

# Índice

## Sumário

1. PLANO DE DISCIPLINA.....	3
1. DISCIPLINA.....	3
2. EMENTA.....	3
3. OBJETIVO GERAL.....	3
4. OBJETIVOS ESPECÍFICOS.....	3
2. Modulo.....	6
Mainframe.....	7
JES2.....	7
JCL.....	7
Algoritmos.....	7
Módulo 1. INTRODUÇÃO A LINGUAGEM COBOL.....	8
Módulo 2. Movimentando valores em Variáveis/Literais - Inicial.....	8
Importante : Praticas no TSO.....	9
Módulo 3. Explorando Variáveis/Literais – Intermediário.....	9
Módulo 4. Processamento de arquivos sequenciais.....	10
Módulo 5. Calculando, conceitos matemáticos no COBOL.....	10
Módulo 6. Dominando Variáveis/Literais – Avançado.....	11
Módulo 7. Reutilização de código e ganhos de produtividade.....	12
Módulo 8. Recebendo parâmetros do JCL e o uso de SYSINs e PARAMs.....	12
Módulo 9. Loopings de Parágrafos.....	13
Módulo 10. Arquivos Sequenciais Indexados – VSAM.....	14
Módulo 11. Relatórios o Retorno.....	14
Módulo 12. Overview DB2.....	15

Módulo 13. Overview CICS Online.....	15
Bônus.....	15
2. METODOLOGIA.....	16
2.1. METODOLOGIA DE ENSINO.....	16
2.2. METODOLOGIA DE AVALIAÇÃO.....	16
2.3. BIBLIOGRAFIA – BÁSICA.....	16

## **1. PLANO DE DISCIPLINA**

### **1. DISCIPLINA**

Nome da disciplina: **COBOL**

Código da Disciplina: Cobol

Nome do Curso: **Formação COBOL**

Carga Horária: **100 hs**

### **2. EMENTA**

A disciplina tem por finalidade fornecer ao aluno os principais conceitos e funcionalidades da linguagem de programação COBOL na plataforma mainframe IBM.

### **3. OBJETIVO GERAL**

Proporcionar ao aluno um entendimento dos principais conceitos e funcionalidades da linguagem de programação COBOL. O aluno aprenderá sobre a linguagem de programação básica de computadores de grande porte (mainframe) utilizada em sistemas comerciais/negociais mais difundida no mundo, a Common Business Oriented Language - COBOL.

### **4. OBJETIVOS ESPECÍFICOS**

Os objetivos específicos são alcançados a partir do entendimento da linguagem de programação COBOL. Sua estrutura, a elaboração de um programa, codificação (escrita do código), compilação e execução de programas. Palavras reservadas da linguagem, uso de arquivos e manipulação na construção de sistemas de informação. O uso de rotinas de classificação do COBOL, funções de manipulação de tabelas, técnicas de codificação estruturadas e organização sequencial de arquivos.

**Tópicos adicionais incluem:**

- Introdução ao Mainframe
- JES2 – Job Entry Subsystem
- Navegação na IDE TSO;
- Conceitos de Arquivos sequências Fixo Blocado e Particionados;
- Uso de JCL e navegação SDSF;
- Conhecimento do Spool SDSF, consola e submeter JCLs;
- Organização de arquivos sequenciais;
- Compilação, debugging de compilação e testes;
- Funções de manipulação de tabelas internas;
- Técnicas de codificação imperativa e estruturada;
- Noções funcionais e massa de teste;
- Subprograma;
- Recursividade;
- Multithreading;
- Performance e otimização;
- Balanceamento de arquivos (balance-line).
- Noções básicas DB2;
- Navegação de CICS;

**Histórico da Revisão**

<b>Data Publicação</b>	07/06/2023	<b>Autor</b>	Vagner Bellacosa
<b>Data Revisão</b>	25/06/2023	<b>Autor</b>	Vagner Bellacosa
<b>Data Revisão</b>	25/01/2024	<b>Autor</b>	Vagner Bellacosa

# CONTEÚDO PROGRAMÁTICO

## 2. Modulo

Esta disciplina esta programada para ter 25 aulas que serão divididas em duas sessões de duas horas.

Cada sessão contará com parte teórica e prática, onde o aluno poderá navegar no TSO e treinar os conceitos aprendidos na teoria.

