

Unidade de Ensino Médio e Técnico - CETEC

Plano de Trabalho Docente - 2019

FORMAÇÃO PROFISSIONAL

Ensino Técnico Integrado ao Médio

Plano de Curso no. 361 aprovado pela Portaria do Coordenador do Ensino Médio e Técnico 1567, de 6-11-2018, publicada no Diário Oficial de 7-11-2018 - Poder Executivo - Seção I - página 75.

ETEC:	Escola Técnica Estadual Rodrigues de Abreu		
Código:	135	Município:	Bauru
Eixo Tecnológico	Informação e Comunicação		
Habilitação Profissional:	Habilitação Profissional de TÉCNICO EM DESENVOLVIMENTO DE SISTEMAS INTEGRADO AO ENSINO MÉDIO		
Qualificação:	Qualificação Profissional Técnica de Nível Médio de AUXILIAR EM DESENVOLVIMENTO DE SISTEMAS		
Componente Curricular:	Técnicas de Programação de Algoritmos		
Módulo:	1	C. H. Semanal:	3,00
Professor:	ROBERTA RIBEIRO SOARES MOURA PADOAN ;		

I – Atribuições e atividades profissionais relativas à qualificação ou à habilitação profissional, que justificam o desenvolvimento das competências previstas nesse componente curricular.

Verificar usabilidade no desenvolvimento de programas.
 Realizar versionamento no desenvolvimento de programas.
 Desenvolver programas de computador, utilizando princípios de boas práticas.
 Implementar algoritmos em linguagem de programação, utilizando ambientes de desenvolvimento de acordo com as necessidades

II – Competências, Habilidades e Bases Tecnológicas do Componente Curricular

Competências

1. Implementar algoritmos de programação.
2. Desenvolver sistemas aplicando princípios e paradigmas de programação.

Habilidades

- 1.1 Elaborar algoritmos.
- 2.1 Codificar programas, utilizando técnica de programação estruturada.
- 2.2 Depurar e versionar programas, utilizando ambiente de desenvolvimento integrado.

Bases Tecnológicas

1. Conceitos de Lógica de Programação e construção de algoritmos.
2. Princípios de programação:
 - 2.1 Paradigmas de programação;
 - 2.2 Conceitos de usabilidade de sistemas;
 - 2.3 Linguagens de programação e códigos fonte, objeto e arquivo executável.
3. Comandos da linguagem de programação:
 - 3.1 Funções pré-definidas;
 - 3.2 Expressões e tabela da verdade;
 - 3.3 Tratamento de erros e exceções;
 - 3.4 Memória, tipos de dados e variáveis;
 - 3.5 Entrada, saída e conversão de tipos;
 - 3.6 Operadores aritméticos, relacionais e lógicos.
4. Programação estruturada:
 - 4.1 Laços;
 - 4.2 Iteração;
 - 4.3 Teste de mesa;
 - 4.4 Decisão simples;
 - 4.5 Decisão múltipla.
5. Ferramentas para o desenvolvimento:
 - 5.1 Ambiente integrado de desenvolvimento (IDE);
 - 5.2 Editor de código:
 - 5.2.1 navegação;
 - 5.2.2 completar comandos;
 - 5.2.3 coloração de sintaxe;
 - 5.2.4 marcas de erro.
 - 5.3 Compilação, empacotamento e distribuição (build and deploy);
 - 5.4 Bibliotecas, frameworks e gestão de dependências;
 - 5.5 Modularização e organização em projetos de programas e sistemas.
6. Verificação e depuração de código:
 - 6.1 Execução passo a passo;
 - 6.2 Criação de pontos de interrupção (breakpoints);
 - 6.3 Visualização de valores de variáveis em tempo de execução;
 - 6.4 Pilha de chamadas (call stack);
 - 6.5 Interpretação de informações detalhadas sobre exceções.
7. Programação modular:

