

UNIVERSIDADE PAULISTA - UNIP

LEONARDO CARCELIM FERNANDES - T1689B4

LEONARDO DA SILVA A DE SOUZA - N112042

RICHARD WILLIAN RIBEIRO DIVINO - D074873

THOMAS RICHARD COSTA - N799AF7

CONTROLE DE TRATAMENTO DE EFLUENTES DO SETOR METALÚRGICO

ARARAQUARA

2016

## **RESUMO**

O presente trabalho tem como objetivo principal a apresentação de um software para fazer o controle dos produtos tratados pelas empresas, facilitando assim a forma de registro dessas informações, também apresenta e explica etapas da programação até o resultado final que é a sua utilização. Nele também se encontra informações a respeito de tratamento de efluentes na área geral, porém busca se aprofundar na área metalúrgica por ser a área escolhida pelo grupo, e tem o intuito de fazer parte da mudança de mentalidade da população principalmente brasileira, para com os exercícios de desenvolvimento sustentável.

Palavras-Chave: Software. Controle. Desenvolvimento Sustentável.

## SUMÁRIO

1. INTRODUÇÃO .....	4
2. RESÍDUOS INDUSTRIAIS E URBANOS .....	5
2.1. CONCEITO DE RESÍDUO .....	5
2.2. RESÍDUOS SÓLIDOS URBANOS .....	5
2.3. ATERROS SANITÁRIOS .....	5
2.4. SUBSTÂNCIAS LIXIVIADAS .....	5
2.5. COLETA SELETIVA E RECICLAGEM .....	6
3. TIPOS DE RESÍDUOS QUÍMICOS E EFLUENTES NO SETOR ESCOLHIDO .....	6
4. CONDIÇÕES ATUAIS .....	7
4.1. MEDIDAS PARA O DESENVOLVIMENTO SUSTENTÁVEL .....	8
4.1.1. INCINERAÇÃO .....	8
4.1.2. ATERRO INDUSTRIAL .....	10
4.1.2.1. LOCALIZAÇÃO DE ATERROS DE RESÍDUOS INDUSTRIAIS PERIGOSOS ....	10
4.1.2.2. IMPERMEABILIZAÇÃO INFERIOR .....	11
4.1.2.3. IMPERMEABILIZAÇÃO SUPERIOR (COBERTURA FINAL) .....	12
4.1.3. RECICLAGEM DE RESÍDUOS SÓLIDOS .....	13
4.1.4. PADRONIZAÇÃO DE RECIPIENTES DE MATERIAIS RECICLÁVEIS .....	13
4.1.5. OUTROS PROCESSOS DE TRATAMENTO .....	13
4.1.6. MELHORIAS PROPOSTAS .....	14
5. PROPOSTA DO PROGRAMA .....	16
6. LINHAS DE CÓDIGO DO PROGRAMA .....	17
7. CONCLUSÃO .....	27
REFERÊNCIAS .....	28

## 1. INTRODUÇÃO

A sociedade de hoje é uma imensa massa de produção de lixo, de materiais tóxicos e produtos químicos que podem arruinar todo meio ambiente no qual vivemos e sabemos que os maiores produtores de todos esses efluentes são as empresas, porém da mesma forma que muito é produzido, também muito é reciclado e neutralizado mas, como é feito o processo de reciclagem e neutralização desses efluentes.

É dever de toda empresa ter em suas dependências um setor de reciclagem de efluentes ou um contrato com uma empresa especializada nesse tipo de serviço. Esse tipo de reciclagem e método (Terceirizado ou na Própria empresa) varia de acordo com o produto gerado no processo de fabricação da mesma. Se for um efluente químico onde poucos são especializados na neutralização a tendência é ter um setor na própria empresa, como a Petrobrás BR, Fibria e El Dorado.

Já empresas metalúrgicas tendem a terceirizar essa reciclagem por ser um serviço aparentemente menos complicado e existem bastante empresas especializadas na área, porém há algumas que possuem o próprio setor de reciclagem e neutralização.

Dessa forma, visando facilitar o registro de dados dos elementos que são utilizados para os tratamentos, a quantidade de materiais neutralizados, qual o setor com mais eficiência de uma determinada empresa, etc. Foi desenvolvido um programa no qual a partir da entrada de um arquivo de documento de texto com algumas informações, tem como saída os dados mencionados acima.

## **2. RESÍDUOS INDUSTRIAIS E URBANOS**

### **2.1. CONCEITO DE RESÍDUO**

Qualquer substância ou objeto de que o ser humano pretende desfazer-se por não lhe reconhecer utilidade. A produção de resíduos é causadora de poluição e tem vindo a aumentar com o desenvolvimento socioeconómico e tecnológico das sociedades.

### **2.2. RESÍDUOS SÓLIDOS URBANOS**

São correntemente designados lixos. Incluem resíduos domésticos, industriais e hospitalares. Podem causar poluição da água, do solo ou da atmosfera.

### **2.3. ATERROS SANITÁRIOS**

Os Aterros Sanitários devem ser construídos em locais com características geológicas adequadas e são revestidos com materiais impermeáveis, como argila ou plástico, que previnem a infiltração no solo de substâncias lixiviadas.

### **2.4. SUBSTÂNCIAS LIXIVIADAS**

As substâncias lixiviadas (quando a água das chuvas se infiltra, dissolve substâncias químicas e arrasta-as consigo) são recolhidas e enviadas para uma estação de tratamento e os gases produzidos pelas bactérias decompositoras (biogás) podem ser utilizados na obtenção de energia.

Após estarem lotados, os aterros são selados, ou seja, tapados com uma cobertura de plásticos e de terra que permite o desenvolvimento de plantas que diminuirão o impacto paisagístico.