Para aprofundar o tema serão fornecidos programas exemplos para o aluno conhecer o funcionamento do comando e poder inovar experimentando novas combinações e funcionalidades.

O material de consulta serão os slides fornecidos antes de cada aula e o fundamento Teórico será o Manual IBM da Linguagem COBOL.

Espero ajudar a compartilhar o conhecimento do COBOL a todos e juntos comemoraremos os 65 anos desta Linguagem de Programação, que é a mãe de todas as linguagens.

Gratidão eterna a Grace Hopper, uma matemática e programadora, que primeiramente abriu o caminho e construiu as trilhas, que evoluíram nas super-rodovias da programação em nossos dias.

Bom curso

Vagner Renato Bellacosa

## **Mainframe**

Introdução ao Ambiente Mainframe, conhecimento técnico sobre o funcionamento de um computador mainframe, conceitos de RACF, Sysplex, LPAR e comunicação entre mainframes e máquinas virtuais.

**Objetivo:** Ao final do módulo, o aluno será capaz de entender o funcionamento de um Computador Mainframe, uso de memória, disco e sistema operacional z/OS.

## **JES2**

Job Entry System 2 é um subsistema de entrada de tarefas, ou seja, um monitor que dialoga com o sistema operacional para receber, executar, planejar e agendar as tarefas de processamento Batch.

**Objetivo:** Ao final do módulo, o aluno será capaz de entender o fluxo de execução de um JOB Batch e as diversas filas dentro do sistema, ler execução de trabalhos e saber se foram concluídos com êxito.

## **JCL**

Job Control Language é uma linguagem de programação utilizada para compilar e executar programas em processo batch.

**Objetivo:** Ao final do módulo, o aluno será capaz de escrever JCL simples de compilação e execução de programas, alocando datasets de Input e Output. Solucionando ABENDs e parametrizando programas.

## **Algoritmos**

Apresentar aos alunos noções sobre lógica de programação, programação procedural e conceito gerais aplicado ao desenvolvimento COBOL.

**Objetivo:** Ao final do modulo, o aluno será capaz de escrever programas em português estruturado, entender fluxogramas e codificar programas em logica procedural.

## **Módulo 1. INTRODUÇÃO A LINGUAGEM COBOL**

Um pouco de história sobre o COBOL e seus quase 50 anos, IBM e os 50 anos do Mainframe, o sistema operacional z/OS, Codashl e a ISO 1989/2022. Conhecer a estrutura de um programa COBOL, suas quatro divisões e entender como funciona o programa Logico e o programa Físico, divisão de trabalhos Batch e Online.

Comandos explorados neste modulo: IDENTIFICATION, ENVIRONMENT, DATA, PROCEDURE, DIVISION, SECTION, EXIT.

**Objetivo:** Ao final do modulo, o aluno será capaz de reconhecer um programa Cobol, ter noções do ambiente e terá navegado na IDE do TSO.

## **Módulo 2. Movimentando valores em Variáveis/Literais - Inicial**

Conhecer os elementos da linguagem e iniciar na criação de literais com seus formatos, entender espaço de memória, mascara de edição, iniciar os experimentos em hierarquia de nível de variáveis e o funcionamento do 77 e 88 Levels.

Na IDE TSO aprenderá a compilar um programa e executar o programa Hello World, com noções de JCL e como submetê-lo em linha de comando.

Comandos explorados neste modulo: PROGRAM-ID, AUTHOR, DATE-WRITTEN, DATE-COMPILER, FILE-SECTION, WORKING-STORAGE, PICTURE, VALUE, MOVE, TO, DISPLAY, GOBACK, STOP RUN, REDEFINES.



**Objetivo:** Ao final do modulo, codificará programas básicos com uso de literais, inicializando variáveis e exibindo no SDSF, distinguindo os diversos tipos e seu comportamento prático, saberá utilizar variáveis lógicas, redefinindo valores e priorizar suas as movimentações de campo a campo, ajudando na performance do programa.

## **Importante : Praticas no TSO**

A partir deste modulo serão fornecidos programas exemplos, para o aluno digitar na IDE TSO, treinar navegação nas Library, o uso dos arquivos-fontes no TSO, compilar, debugar e submetê-los a execução.