- 7.1 Sub-rotinas;
- 7.2 Recursividade;
- 7.3 Procedimentos e funções;
- 7.4 Argumentos e escopo de identificadores.
- 8. Tipos de dados estruturados:
 - 8.1 Vetores;
 - 8.2 Matrizes;
 - 8.3 Arquivos binários e de texto.
- 9. Versionamento e colaboração:
 - 9.1 Conceitos de controle de versão e gestão de código fonte;
 - 9.2 Software livre e colaboração com repositórios remotos; 9.6 Ramificação (branch), comparação (diff) e mesclagem (merge).
 - 9.3 Criação de repositórios locais e remotos;
 - 9.4 Envio (commit) e resgate de versões, checkin e checkout;
 - 9.5 Controle de usuários para o desenvolvimento colaborativo;
 - 9.6 Ramificação (branch), comparação (diff) e mesclagem (merge).
- 10. Práticas de programação:
 - 10.1 Estilo de codificação, indentação, legibilidade, comentários;
 - 10.2 Refatoração;
 - 10.3 Programação em par;
 - 10.4 Testes unitários.

III – Procedimento Didático e Cronograma de Desenvolvimento

Habilidades	Bases Tecnológicas	Bases Científicas	Procedimentos Didáticos	De	Até
1.1 Elaborar algoritmos.;	1. Conceitos de Lógica de Programação e construção de algoritmos. ;	Matemática: análise de dados - tipos de amostragem	Apresentar bases tecnológicas, habilidades e competências do componente curricular	04/02/19	08/02/19
1.1 Elaborar algoritmos.;	1. Conceitos de Lógica de Programação e construção de algoritmos. ;	Matemática: análise de dados - tipos de amostragem	Apresentar exercícios de lógica do dia a dia (fazer, bolo, trocar pneu de carro, lógica matemática)	11/02/19	15/02/19
1.1 Elaborar algoritmos.;	1. Conceitos de Lógica de Programação e construção de algoritmos. ;	Matemática: análise de dados - tipos de amostragem	Aplicar avaliação diagnóstica para verificar conhecimento prévio. Definição de declaração de variáveis	18/02/19	01/03/19
1.1 Elaborar algoritmos.;	1. Conceitos de Lógica de Programação e construção de algoritmos. ; 2. Princípios de programação ;	matemática	Apresentar as ferramentas de fluxograma, sua aplicação e uso no aplicativo Dia	14/03/19	27/03/19
1.1 Elaborar algoritmos.; 2.1 Codificar programas, utilizando técnica de programação estruturada.;	2. Princípios de programação; ; 2.1 Paradigmas de programação; ; 2.2 Conceitos de usabilidade de sistemas; ;	matemática: conjuntos numéricos	Declarar e definir variáveis e aplicar em pseudocódigos usando Portugol Studio.	01/04/19	12/04/19
2.1 Codificar programas, utilizando técnica de programação estruturada.;	1. Conceitos de Lógica de Programação e construção de algoritmos. ; 2. Princípios de programação; ; 2.1 Paradigmas de programação; ; 2.2 Conceitos de usabilidade de sistemas; ; 3. Comandos da linguagem de programação; ; 3.4 Memória, tipos de dados e variáveis; ; 3.5 Entrada, saída e conversão de tipos; ;	matemática: análise de dados, trabalhar com equivalência e quantificadores.	Comandos de impressão e comandos de atribuição, com exercícios para aplicação. Aplicação das definições de pseudocódigos, fluxogramas e declaração e uso de variáveis em Análise de Sistemas, no item relacionado a Especificação de requisitos.	15/04/19	30/04/19
1.1 Elaborar algoritmos.; 2.1 Codificar programas, utilizando técnica de programação estruturada.;	1. Conceitos de Lógica de Programação e construção de algoritmos. ; 2. Princípios de programação; ; 2.1 Paradigmas de programação; ; 2.2 Conceitos de usabilidade de sistemas; ; 2.3 Linguagens de programação e códigos fonte, objeto e arquivo executável. ; 3. Comandos da linguagem de programação; ; 3.5 Entrada, saída e conversão de tipos; ;	matemática: análise de dados, trabalhar com equivalência e quantificadores.	Comandos de leitura. Exercícios de fixação usando todos os comandos aprendidos.	06/05/19	17/05/19
1.1 Elaborar algoritmos.; 2.1 Codificar programas, utilizando técnica de programação estruturada.;	3. Comandos da linguagem de programação; ; 3.5 Entrada, saída e conversão de tipos; ;	matemática: análise de dados, trabalhar com equivalência, quantificadores e conectivos.	Comando de impressão. Exemplos de uso. Ferramentas para fluxograma.	20/05/19	31/05/19
1.1 Elaborar algoritmos.; 2.1 Codificar programas, utilizando técnica de programação estruturada.;	3. Comandos da linguagem de programação; ; 3.4 Memória, tipos de dados e variáveis; ; 3.5 Entrada, saída e conversão de tipos; ; 4. Programação estruturada; ;	matemática	Exercícios de aplicação, usando declaração, atribuição, comandos de leitura e escrita. Introdução a linguagem C/C++	03/06/19	14/06/19
1.1 Elaborar algoritmos.; 2.1 Codificar programas, utilizando técnica de programação estruturada.;	3. Comandos da linguagem de programação; ; 3.1 Funções pré-definidas; ; 3.2 Expressões e tabela da verdade; ; 3.3 Tratamento de erros e exceções; ; 3.4 Memória, tipos de dados e variáveis; ; 3.5 Entrada, saída e conversão de tipos; ; 3.6 Operadores aritméticos, relacionais e lógicos. ;	matemática: sistemas lineares e análise de dados (trabalhar com equivalência e quantificadores e conectivos).	Operadores aritméticos. Exemplos de uso. Exercícios de fixação com aplicação em fluxogramas e pseudocódigos.	17/06/19	28/06/19
1.1 Elaborar algoritmos.; 2.1 Codificar programas, utilizando técnica de programação estruturada.;	3. Comandos da linguagem de programação; ; 3.1 Funções pré-definidas; ; 3.2 Expressões e tabela da verdade; ; 3.3 Tratamento de erros e exceções; ; 3.4 Memória, tipos de dados e variáveis; ; 3.5 Entrada, saída e conversão de tipos; ; 3.6 Operadores aritméticos, relacionais e lógicos. ;	matemática: sistemas lineares e análise de dados (trabalhar com equivalência e quantificadores e conectivos).	Exercícios de fixação para aplicação de comandos aritméticos e lógicos. Utilização de funções pré definidas.	01/07/19	03/07/19