## **2.5. COLETA SELETIVA E RECICLAGEM**

Muitos destes resíduos sólidos são compostos de materiais recicláveis e podem retornar a cadeia de produção, gerando renda para trabalhadores e lucro para empresas. Para que isto ocorra, é necessário que haja nas cidades um bom sistema de coleta seletiva e reciclagem de lixo. Cidades que não praticam este tipo de processo, jogando todo tipo de resíduo sólido em aterros sanitários, acabam poluindo o meio ambiente. Isto ocorre, pois muitos resíduos sólidos levam décadas ou até séculos para serem decompostos.

## **3. TIPOS DE RESÍDUOS QUÍMICOS E EFLUENTES NO SETOR ESCOLHIDO**

Efluentes industriais representam grande parte da poluição do meio ambiente por conta do uso de produtos químicos aliados ao alto consumo de água no processamento de matéria-prima. Os efluentes líquidos são águas que foram utilizadas em algum tipo de processo e se encontram contaminadas com substâncias como amônia e benzeno.

Na indústria metalúrgica isso não é diferente. Os resíduos gerados (como óleos e graxas, só para citar alguns) são os de Classe I, considerados perigosos, provenientes dos processos de lavagem das chapas, tornos e de pintura. No processo de galvanização, em que as peças metálicas recebem tratamento anticorrosivo, das quais são utilizados desengraxantes alcalinos e banhos de fosfato, é preciso lavá-las várias vezes até a água sair limpa, eliminando óleos, óxidos e tintas. Os compostos de metais pesados são tóxicos e alguns até cancerígenos. Por isso, o descarte de forma irresponsável em lençóis freáticos, rios e mares acarretam na contaminação dessas áreas.

Os resíduos químicos e efluentes escolhidos para análise foram: Mercúrio, Nitrato, Nitrito, Oxigênio, Ozônio, Permanganato, Sulfato, Sulfito, Zinco.

#### 4. CONDIÇÕES ATUAIS

É evidente que nos dias de hoje, vivemos em um país, no qual a poluição por resíduos e efluentes químicos é altíssima, no entanto, deve haver uma rigorosa forma de tratamento nas indústrias para que a sociedade e a natureza não sofram as consequências por conta disso, vindas a acarretar problemas de saúde e desmatando à natureza.

Acredita-se que os metais talvez sejam os agentes tóxicos mais conhecidos pelo homem. Há aproximadamente 2.000 anos a.C., grandes quantidades de chumbo eram obtidas de minérios, como subproduto da fusão da prata e isso provavelmente tenha sido o início da utilização desse metal pelo homem. O elevado desenvolvimento industrial ocorrido nas últimas décadas, tem sido um dos principais responsáveis pela contaminação de nossas águas e solos, seja pela negligência no seu tratamento antes de despejá-las nos rios ou por acidentes e descuidos cada vez mais frequentes, que propiciam o lançamento de muitos poluentes nos ambientes aquáticos.

A Indústria metalúrgica é uma grande responsável pela poluição nas águas e solos por resíduos químicos, por isso deve haver uma grande atenção voltada para o tratamento desses resíduos.

Podemos inferir, que há poluição sim, porém também há como fazer um tratamento de qualidade, para que isso aconteça, deve haver uma maior fiscalização e relação a isso pelo governo, e muitas mais rigorosas.

Hoje em dia algumas empresas começaram a perceber que gerar resíduos é sinônimo de perdas econômicas a longo prazo, pois isto representa:

- Perda de insumos, isto é, desperdício de matérias primas, água e energia;
- Gastos adicionais com o tratamento, armazenamento e disposição final dos resíduos;
- Risco potencial à saúde pública e ao meio ambiente: resíduos podem provocar graves acidentes ambientais quando manuseados, tratados, ou dispostos de forma inadequada.

- No Entanto, hoje o tratamento em relação a liberação de resíduos é mais forte.

#### **4.1. MEDIDAS PARA O DESENVOLVIMENTO SUSTENTÁVEL**

A tendência atual é que as empresas façam do seu desempenho ambiental um fator diferencial no mercado. O que significa, em alguns casos, adotar requisitos internos até mais restritivos que os legalmente impostos no país.

É comum proceder ao tratamento de resíduos industriais com vistas à sua reutilização ou pelo menos à sua inertização. Dada à diversidade destes resíduos, não existe um processo de tratamento pré-estabelecido, havendo sempre a necessidade de realizar pesquisas e desenvolvimento de processos economicamente viáveis.

##### **4.1.1. INCINERAÇÃO**

A incineração é um processo de queima controlada na presença de oxigênio, no qual os materiais à base de carbono são reduzidos a gases e materiais inertes (cinzas e escórias de metal) com geração de calor. Esse processo permite a redução em volume e peso dos resíduos sólidos em cerca de 60 a 90%. Normalmente, o excesso de oxigênio empregado na incineração é de 10 a 25% acima das necessidades de queima dos resíduos.

Em grandes linhas, um incinerador é um equipamento composto por duas câmaras de combustão, onde na primeira câmara os resíduos sólidos e líquidos são queimados à temperatura variando entre 800 e 1.000 °C. Na segunda câmara, os gases provenientes da combustão inicial são queimados a temperaturas da ordem de 1.200 a 1.400 °C. Os gases da combustão secundária são rapidamente resfriados para evitar a recomposição das extensas cadeias orgânicas tóxicas e em seguida tratados em lavadores, ciclones ou precipitadores eletrostáticos, antes de serem lançados na atmosfera através de uma chaminé.