## **Módulo 3. Explorando Variáveis/Literais – Intermediário**

Avançando no conceito das variáveis, conheceremos como o Mainframe reconhece os caracteres, através da Tabela EBCDIC, reforçando conceitos de bits e bytes, valores hexadecimal, culminando com o uso de literais COMPUTACIONAL. Exploramos o conceito de relatórios e o uso de máscara de edição, o uso destas variáveis no dia a dia do CPD. Exploramos o conceito de Division, Parágrafos, Sections e navegação do ponteiro de execução, explorando temas como massa de teste e evidência de testes, visualizando a clássica divisão de ambiente processional.

Comandos explorados neste modulo: GO TO, PERFORM, COMPUTACIONAL, USAGE, VALUE, LENGTH OF, INITIALIZE, ACCEPT, FROM, DAY, DATE, TIME, DAY-OF-WEEK, RETURN-CODE, EXIT.

**Objetivo:** Ao final do modulo, o aluno será capaz de trabalhar com variáveis COMP, entender o funcionamento do fluxo do processamento, entender o posicionamento dos campos na tela, saber posicionar o ponteiro do programa em um paragrafo especifico.

## **Módulo 4. Processamento de arquivos sequenciais**

Neste modulo o aluno conhecerá o processamento batch de arquivos sequenciais, entender o fluxo de trabalho, o procedimento para definir arquivos, definir áreas de arquivo, iniciar o processamento lógico, direcionando o fluxo do ponteiro, noções sobre CRUD e Registros.

Reforçando o funcionamento imperativo e procedural do COBOL, utilizando Seções específicas para tratar o arquivo, entender a criação de laços looping, condições de retorno do arquivo, abrindo e recebendo dados previamente armazenados, reforço no uso de desvios condicionais e controle de fluxo.

Comandos explorados neste modulo: CONFIGURATION-SECTION, SOURCE-COMPUTER, OBJECT-COMPUTER, SPECIAL-NAMES, IF, AND, OR, EVALUATE, INPUT-OUTPUT-SECTION, FILE CONTROL, SELECT, ASSIGN, ORGANIZATION, ACCESS MODE, RECORD KEY, OPEN, READ, WRITE, ADD, CLOSE, AT END.

**Objetivo:** Ao final do modulo, o aluno estará preparado a efetuar manutenções simples em processo batch de Logica CRUD, criará arquivos com dados processados, poderá recuperar informações e entender um fluxo batch de processamento encadeado. Usará desvios condicionais e controle de fluxo processional.

## **Módulo 5. Calculando, conceitos matemáticos no COBOL**

Agora que temos o conceito de arquivos, armazenamento de dados, vamos avançar mais um passo, desta vez, trabalhando com o conceito aritmético, atribuindo valores as literais, computando valores, calculando medias, subtraindo resultados.

Estes comandos são muito utilizados em cálculos, além disso verificaremos como criar fórmulas e usar funções intrínsecas para treinar rotinas aritméticas.

Neste modulo completamos os comandos básicos para a atividade de programação, por isso exploraremos alguns conceitos de teste de Software, massa de Teste e suas evidências provando sua conclusão.

Comandos explorados neste modulo : COMPUTE, ADD, SUBTRACT, DIVIDE, MULTIPLY, INITIALIZE, GIVING, ROUNDED, ON SIZE ERROR, END-ADD,. END-SUBTRACT, END-DIVIDE, END-MULTIPLY, CORRESPONDING, SIGN, LEADING, SEPARATE.

**Objetivo:** Ao final do modulo, o aluno estará apto a fazer inúmeros cálculos e com isso implementar programas com grau mediano de complexidade, conhecera como calcular valores numéricos complexos e implementar fórmulas matemáticas e estatísticas no COBOL e será capaz de criar documentos evidenciando estes testes..

## **Módulo 6. Dominando Variáveis/Literais – Avançado**

Estamos atingindo o meio do curso, nesta jornada exploramos vários comandos e sua utilização no COBOL, mas o programador percebera que em algumas situações as variáveis precisaram ter superpoderes e poder acumular dados em estruturas ou coleção de dados, eis que surgem o array unidimensional e em casos mais complexos usamos as Matrizes bi-tri-n direcionais e a cereja em cima do Bolo: Variáveis Alfanuméricas/Alfabéticas com tamanho variável.