1.1 Elaborar algoritmos.; 2.1 Codificar programas, utilizando técnica de programação estruturada.;	3. Comandos da linguagem de programação: ; 3.2 Expressões e tabela da verdade; ; 3.3 Tratamento de erros e exceções; ; 3.4 Memória, tipos de dados e variáveis; ; 3.5 Entrada, saída e conversão de tipos; ; 3.6 Operadores aritméticos, relacionais e lógicos. ; 4. Programação estruturada; ;	matemática: sistemas lineares e análise de dados (trabalhar com negação, equivalência, quantificadores e conectivos).	Operadores relacionais. Aplicar lista de exercícios para fixação do conteúdo estudado até o momento. Com aplicação em fluxogramas e pseudocódigo.	22/07/19	02/08/19
1.1 Elaborar algoritmos.; 2.1 Codificar programas, utilizando técnica de programação estruturada.; 2.2 Depurar e versionar programas, utilizando ambiente de desenvolvimento integrado.;	3. Comandos da linguagem de programação: ; 3.1 Funções pré-definidas; ; 3.2 Expressões e tabela da verdade; ; 3.3 Tratamento de erros e exceções; ; 3.4 Memória, tipos de dados e variáveis; ; 3.5 Entrada, saída e conversão de tipos; ; 3.6 Operadores aritméticos, relacionais e lógicos. ; 4. Programação estruturada: ; 4.3 Teste de mesa; ; 4.4 Decisão simples; ; 4.5 Decisão múltipla. ;	matemática: conjuntos numéricos, análise de dados (trabalhar com equivalência e quantificadores e conectivos).	Estrutura de controle condicional. Correção de exercícios da lista, aplicáveis em matemática usando noções de conjuntos.	05/08/19	16/08/19
1.1 Elaborar algoritmos.; 2.1 Codificar programas, utilizando técnica de programação estruturada.; 2.2 Depurar e versionar programas, utilizando ambiente de desenvolvimento integrado.;	3. Comandos da linguagem de programação: ; 3.1 Funções pré-definidas; ; 3.2 Expressões e tabela da verdade; ; 3.3 Tratamento de erros e exceções; ; 3.4 Memória, tipos de dados e variáveis; ; 3.5 Entrada, saída e conversão de tipos; ; 3.6 Operadores aritméticos, relacionais e lógicos. ; 4. Programação estruturada: ; 4.3 Teste de mesa; ; 4.4 Decisão simples; ; 4.5 Decisão múltipla. ;	matemática: conjuntos numéricos, análise de dados (trabalhar com equivalência, negação, quantificadores e conectivos).	Trabalho para revisão do conteúdo e verificação de assimilação e possíveis dúvidas.	19/08/19	23/08/19
1.1 Elaborar algoritmos.; 2.1 Codificar programas, utilizando técnica de programação estruturada.; 2.2 Depurar e versionar programas, utilizando ambiente de desenvolvimento integrado.;	3. Comandos da linguagem de programação: ; 3.2 Expressões e tabela da verdade; ; 3.4 Memória, tipos de dados e variáveis; ; 3.5 Entrada, saída e conversão de tipos; ; 3.6 Operadores aritméticos, relacionais e lógicos. ; 4. Programação estruturada: ; 4.1 Laços; ; 4.2 Iteração; ; 4.3 Teste de mesa; ; 4.4 Decisão simples; ; 4.5 Decisão múltipla. ;	matemática: noções de conjunto, análise de dados (trabalhar com equivalência, negação e quantificadores e conectivos).	Operadores lógicos. Exemplos para fixação. Aplicar exercícios relacionados a comandos lógicos (e, ou e nao), aplicados a conjuntos no uso da matemática (atividade interdisciplinar).	26/08/19	06/09/19