Como a temperatura de queima dos resíduos não é suficiente para volatilizar os metais, estes se misturam às cinzas, podendo ser posteriormente separados destas e recuperados para comercialização.

Para os resíduos tóxicos contendo cloro, fósforo ou enxofre, além da necessidade de maior permanência dos gases na câmara (cerca de dois segundos), são necessários sofisticados sistemas de tratamento para que estes possam ser lançados na atmosfera.

Já os resíduos compostos apenas por átomos de carbono, hidrogênio e oxigênio necessitam somente um sistema eficiente de remoção do material particulado expelido juntamente com os gases da combustão.

Existem diversos tipos de fornos de incineração. Os mais comuns são os de grelha fixa, de leito móvel e o rotativo.

#### **Vantagens:**

- Garantia da eficiência de tratamento, quando em perfeitas condições de funcionamento;
- Redução substancial do volume de resíduos a ser disposto (cerca de 95%).

#### **Desvantagens:**

- Custo operacional e de manutenção elevado;
- Manutenção difícil, exigindo trabalho constante de limpeza no sistema de alimentação de combustível auxiliar, exceto se for utilizado gás natural;
- Elevado risco de contaminação do ar devido a geração dioxinas da queima de materiais clorados;
- Risco de contaminação do ar pela emissão de materiais particulados;
- Elevado custo de tratamento dos efluentes gasosos e líquidos (águas de arrefecimento das escórias e de lavagem de fumos).

#### 4.1.2. ATERRO INDUSTRIAL

É uma alternativa de destinação de resíduos industriais, que se utiliza de técnicas que permitem a disposição controlada destes resíduos no solo, sem causar danos ou riscos à saúde pública, e minimizando os impactos ambientais.

Essa técnica consiste em confinar os resíduos industriais na menor área e volume possíveis, cobrindo-os com uma camada de material inerte na conclusão de cada jornada de trabalho ou intervalos menores, caso necessário.

Os aterros industriais são classificados nas classes I, II ou III, conforme a periculosidade dos resíduos a serem dispostos. Os aterros **Classe I** podem receber resíduos industriais perigosos, já os **Classe II** recebem resíduos não inertes e por fim os da **Classe III**, somente resíduos inertes.

Célula é módulo de um aterro industrial que contempla isoladamente todas as etapas de construção, operação e controle exigidas para um aterro industrial.

##### 4.1.2.1. LOCALIZAÇÃO DE ATERROS DE RESÍDUOS INDUSTRIAIS PERIGOSOS

Deverão ser selecionadas preferencialmente, áreas naturalmente impermeáveis para construção de aterros de resíduos industriais. Estas áreas se caracterizam pelo baixo grau de saturação, pela relativa profundidade do lençol freático e pela predominância, no subsolo, de material argiloso.

Não é possível instalar aterros industriais em áreas inundáveis, de recarga de aquíferos, em áreas de proteção de mananciais, mangues e habitat de espécies protegidas, ecossistemas de áreas frágeis ou em todas aquelas definidas como de preservação ambiental permanente, conforme legislação em vigor.

Deverão ser respeitadas as distâncias mínimas estabelecidas em norma, a corpos d'água, núcleos urbanos, rodovias e ferrovias, quando da escolha da área do aterro.

A construção de aterros em áreas cujas dimensões não possibilitem uma vida útil para o aterro igual ou superior a 20 (vinte) anos, não deverá ser executada.

#### **4.1.2.2. IMPERMEABILIZAÇÃO INFERIOR**

Os aterros industriais deverão possuir sistema duplo de impermeabilização inferior composto de manta sintética sobreposta a uma cama de argila compactada, de forma a alcançar coeficiente de permeabilidade menor ou igual a  $1,0 \times 10^{-7}$  cm/s, com espessura mínima de 60 centímetros, devendo ser mantida uma distância mínima de 2 metros entre a superfície inferior do aterro e o nível mais alto do lençol freático.

Sobre o material sintético deverá ser assentada uma camada de terra com espessura mínima de 50 centímetros. Na escolha da manta sintética a ser aplicada, deverão ser observados os seguintes aspectos:

- Resistência química aos resíduos a serem dispostos, assim como o envelhecimento à ozona, à radiação, à ultravioleta e aos micro-organismos, essas características devem ser comprovadas através de ensaios de laboratório;
  - Resistência à intempéries para suportar os ciclos de umedecimento;
  - Secagem;
  - Resistência a tração, flexibilidade e alongamento, suficiente para suportar os esforços de instalação e de operação;
  - Resistência à laceração, abrasão e punção de qualquer material pontiagudo ou cortante que possa estar presente nos resíduos;
  - Facilidade para execução de emendas e reparos em campo, em quaisquer circunstâncias.

O sistema duplo de impermeabilização deverá ser construído de modo a evitar rupturas devido a pressões hidrostáticas e hidrogeológicas, condições climáticas, tensões da instalação, da impermeabilidade ou aquelas originárias da operação diária.

O sistema duplo de impermeabilização deverá ser assentado sobre uma base ou fundação capaz de suportá-lo, bem como resistir aos gradientes de pressão acima e abaixo da impermeabilização de forma a evitar sua ruptura por assentamento com pressão ou levantamento do aterro.

#### **4.1.2.3. IMPERMEABILIZAÇÃO SUPERIOR (COBERTURA FINAL)**

Quando do fechamento de cada célula de um aterro industrial, a impermeabilização superior a ser aplicada deverá garantir que a taxa de infiltração na área seja tão pequena quanto possível. Desta forma, esta impermeabilização deverá ser no mínimo tão eficaz quanto o sistema de impermeabilização inferior empregado.