Entenderemos a alocação contínua de memória e o uso de índices para navegar e utilizar o conceito de CRUD para acessar estes dados. Criando Tabelas de dados em tempo de execução e noções de pesquisa.

Comandos explorados neste modulo : OCCURS, TIMES, ASCENDING, DESCENDING, KEY, INDEXED BY, WHEN, AT END, DEPENDING ON, SET, UP . SEARCH, TABLE, ALL, WHEN, VARYING, END-SEARCH.

**Objetivo:** Ao final do modulo, o aluno conhecerá usos pratico de matrizes e arrays, incrementando e navegando nas ocorrências, poderá realizar busca dinâmica em Tabela interna e colocará em prática conceitos de fila e armazenamento.

## **Módulo 7. Reutilização de código e ganhos de produtividade.**

Como vimos no exemplo de arquivos sequenciais, os alunos foram obrigados a repetir imensos blocos de códigos, tanto na definição de arquivo, como na definição de variáveis. Visando ganhos de produtividade e evitar retrabalho existem bibliotecas auxiliares chamadas COPYBOOK, que em linhas gerais seriam como Templates.

Vamos explorar este conceito e seu uso na DATA DIVISION e PROCEDURE DIVISION, onde blocos de sentenças poder ser reutilizados na refatoração. Introduzindo novas funções intrínsecas.

Um passo importante e conhecer o conceito de sub-programa e suas formas de chamada, passagem de dados e recebimento de retornos.

Comandos explorados neste modulo : CALL, COPY, REPLACING, CONTINUE, FUNCTION, MAX, MIN, SUM, RANGE, REM, LOG, WHEN-COMPILED, USING, REFERENCE, CONTENT, RETURNING. ON EXCEPTION, NOT ON EXCEPTION, .

**Objetivo:** Ao final do modulo, o aluno deverá entender o conceito de reutilização de código, ganhos de performance, manutenibilidade do fonte. Aprenderá a criar subrotinas externas no COBOL estáticas e dinâmicas, sabendo diferenciar entre ambas e também conhecerá e utilizará algumas funções do COBOL.

## **Módulo 8. Recebendo parâmetros do JCL e o uso de SYSINs e PARAMs**

Num processo Batch em algumas situações surge a necessidade de processar determinados ranges de dados, datas específicas e etc, como podemos parametrizar dinamicamente um programa COBOL? Existem duas maneiras a serem exploradas via PARAM e a via SYSIN, com vantagens e desvantagens.

Ao mesmo tempo descobriremos como alinhar textos nas variáveis alfabéticas e alfanuméricas, arredondar valores calculados, veja o controle de fluxo e as possibilidades resultantes dos parâmetros recebidos via operador nos JCLs, Sysin e Linkage, riscos e vantagens do endereçamento de memória.

Comandos explorados neste modulo : STRING, DELIMITED BY, SIZE, WITH POINTER, ADDRESS OF, CORRESPONDING,, JUSTIFY, LEFT, RIGHT, LINKAGE, CHAR, LOWERCASE, UPPERCASE, ROUND, ON SIZE ERROR.

**Objetivo:** Ao final do modulo, o aluno deverá utilizar JCL para passar parâmetros em tempo de execução ao programa, seja via Param, bem como Sysin. No trabalho de variáveis poderá verificar erros e alinhar textos.

## **Módulo 9. Loopings de Parágrafos.**

Muitas vezes num calculo, numa atualização, numa array/matriz precisamos repetir trechos de código, seja N vezes, seja até atingir um Goal, ou mesmo um contador com limite, vamos reforçar os tipos de instrução no COBOL. Falaremos sobre SECTION, Paragrafo, Frase e Declarações e a organização interna do PROCEDURE DIVISION. Ao mesmo tempo, iniciar o tema sobre otimização e performance, explorando o conceito inline, outline, recursividade e multithreading.

Reforço do conteúdo de laços PERFORM, simples, simulando loops de contador, de variável lógica, com teste a entrada e testes a saída e repetindo N vezes, de modo a concluir etapas de processamento.

Comandos explorados neste modulo : RECURSIVE, PERFORM, THRU, GO TO, TIME, UNTIL, VARYING, WITH TEST, FROM, AFTER, BEFORE, BY, END-PERFORM.