1.1 Elaborar algoritmos.; 2.1 Codificar programas, utilizando técnica de programação estruturada.; 2.2 Depurar e versionar programas, utilizando ambiente de desenvolvimento integrado.;	3. Comandos da linguagem de programação: ; 3.1 Funções pré-definidas; ; 3.2 Expressões e tabela da verdade; ; 3.3 Tratamento de erros e exceções; ; 3.4 Memória, tipos de dados e variáveis; ; 3.5 Entrada, saída e conversão de tipos; ; 3.6 Operadores aritméticos, relacionais e lógicos. ; 4. Programação estruturada: ; 4.1 Laços; ; 4.2 Iteração; ; 4.3 Teste de mesa; ; 4.4 Decisão simples; ; 4.5 Decisão múltipla. ;	matemática	Lista de exercícios com aplicação de operadores relacionais e lógicos.	09/09/19	20/09/19
1.1 Elaborar algoritmos.; 2.1 Codificar programas, utilizando técnica de programação estruturada.; 2.2 Depurar e versionar programas, utilizando ambiente de desenvolvimento integrado.;	3. Comandos da linguagem de programação: ; 3.1 Funções pré-definidas; ; 3.2 Expressões e tabela da verdade; ; 3.3 Tratamento de erros e exceções; ; 3.4 Memória, tipos de dados e variáveis; ; 3.5 Entrada, saída e conversão de tipos; ; 3.6 Operadores aritméticos, relacionais e lógicos. ; 4. Programação estruturada: ; 4.1 Laços; ; 4.2 Iteração; ; 4.3 Teste de mesa; ; 4.4 Decisão simples; ; 4.5 Decisão múltipla. ;	matemática: sistemas lineares, conjuntos e séries estatísticas	Estrutura de repetição. Aplicados em exercícios usando pseudocódigos e linguagem C/C++.	23/09/19	04/10/19
1.1 Elaborar algoritmos.; 2.1 Codificar programas, utilizando técnica de programação estruturada.; 2.2 Depurar e versionar programas, utilizando ambiente de desenvolvimento integrado.;	3.5 Entrada, saída e conversão de tipos; ; 3.6 Operadores aritméticos, relacionais e lógicos. ; 4. Programação estruturada: ; 4.1 Laços; ; 4.2 Iteração; ; 4.3 Teste de mesa; ; 4.4 Decisão simples; ; 4.5 Decisão múltipla. ;	matemática: sistemas lineares, conjuntos numéricos e séries estatísticas (trabalhar com tabela verdade, conectivos, equivalência, quantificadores, negação) e tipos de amostragem.	Aplicar em algoritmos e linguagem C/C++, exercícios realizados como estudo de caso	07/10/19	18/10/19
1.1 Elaborar algoritmos.; 2.1 Codificar programas, utilizando técnica de programação estruturada.; 2.2 Depurar e versionar programas, utilizando ambiente de desenvolvimento integrado.;	3. Comandos da linguagem de programação: ; 3.4 Memória, tipos de dados e variáveis; ; 3.5 Entrada, saída e conversão de tipos; ; 3.6 Operadores aritméticos, relacionais e lógicos. ; 4. Programação estruturada: ; 4.1 Laços; ; 4.2 Iteração; ; 4.3 Teste de mesa; ; 4.4 Decisão simples; ; 4.5 Decisão múltipla. ;	matemática: noções de função, sistemas lineares, matrizes e determinantes.	Apresentar a definição e uso de vetores. Exercícios de fixação e compreensão. Apresentar exercícios de lógica com aplicação matemática para uso de funções (atividade interdisciplinar). Semana da casa aberta	21/10/19	01/11/19
1.1 Elaborar algoritmos.; 2.1 Codificar programas, utilizando técnica de programação estruturada.; 2.2 Depurar e versionar programas, utilizando ambiente de	3. Comandos da linguagem de programação: ; 3.4 Memória, tipos de dados e variáveis; ; 3.5 Entrada, saída e conversão de tipos; ; 3.6 Operadores aritméticos, relacionais e lógicos. ; 4. Programação estruturada: ; 4.1 Laços; ; 4.2 Iteração; ; 4.3 Teste de mesa; ; 4.4 Decisão simples; ; 4.5 Decisão múltipla. ;	matemática: noções de função, sistemas lineares, matrizes e determinantes.	Aplicar os conceitos de vetores em exercícios usando algoritmos e linguagem C/C++. Apresentar o uso de matriz.	04/11/19	14/11/19