O sistema de impermeabilização superior deverá compreender das seguintes camadas, de cima para baixo:

- Camada de solo original de 60 (sessenta) centímetros, para garantir o recobrimento com vegetação nativa de raízes não axiais;
- Camada “Drenante” de 25 (vinte e cinco) centímetros de espessura, com coeficiente de permeabilidade maior ou igual a  $1,0 \times 10^{-3}$  cm/s;
- Manta sintética com a mesma especificação utilizada no sistema de impermeabilização inferior;
- Camada de argila compactada de 50 (cinquenta) centímetros de espessura, com coeficiente de permeabilidade menor ou igual a  $1,0 \times 10^{-7}$  cm/s.
-

#### **4.1.3. RECICLAGEM DE RESÍDUOS SÓLIDOS**

A reciclagem em geral trata de transformar os resíduos em matéria-prima, gerando economias no processo industrial. Isto exige grandes investimentos com retorno imprevisível, já que é limitado o repasse dessas aplicações no preço do produto, mas esse risco reduz-se na medida em que o desenvolvimento tecnológico abre caminhos mais seguros e econômicos para o aproveitamento desses materiais.

Para incentivar a reciclagem e a recuperação dos resíduos, alguns estados possuem bolsas de resíduos, que são publicações periódicas, gratuitas, onde a indústria coloca os seus resíduos à venda ou para doação.

#### **4.1.4. PADRONIZAÇÃO DE RECIPIENTES DE MATERIAIS RECICLÁVEIS**

O Conselho Nacional do Meio Ambiente – CONAMA publicou no dia 19 de junho de 2001, no Diário Oficial, a Resolução nº 275, de 25 de abril de 2001, que define as cores que serão utilizadas nos recipientes de materiais recicláveis. O objetivo da decisão é estabelecer um padrão nacional de cores e adequá-lo aos padrões internacionais.

#### **4.1.5. OUTROS PROCESSOS DE TRATAMENTO**

- Neutralização, para resíduos com características ácidas ou alcalinas;
- Secagem ou mescla, que é a mistura de resíduos com alto teor de umidade com outros resíduos secos ou com materiais inertes, como serragem;
- Encapsulamento, que consiste em revestir os resíduos com uma camada de resina sintética impermeável e de baixíssimo índice de lixiviação;
- Incorporação, onde os resíduos são agregados à massa de concreto ou de cerâmica em uma quantidade tal que não prejudique o meio ambiente, ou ainda que possam ser acrescentados a materiais combustíveis sem gerar gases prejudiciais ao meio ambiente após a queima;

O Sistema de Gestão Ambiental, conforme a série ISO 14000, fundamenta-se na adoção de ações preventivas à ocorrência de impactos

adversos ao meio ambiente. Trata-se de assumir uma postura proativa com relação às questões ambientais.

### **OS CINCO PRINCÍPIOS DO SGA:**

- Conhecer o que deve ser feito, definindo sua política de meio ambiente;
- Elaborar o Plano de Ação para atender aos requisitos de sua política ambiental;
- Assegurar condições para o cumprimento dos objetivos e metas ambientais e implementar ferramentas de sustentação necessárias;
- Realizar avaliações qualitativas e quantitativas periódicas do desempenho ambiental da empresa;
- Revisar e aperfeiçoar a política do meio ambiente, os objetivos e metas ambientais e as ações implementadas para assegurar a melhoria contínua do desempenho ambiental da empresa.

#### **4.1.6. MELHORIAS PROPOSTAS**

O tratamento dos efluentes da indústria metalúrgica ocorre por processos físico-químicos, no caso, precipitação química, e também por processos biológicos. Também existe a possibilidade de serem utilizados agentes adsorventes, como carvão ativado e zeólitas de alumínio silicato, entre outros, para a possível remoção de contaminantes. Existem casos em que os efluentes tratados podem ser reaproveitados no próprio processo produtivo, como no resfriamento de misturas reativas. No caso do lodo, ele recebe um tratamento físico-químico e fica estocado em um tanque de decantação. O encaminhamento para uma Estação de Tratamento deve seguir à legislação e ser feito por empresa idônea e autorizada no transporte desses resíduos.

Com a escassez de água dos dias atuais, toda e qualquer possibilidade de reuso e preservação de mananciais e afins devem ser consideradas. O uso racional deste recurso deve ser visto com muito cuidado pelas empresas e fiscalizados pela sociedade ao adquirirem produtos oriundos de empresas ambientalmente responsáveis.

A gestão ambiental vem ganhando seu espaço no meio empresarial, através da consciência ecológica que vem sido desenvolvida em vários setores da sociedade mundial, envolvendo também, a educação. Conscientizando as pessoas de que se deve originar o menor impacto possível sobre o meio

ambiente, desde a escolha das melhores técnicas, até o cumprimento da legislação.

Ter um forte comprometimento com a política ambiental é de suma importância, algumas das políticas são:

- Atender aos requisitos legais aplicáveis à indústria e seus produtos e processos;
- Conscientizar colaboradores, fornecedores e comunidades vizinhas para garantir o compromisso de todos com a preservação ambiental;
- Capacitar constantemente os colaboradores;
- Identificar e monitorar aspectos e impactos ambientais, visando minimizá-los;
- Utilizar de maneira eficiente e sustentável os insumos e recursos naturais;
- Reduzir as emissões de resíduos, visando sempre à reutilização ou reciclagem quando possível;
- Manter equipamentos com tecnologias avançadas, devidamente reguladas e em constante manutenção;
- Manter atualizado o Sistema de Gestão Ambiental;

## 5. PROPOSTA DO PROGRAMA

Em vista de tudo o que foi mostrado, foi elaborado esse programa como uma forma de tentar ajudar, com uma pequena ação, para que um dia essa pequena ação possa chegar a mais pessoas e quem sabe, gerar algo bem maior do que apenas um programa.