**Objetivo:** Ao final do modulo, o aluno deverá conhecer o processamento de looping, podendo utilizar controles de fluxo e contadores de

acesso, acessando parágrafos e seções específicas do programa, conceito usado desde em cálculos matemáticos, como leitura e atualização de tabelas e Bancos de Dados. Entender conceitos avançados de alto nível na linguagem.

## **Módulo 10. Arquivos Sequenciais Indexados – VSAM**

Conhecer o conceito de arquivo indexado, acessando registros por chave, efetuando CRUD, entendendo o processo de criação de índice e processando lotes de informação, recapitulando os conceitos de VSAM e sua funcionalidade no ambiente Mainframe.

Comandos explorados neste modulo : RECORD KEY, INVALID KEY, REWRITE. DELETE.

**Objetivo:** Ao final do modulo, o aluno poderá efetuar manutenções em programas VSAM, com total controle sobre o arquivo e seus registros, saberá posicionar no índice e obter as informações desta chave primaria.

## **Módulo 11. Relatórios o Retorno**

Voltamos ao início, recapitulando o conceito de relatórios e desta vez, criando estruturas de output mais elaboradas onde o programador, poderá exercitar todo o conhecimento adquirido no decorrer do curso, reforçando o conceito de Tabelas Interna faremos pesquisa por índice e campo.

Comandos explorados neste modulo : SEARCH, EXHBIT, INSPECT, CONVERTING, UNSTRING, TRIM, .

**Objetivo:** Ao final do modulo, o aluno poderá será capaz de atender qualquer solicitação de relatório, saberá fazer pesquisa em Tabela Interna e poderá implementar rotinas mais sofisticadas em COBOL.

## **Módulo 12. Overview DB2**

Neste tópico será apresentado um programa COBOL com todos os comandos básicos de SQL DB2, conhecerá o conceito de Cursor e navegação sequencial de dados externo, conhecerá o SQLCA.

Comandos explorados neste modulo : EXEC, END-EXEC .

**Objetivo:** Ao final do modulo, o aluno poderá será capaz de interpretar um programa COBOL com acesso ao DB2, tendo know-how para trabalhar em manutenções iniciais.

## **Módulo 13. Overview CICS Online**

Neste tópico será apresentado um programa COBOL com todos os comandos básicos de CICS, conhecerá o conceito de navegação de telas BMS, comandos CICS Internos e fluxo de programas online.

**Objetivo:** Ao final do modulo, o aluno poderá será capaz de interpretar um programa COBOL com CICS tendo know-how para trabalhar em manutenções iniciais.

## **Bônus**

Todos os módulos serão acompanhados de programas exemplos para o aluno compilar e testar no SAND-BOX Inefe, um ambiente Mainframe funcional com todos os acessos necessários para Editar, Compilar, Corrigir, Debugar e Criar evidencias de testes.

## **2. METODOLOGIA**

### **2.1. METODOLOGIA DE ENSINO**

Exposição de aulas teóricas, laboratórios e exercícios práticos no ambiente Mainframe.

### **2.2. METODOLOGIA DE AVALIAÇÃO**

- Participação nas aulas (assiduidade, pontualidade e participação no desenvolvimento e entrega de exercícios)
- Atividades escritas:
- Laboratórios: elaboração, compilação e execução de programas na linguagem COBOL
- Avaliações escritas:

### **2.3. BIBLIOGRAFIA – BÁSICA**

1. IBM Enterprise COBOL for z/OS 6.2.0 Language Reference -  
[https://www.ibm.com/support/knowledgecenter/SS6SG3\\_6.2.0/com.ibm.cobol62.ent.doc/PgandLR/lrmvspdf.html](https://www.ibm.com/support/knowledgecenter/SS6SG3_6.2.0/com.ibm.cobol62.ent.doc/PgandLR/lrmvspdf.html)
2. IBM Enterprise COBOL for z/OS 6.2.0 Programming Guide -  
[https://www.ibm.com/support/knowledgecenter/SS6SG3\\_6.2.0/com.ibm.cobol62.ent.doc/PgandLR/pgmvspdf.html](https://www.ibm.com/support/knowledgecenter/SS6SG3_6.2.0/com.ibm.cobol62.ent.doc/PgandLR/pgmvspdf.html)
3. Material de Apoio fornecido em aula
4. Programas exemplos