desenvolvimento integrado.;					
1.1 Elaborar algoritmos.; 2.1 Codificar programas, utilizando técnica de programação estruturada.; 2.2 Depurar e versionar programas, utilizando ambiente de desenvolvimento integrado.;	3. Comandos da linguagem de programação: ; 3.5 Entrada, saída e conversão de tipos; ; 3.6 Operadores aritméticos, relacionais e lógicos. ; 4. Programação estruturada: ; 4.1 Laços; ; 4.2 Iteração; ; 4.3 Teste de mesa; ; 4.4 Decisão simples; ; 4.5 Decisão múltipla. ;	matemática: noções de função, sistemas lineares, matrizes e determinantes.	Aplicar exercícios de fixação e verificação de dúvidas quanto ao uso de vetores e matriz.	18/11/19	29/11/19
1.1 Elaborar algoritmos.; 2.1 Codificar programas, utilizando técnica de programação estruturada.; 2.2 Depurar e versionar programas, utilizando ambiente de desenvolvimento integrado.;	3.5 Entrada, saída e conversão de tipos; ; 3.6 Operadores aritméticos, relacionais e lógicos. ; 4. Programação estruturada: ; 4.1 Laços; ; 4.2 Iteração; ; 4.3 Teste de mesa; ; 4.4 Decisão simples; ; 4.5 Decisão múltipla. ;	matemática: vetores e matriz	Resolução dos exercícios com uso de vetores e matriz.	02/12/19	13/12/19

