS, WINDOWS, ETC

**Cobol**

*Cobol es un lenguaje de programación estructurado modular, principalmente utilizado para desarrollar programas que corran en mainframe para trabajos de lote y procesamiento de transacciones a gran escala. Se utiliza principalmente en sistemas comerciales, financieros y administrativos para empresas y gobiernos.*

**Estructura de un programa cobol:**

● ¿Qué recuerdo de la estructura básica de un programa cobol?:

○ ¿Cuántas divisiones tiene y qué información contiene cada una?:

Tiene 4 divisiones.

**Identification División:** identifica el programa, contiene nombres con el que esta catalogado

**Enviroment División:** información del entorno(versiones con las que trabaja), identificar archivos, servidor, notacion decimal, monedas

**Data División** Contiene toda estrctura de datos de entrada y salida de mi programa

**procedure division** Contiene las instrucciones cobol del programa

TODAS LAS DIVISIONES TIENEN QUE ESTAR ESCRITA EN EL ORDEN

CORRESPONDIENTE PARA QUE MI PROGRAMA COMPILE

tarea

○ ¿Cuántas secciones tiene y qué información contiene cada una?

● ¿Una variable y un dato son lo mismo?

Variable es un lugar de memoria direccionable a trabes de su nombre, y tiene atributos, esos atributos tiene una definición de tipo. En cambio el dato es el contenido que tiene una variable, un valor, no un lugar de memoria



**Base de conocimiento TRAMO 1 - Cobol**

● ¿Cuáles son los tipos de datos que maneja el cobol?

Entero, carácter, booleano

● ¿Con qué objetivo defino una variable de edición?

Se usa formatear,darle forma a un número o carácter sin cambiar su contenido,

● ¿Puedo usar una variable de edición para realizar operaciones

No se puede, hacer operaciones aritmeticas.

Aritméticas?

● ¿En qué sección escribo las instrucciones cobol ?

PROCEDURE DIVISION.

● Para que mi programa compile debe respetar:

○ Es obligatorio el uso del ELSE en la instrucción IF. V o F

No es necesario el uso de else, pero si end-if o .

○ Es obligatorio un punto al final de cada línea. V o F

a veces si a veces no, si esta en un parrafo puede terminarse antes, asi que es necesario identificar donde poner el punto

○ Las instrucciones cobol se escriben a partir de la columna 8.

no a partir de la 12

V o F ○ Los párrafos empiezan y terminan en la línea 8. V o F verdadero el parrafo inicia en la columna 8

○ El comentario se indica con un signo # en la línea 7. V o F

Si, comienza en la columna 7, sino no compila

○ Todas las variables declaradas en la Working Storage deben ser utilizadas dentro de la procedure. V o F

Falso, no es necesario, pero es una mala practica.

● ¿De qué formas puedo utilizar la instrucción PERFORM en un programa cobol?

Principalmente para llamar a un parrafo o recorrer un bucle, pero tambien tiene muchisimás más utilizaciones, consultar al manual de ibm para más detalles.

https://www.ibm.com/docs/es/developer-for-zos/14.1?topic=commands-perform-command-cobol

● ¿De qué maneras puedo resolver una operación de suma utilizando instrucciones cobol?

Con el ADD y con el COMPUTE