A principal ideia do programa é facilitar o registro e a comparação de dados, no caso, a comparação dos efluentes reciclados e neutralizados de uma empresa ou determinado setor gerador de resíduos de uma sociedade, bem como procurar resíduos e suas características através do nome digitado pelo usuário, mostrar a eficiência do setor em análise.

Para que essa proposta seja concluída, o programa conta com as seguintes principais funções:

- Func1: Le os dados do arquivo e os coloca em variáveis;
- Func2: Faz a comparação entre os dados e mostra o de maior quantidade de resíduo reciclado e neutralizado;
- Func3: Faz a soma de todos os resíduos neutralizados encontrados no arquivo e exibe ao usuário;
- Func4: Inicia uma pesquisa pelo arquivo através de um nome de elemento digitado pelo usuário;
- Func5: Calcula a eficiência do setor escolhido pelo usuário, usando a relação, Reagente x Produto Reciclado;



## 6. LINHAS DE CÓDIGO DO PROGRAMA

```

1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <string.h>
4  #include <ctype.h>
5  #include <locale.h>
6  #include <conio.h>
7  #include <windows.h>
8
9  #define maxest 4
10 #define TAM 13
11 #define MAX 50
12
13
14 void textcolor (int color){
15     SetConsoleTextAttribute (GetStdHandle (STD_OUTPUT_HANDLE), color );
16 }

```

Da linha 1 até linha a 7 o programa está fazendo a chamada de 7 bibliotecas:

Linha 1: Biblioteca padrão, para as funções de entrada(i) e saída(o);

Sem essa biblioteca não seria possível utilizar comandos como “**scanf**” ou “**printf**”;

Linha 2: Biblioteca que possui funções de alocação de memória, controle de processos, conversões entre outras;

Linhas 3: Biblioteca onde estão localizadas as funções de STRINGS, sem a chamada dessa biblioteca não seria possível utilizar alguns comandos como “**strcat**”, “**strcmp**”, “**strcpy**” entre vários outros relacionados a STRINGS;

Linha 4: Biblioteca contém declarações para manipulação de caracteres. Usada quando se trabalha com diferentes idiomas e alfabetos;

Linha 5: Conhecido como “localizar”, que tem por finalidade fazer a localização do programa a uma determinada região ou idioma, dessa forma, podendo usar os acentos, caracteres especiais, etc;

Linha 6: A ideia de colocar a biblioteca “**Conio.h**” é deixar o programa restrito apenas para usuários de que tenham MS-DOS, é apenas uma biblioteca a mais, um recurso que pode ser utilizado, e não algo essencial para que todo o programa funcione;

Linhas 7: É uma biblioteca que contém declarações para todas as funções da API do WINDOWS, todos os macros comuns utilizados pelos programadores do Windows, e todos os tipos de dados utilizados pelas várias funções e subsistemas.

API (Application Programming Interface), que significa “Interface de Programação de Aplicativos”, é um conjunto de rotinas e padrões de programação para acesso a um aplicativo de software ou plataforma.

Possibilita você a fazer coisas como:

- Criar janelas e botões.

Enumerar arquivos de um diretório.

Consultar informações sobre o sistema (processos, serviços, janelas)

Coisas do gênero.

Da linha 9 a 11 estão as definições que foram criadas para o programa, ao utiliza-las, todas as vezes que for colocado a sequencia de caracteres TAM, será atribuído o valor definido que está em frente a essa sequência, no caso, 13.

Na linha 14 tem se a primeira função do programa, a qual é acionada toda vez que se encontrar a palavra “textcolor”, ela tem a função de procurar na API do Windows (por isso a aplicação de <Windows.h>) uma determinada cor, que é pedida pelo programa, e por resultado, colore a linha na qual a função foi chamada.

```

170 int main(void)
171 {
172     //////////////// HABILITANDO ACENTUAÇÃO //////////////////
173
174     setlocale(LC_ALL, "portuguese");
175
176     menu();
177 }
178

```

Função Main, a função principal do programa, no caso só esta servindo para chamar o menu e habilitar o idioma português, para que as palavras digitadas possam ter acento e caracteres especiais.

```

18 //////////////////////////////////////////////////////////////////// PROTÓTIPOS //////////////////////////////////////////////////////////////////////
19 int menu();
20
21 int data_do_arquivo (int dia, int mes, int ano);
22 void exibir(dados, produto);
23 void voltar();
24 void func1(dados, produto,maxlin);
25 void func2 (dados,produto);
26 void func3 (dados);
27 void func4 (produto,dados);
28 void func5 (produto,dados);
29
30 ////////////////////VARIÁVEIS GLOBAIS ////////////////////
31 int lin[maxest];
32 float dados[MAX][TAM];
33 char produto[MAX][TAM];

```

Na linha 18 temos uma forma de comentário bastante peculiar, talvez pouco usada, mas aqui foi usada dessa forma como uma maneira de chamar atenção, ao invés de apenas “//” ou “/\*” usa se essa quantidade anormal de “/” com a finalidade de identificar de forma mais fácil algo, nas linhas subsequentes da 19 a 28 tem-se os protótipos das funções que serão usadas no programa.

Os protótipos servem como um alerta ao compilador para se preparar e reservar os itens necessários para a criação da função e para que possa achar a mesma sem que ele tenha visto a declaração da função por exemplo.