IV - Plano de Avaliação de Competências

Competências	Instrumento(s) e Procedimentos de Avaliação	Críticos de Desempenho	Evidências de Desempenho
1. Implementar algoritmos de programação.	Avaliação Escrita ; Avaliação Prática ; Estudo de Caso ; Lista de Exercícios ; Observação Direta ; Participação em Aula ; Recuperação ;	Criatividade na Resolução de Problemas ; Cumprimento das Tarefas Individuais ; Execução do Produto ; Interatividade, Cooperação e Colaboração ; Objetividade ; Organização ; Pertinência das Informações ; Pontualidade e Cumprimento de Prazos ; Postura Adequada, Ética e Cidadã ; Relacionamento de Conceitos ; Relacionamento de Ideias ;	Organização das ideias desenvolvidas focando o raciocínio lógico, a aplicação dos conceitos estudados e a precisão das informações conforme descrito, usando recursos do Portugol Studio. Participação, organização cooperação e interatividade nas aulas.
2. Desenvolver sistemas aplicando princípios e paradigmas de programação.	Avaliação Escrita ; Avaliação Prática ; Avaliação Prática ; Estudo de Caso ; Lista de Exercícios ; Observação Direta ; Participação em Aula ; Recuperação ;	Criatividade na Resolução de Problemas ; Cumprimento das Tarefas Individuais ; Execução do Produto ; Interatividade, Cooperação e Colaboração ; Objetividade ; Interlocução: Ouvir e Ser Ouvido ; Organização ; Pertinência das Informações ; Pontualidade e Cumprimento de Prazos ; Postura Adequada, Ética e Cidadã ; Relacionamento de Conceitos ; Relacionamento de Ideias ;	Organização das ideias desenvolvidas focando o raciocínio lógico, a aplicação dos conceitos estudados usando linguagem de programação C/C++

V – Plano de atividades docentes

Atividade Previstas	Projetos e Ações voltados à redução da Evasão Escolar	Atendimento a alunos por meio de ações e/ou projetos voltados à superação de defasagens de aprendizado ou em processo de Progressão Parcial	Preparo e correção de avaliações	Preparo de material didático	Participação em reuniões com Coordenador de Curso e/ou previstas em Calendário Escolar
Fevereiro	Preparar atividades para que os alunos se integrem aos colegas e a escola (atividades de integração) e aplicação de avaliação diagnóstica				Reunião de planejamento
Março	Verificar ausência continuada e saber os motivos e apresentar a coordenação, também apresentar relatório sobre avaliação diagnóstica	Preparar exercícios de fixação para conteúdos ensinados	Preparar atividades voltadas ao desempenho da sala, como observação direta	Material com conteúdo resumido para facilitar a aplicação dos conceitos aprendidos pelos alunos	Reunião de cursos. Reunião pedagógica
Abril	Apresentar possíveis casos de desinteresse dos alunos a coordenação para que seja avaliada possíveis ações.		Preparar e corrigir avaliações para verificação do conteúdo assimilado, usando recurso de avaliação teórica	Trabalhar listas de exercícios que contemplem todo conteúdo estudado até o momento.	Conselho de classe intermediário
Maio	Semana Paulo Freire	Trabalhar em grupos e desenvolver projetos para promover a integração dos alunos	Preparar atividades voltadas a aplicação do projeto interdisciplinar		Reunião de cursos. reunião de pais. Reunião pedagógica
Junho	Arraiá da Etec	Recuperação contínua de dificuldades	Preparar e corrigir avaliações para verificação do conteúdo assimilado, usando recurso de avaliação teórica	Incluir novos comandos ao resumo apresentado	
Julho					Conselho Intermediário
Agosto		Listas de exercícios extra para alunos com dificuldades	Listas de exercícios extra para alunos com dificuldades		Reunião pedagógica
Setembro	Preparação de projetos para casa aberta	Atendimento aos projetos da semana da casa aberta	Atendimento aos projetos da semana da casa aberta	Incluir novos comandos ao resumo apresentado e material para semana da casa aberta e semana da ciência e tecnologia	reunião de cursos. Reunião de pais.
Outubro	Semana da casa aberta			Exercícios de fixação e dúvidas	Reunião de cursos. Conselho de classe intermediário
Novembro		atendimento as dúvidas relacionadas ao conteúdo apresentado	atendimento as dúvidas relacionadas ao conteúdo apresentado		Reunião pedagógica