Nas linhas 31,32 e 33 há 3 variáveis globais que serão usadas por todo o programa, lembrando que variáveis do tipo global têm a singularidade de poder ser utilizada e terem seus valores alterados por qualquer função do programa, basta apenas que tenha as indicações corretas.

```

34 ////////////////////////////////////////////////// MENU ///////////////////////////////////
35 int menu()
36 {
37
38     setlocale(LC_ALL, "portuguese");
39
40     //////////// * VARIÁVEIS * ////////////
41
42     int op, i, j, maxlin[5];
43
44     //////////// INICIALIZANDO MATRIZ COM OS DADOS NUMÉRICOS DE CADA INFORMAÇÃO ////////////
45     do{
46         textcolor(10);
47         printf("\n\n\n\t\t.: Centro de Análise de Tratamento de Efluêntes  :.\n\n");
48         textcolor(15);
49         printf("\t(1) Carregar dados da estação\n");
50         printf("\t(2) Principal resíduo produzido por estação\n");
51         printf("\t(3) Total de resíduos tratados por estação\n");
52         printf("\t(4) Consulta por resíduo: maior e menor consumo de reagentes\n");
53         printf("\t(5) Listagem de Eficiência: Resíduo x Estação (todos os resíduos)\n");

```

A partir da linha 35 inicia-se a função “**menu**”, onde será exibida essas informações todas as vezes em que a função “**menu()**” for chamada durante o programa. Na linha 42 estão declaradas as 4 variáveis que serão utilizadas enquanto “roda” essa parte do programa.

```

54
55     printf("\t(6) Encerrar\n\n");
56     printf("\tOpção: ");
57     scanf("%d", & op);
58     fflush(stdin);
59     if(op <= 0 || op >= 7){
60
61         textcolor(14);
62         printf("\n\n\n\t\tOpção inválida! Tente novamente.\n\n\n");
63         textcolor(15);
64         system("pause");
65         system("cls");
66     }
67     while(op <= 0 || op >= 7);
68     switch (op){
69
70         case 1:
71             system("cls");
72             func1(dados, produto);
73             break;
74

```

Continuação da função “**menu**”.

Na linha 57 “**fflush(stdin)**” comando usado para limpar o Buffer do teclado.

**Buffer:** região de memória usada como área de armazenamento temporário de dados durante sua transferência entre dispositivos de diferentes taxas de transferência.

Na linha 63 e 64 existem dois comandos cujo nome inicia-se com “system”, A função **system()** com o parâmetro **PAUSE** permite que o usuário veja os resultados antes que a janela seja fechada. Sem essa função, o programa seria executado e a janela seria automaticamente fechada, já a com parâmetro **CLS** limpa a tela, assim como quando se usa o **PROMPT DE COMANDO** e são digitadas essas siglas.

Na linha 68 inicia-se o Switch que leva como parâmetro a variável “**op**”, a primeira chamada de função para os arquivos do programa começa na linha 72, caso o usuário digite 1 no menu.

Linha 73 observa-se o uso do comando break, sem essa aplicação, a função Switch não entenderia que o “case 1” havia terminado, então passaria direto e faria os “cases” seguintes como se fosse um só.

```

75         case 2:
76             func2(dados, produto);
77             break;
78
79         case 3:
80             func3(dados);
81             break;
82
83         case 4:
84             func4(produto, dados);
85             break;
86
87         case 5:
88             func5(produto, dados);
89             break;
90
91         case 6:
92             voltar();
93             break;
94     }
95     return op;
96 }

```

Da linha 34 até a linha 96 está localizada toda a função “**menu**”, onde também estão sendo chamada todas as funções, **func1,func2,func3,func4,func5,func6**, todas dependem da opção selecionada pelo usuário. A função se encerra com um “**return**” obrigando a função a retornar o valor retido na variável “**op**”.

```

98  //////////////////////////////////// TRATATIVA DA OPÇÃO ESCOLHIDA ////////////////////////////////////
99
100 int data_do_arquivo (int dia, int mes, int ano)
101 {
102
103     int ano_bissexto, data_valida = 0; // USADAS COMO VARIÁVEIS BOOLEANAS
104     int data = 0;
105
106     /* Na atribuição abaixo, se as condições forem satisfeitas, a variável
107        recebe um valor maior do que 0. Esta variável será usada para
108        testes booleanos, mais abaixo... */
109
110     ano_bissexto = ( ((ano % 4 == 0) && (ano % 100 > 0)) || (ano % 400 == 0) );
111
112     if (dia >= 1 && dia <= 31 && mes >= 1 && mes <= 12 && ano >= 1900 && ano <= 2100){
113         // TODOS OS DADOS NAS FAIXAS VALIDAS?
114         if (mes == 1 || mes == 3 || mes == 5 || mes == 7 || mes == 8 || mes == 10 || mes == 12){
115             data_valida = (dia <= 31); // ESTES MESES TÊM 31 DIAS
116         }
117
118     }else{
119         if (mes == 2) // FEVEREIRO É UM CASO A PARTE...
120         {
121             if (ano_bissexto) // SE O ANO FOR BISSEXTO
122             {
123                 data_valida = (dia <= 29); // O DIA PODE IR ATÉ 29
124             }else{
125                 data_valida = (dia <= 28); // SENÃO SÓ ATÉ 28
126             }
127         }else
128         {
129             data_valida = (dia <= 30); // SE CHEGOU AQUI É UM MÊS DE 30 DIAS
130         }
131     }
132
133 }
134
135 return (data_valida); // E RETORNAMOS O VALOR; RETORNA 0 SE DATA INVÁLIDA
136 }