Dezembro

VI – Material de Apoio Didático para Aluno (inclusive bibliografia)

OLIVEIRA, Jayr Figueiredo. MANZNO, José Augusto N.G. Algoritmos in Lógica para desenvolvimento de programação computadores> São Paulo. Erica, 2014.

OLIVEIRA, Jayr Figueiredo. MANZNO, José Augusto N.G. Estudo dirigido de algoritmos. São Paulo. Erica, 2011.

PIVA, Dilermano Jr, ENGELBRE, Angela de M. NAKAMITI, Gilberto S. et. al. Algoritmos e Programação de Computadores. Campus, 2012.

SOUZA, Marco A. F. S. GOMES, Marcelo M. SOARES, Marcio V. CONCILIO, Ricardo in. Algoritmos e Lógica de Programação. Cengage Learning, 2012.

Ferramenta Portugal Studio

MIZARAH, Victorine Viviane – Treinamentos em Linguagem C

Material de apoio desenvolvido pela professora do componente curricular

VII – Propostas de Integração e/ou Interdisciplinares e/ou Atividades Extra

Atividade Extra

Disponibilizar exercícios para aplicar conteúdo desenvolvido até momento.

Propostas de Integração e/ou Interdisciplinares

Haverá no semestre a aplicação de atividades interdisciplinares, onde será trabalhado em Técnicas de Programação e Algoritmos os conceitos e utilização de pseudocódigos e definição de variáveis e constantes e operadores aritméticos, relacionais e lógicos serão reforçados em Análise e Projeto de Sistemas aplicados em especificação de requisitos. Também aplicáveis em Técnicas de Programação para sustentar a base de Linguagens de programação e códigos fonte, objeto e executável e Paradigmas de programação

VIII – Estratégias de Recuperação Contínua (para alunos com baixo rendimento/dificuldades de aprendizagem)

Ao identificar o aluno com deficiências ou baixo desempenho na disciplina, será aplicado ao aluno uma revisão do conteúdo duvidoso e em seguida uma lista de exercícios e uma avaliação escrita posterior para verificação da assimilação.

IX – Identificação:

Nome do Professor ROBERTA RIBEIRO SOARES MOURA PADOAN ;

Assinatura

Data 27/02/2019

X – Parecer do Coordenador de Curso:

O Plano de Trabalho Docente, está compatível com o Plano de Curso, apresentando Competências, Habilidade e Bases Tecnológicas, contemplando também as atividades previstas, material de apoio, atividades de integração, interdisciplinaridade, estratégias de recuperação e o plano didático quinzenal.

Nome do Coordenador:

Assinatura:

Data: 03/03/19

Data e ciência do Coordenador Pedagógico

XI - Replanejamento

Data

Descrição

27/02/2019

O conteúdo não aplicado no planejamento do dia será ajustado ao planejamento das aulas subsequentes, resumindo o conteúdo para facilitar a assimilação dos alunos, com aplicação de exercícios extra.

Imprimir