```

Da linha 98 até a 136 está localizada a função “**data\_do\_arquivo**”, a qual verifica a data digitada pelo usuário, para que possa ter certeza de validação do arquivo que contém os dados que serão utilizados no programa.

```

138 void voltar()
139 {
140     int verif;
141     do{
142
143         system("cls");
144         textcolor(14);
145         printf("\nTem certeza que deseja encerrar? Digite (0)Não e (1)Sim\n");
146         textcolor(15);
147         printf("\n\tOpção: ");
148         scanf("%d", &verif);
149         fflush(stdin);
150
151         if((verif <= -1) || (verif >= 2)){
152             textcolor(14);
153             printf("\n\n\t\tOpção inválida!! Tente novamente!\n\n\n");
154             textcolor(15);
155             system("pause");
156         }
157
158     }while((verif <= -1) || (verif >= 2));
159 }

```

```

158 }while((verif <= -1) || (verif >= 2));
159 if(verif == 0)
160 {
161     system("cls");
162     main();
163
164 }else if(verif == 1){
165     exit (1);
166
167 }
168 }

```

Função “voltar” que é chamada quando o usuário digita a tecla “6” quando ainda está dentro da função “menu”, dentro da “voltar” o usuário terá a opção de retornar ao programa caso tenha digitado a tecla “6” por engano e não gostaria de encerrar o programa naquele dado momento.

```

179 void func1(float dados[][TAM], char produto[][TAM])
180 {
181
182     int n = 0;
183     int d, m, a, dias = 0;
184
185     do{
186         printf("\n Por favor informe a numeração da estação que deseja carregar: ");
187         scanf("%d", &n);
188         fflush(stdin);
189
190     }while(n <= 0 || n >= 6);
191
192     while(!dias)
193     {
194
195         printf("Entre com dia, mês e ano no formato dd/mm/aaaa: ");
196         scanf("%d/%d/%d", &d, &m, &a);
197         fflush(stdin);
198         dias = data_do_arquivo(d,m,a);
199
200         if (!dias)
201         {
202             textcolor(14);
203             printf ("\t\7Data inválida! Tente novamente..\n\n");
204         }
205     }
206
207
208     char numEsta[10];
209     sprintf(numEsta, "%d", n);
210     char data1[5];
211     char data2[5];
212     char data3[5];
213     sprintf(data1, "%d", d);
214     sprintf(data2, "%d", m);
215     sprintf(data3, "%d", a);
216
217     char nome[100];
218
219     strcpy(nome, "Est_");
220     strcat(nome, numEsta);

```

Linha 179, início da função “**func1**”, ela servirá para leitura de todos os dados do arquivo.

Já na linha 209,213,214 e 215 há um comando chamado “**sprintf**” que tem por finalidade juntar variáveis e montar em uma string, na linha 219 temos o início da montagem do nome do arquivo que será aberto.

```

221     strcat(nome, "_");
222     strcat(nome, data1);
223     strcat(nome, data2);
224     strcat(nome, data3);
225     strcat(nome, ".txt");
226
227
228     //////////// ABRINDO ARQUIVO ////////////
229
230     char compon[20];
231     float valor1 = 0, valor2 = 0;
232     int i = 0;
233     FILE *arq;
234
235
236     arq = fopen(nome, "r");
237     if(arq == NULL)
238     {
239         textcolor(14);
240         printf("\n\n\t\7Erro, não foi possível abrir o arquivo\n\n\n");
241         textcolor(15);

```

A função “**strcpy(variavel,string ou variavel)**” é usada para copiar uma string que esteja ou fora ou em uma variável e passar para outra variável, como se o que está depois da vírgula, se aglomerasse dentro da variável da esquerda antes da vírgula.

Linha 233, comando **FILE**, utilizado para abrir arquivos, onde é feito um ponteiro no caso “**arq**” que irá indicar regiões da memória que estará localizado um arquivo.

Lendo a linha 236 a 241, o ponteiro “**arq**” abrirá para leitura “**r**” um arquivo cuja “**url**” está localizada na variável “**nome**” e está servindo de parâmetro para função “**fopen**” que serve para abrir dados do tipo “**FILE**”.



```

242     system("pause");
243     system("cls");
244     main();
245 }else
246 {
247     system("cls");
248     printf("\tOs Dados Abaixo...\n\n");
249     textcolor(15);
250     printf("ELEMENTOS QUÍMICOS      QUANTIDADE DE REAGENTE      QUANTIDADE RECICLADA\n\n");
251
252     while(fscanf(arq, "%s%f%f", compon, &valor1, &valor2) != EOF)
253     {
254         //MOVENDO OS DADOS PARA AS MATRIZES GLOBAIS
255
256         strcpy(produto[i],compon);
257         dados[n][i] = valor1; //ELEMENTOS DA ESTAÇÃO N, NA POSIÇÃO I
258         dados[n+5][i] = valor2;
259         i++;
260     }
261
262     exibir(dados, produto);
263 }

```

Linha 252 a 259, armazenagem dos dados para as variáveis, “compon”, “valor1” e “valor2”.

Leitura da linha 252 a 259: Enquanto a variável “arq” for diferente do Fim do Arquivo (EOF), então leia “fscanf” o que há no arquivo e coloque nas variáveis, “compon”, “valor1” e “valor2”. E também ao mesmo tempo, copie o que há em “compon” para “produto” na posição “i”, e “dados” na posição “n” e “i” receba o que há em “valor1” e dados na posição “n+5” e “i” receba o que há na variável “valor2” e por fim, incremente “i” com mais 1.

Ao final, na linha 262 é feita a chamada da função “exibir”.

```

264     system("pause");
265 }
266
267 void exibir(float dados[TAM][TAM], char produto[][TAM])
268 {
269     int i;
270     for(i = 0; i <= TAM; i++)
271     {
272         printf("%s\t\t\t\t%.2f\t\t\t%.2f\n", produto[i], dados[1][i], dados[6][i]);
273     }
274     printf("\n\n\t\t\t\t\t7...Foram importados do arquivo com sucesso !!! \n\n");
275     system("pause");
276     system("cls");
277     menu();
278 }
279
280
281 void func2(float dados[][TAM], char produto[][TAM])
282 {
283     system("cls");
284     int i;
285     int MaiorValor = 0, soma = 0;

```

Linha 267 a 278, função “exibir” mostra o que está sendo lido pela função “func1”, em “exibir” serve também para ver se foi aberto o arquivo correto, já é possível o usuário conferir se será feita a leitura do que deseja.



```

286     for(i = 0; i <= 14; i++)
287     {
288         if(MaiorValor < dados[6][i])
289         {
290             MaiorValor = dados[6][i];
291             soma += i;
292         }
293     }
294     printf("\n\nO Principal resíduo produzido é: %s\n\n", produto[soma]);
295     system("pause");
296     system("cls");
297     menu();
298 }

```

A partir da linha 281 até a 297 é a função “func2” que faz a comparação dos dados das variáveis globais e as compara, mostrando qual o produto com a maior quantidade de material reciclado no final do processo de neutralização.

```

300 void func3(float dados[][TAM])
301 {
302
303     system("cls");
304
305     int i, TotalProduto=0;
306
307     for(i = 0; i <= 14; i++)
308     {
309         TotalProduto += dados[6][i];
310     }
311
312     printf("\n\nA quantidade Total de Produto Reciclado é: %d\n\n", TotalProduto);
313     system("pause");
314     system("cls");
315     menu();
316 }

```

Linha 300 a 316, função “func3” onde é feita a soma dos produtos reciclados e mostrado a quantidade total de Produto reciclado ao usuário.

```

318 void func4(char produto[][TAM], float dados[TAM][TAM])
319 {
320     char pesquisa[50];
321     int i = 0, NaoEncontrado = 0;
322
323     system("cls");
324     fflush(stdin);
325
326     printf("Informe o nome do Resíduo desejado: \n");
327     gets(pesquisa);
328     fflush(stdin);
329
330     for(i = 0; i <= TAM; i++)
331     {
332         if( (strcmp(pesquisa, produto[i]) ) == 0)
333         {
334             printf("\n\tELEMENTO QUÍMICO\t\tQUANTIDADE DE REAGENTE\t\tQUANTIDADE RECICLADA");
335             printf("\n\nPesquisa: %s\t\t\t %0.2f\t\t\t %0.2f\n\n", produto[i], dados[1][i], dados[6][i]);
336             i = TAM;
337             NaoEncontrado = 1;
338         }
339     }

```

Função “func4” onde é feita a procura de um elemento através do nome digitado pelo usuário e mostrado os dados desse elemento. Linha 332, função “strcmp” faz a comparação entre duas strings, ignorando letras maiúsculas.

```

340     if(NaoEncontrado == 0){
341
342         textcolor(14);
343         printf("\t\tResíduo não encontrado!! Verifique se o nome está correto!\n\n\n");
344         textcolor(15);
345     }
346     system("pause");
347     system("cls");
348     menu();
349 }
351 void func5(char produto[][TAM], float dados[TAM][TAM])
352 {
353     int i;
354     float eficiencia[TAM];
355
356     system("cls");
357
358     printf("NOME    |||  EFICIENCIA\n\n");
359     for(i = 0; i < TAM; i++)
360     {
361         eficiencia[i] = ((dados[6][i] - dados[1][i]) / dados[6][i]) * 100;
362         //CALCULO DE EFICIÊNCIA!! E = (AFLUENTE - EFLUENTE / AFLUENTE) * 100;
363         printf("%s |||  %.2f %%\n", produto[i], eficiencia[i]);
364     }
365     getchar();
366     system("cls");
367     menu();
368 }

```

E por fim, função “**func5**” que faz o cálculo de eficiência de todos os elementos do arquivo e mostra na tela ao usuário.

## **7. CONCLUSÃO**

Por fim, com base em tudo que foi mostrado na parte do meio ambiente, e as melhorias que podem ser feitas, pode-se concluir que é necessária uma ação expansível e concreta para a conscientização da população, e tentar mudar a cabeça daqueles que não colaboram por opção própria. O programa é só uma de pequenas boas ideias que podem surgir, basta apenas que essa ação maior ajude a por a prova a mentalidade dos que não cooperam para o desenvolvimento sustentável do Planeta.

## REFERÊNCIAS

REDAÇÃO. **O que é API?** 2016. Disponível em: <<https://canaltech.com.br/o-que-e/software/o-que-e-api/>>. Acesso em: 26 nov. 2016.

NUNES, Lucas (Comp.). **Biblioteca “Windows.h” o que me permite realizar?** 2014. Disponível em: <<http://pt.stackoverflow.com/questions/16495/biblioteca-windows-h-o-que-me-permite-realizar>>. Acesso em: 26 nov. 2016.

ANASTACIO, Bruno. **Linguagem C**. 2011. Disponível em: <<https://blogdecodigo.wordpress.com/2011/07/12/linguagem-c-bibliotecas-funcoes-main-printf-scanf-e-mais-drops-i/>>. Acesso em: 26 nov. 2016.

CASAVELLA, Eduardo. **O comando break em C**. Disponível em: <<http://linguagemc.com.br/o-comando-break/>>. Acesso em: 27 nov. 2016.

EDDYE. **C – Lendo de e para strings com sscanf/sprintf**. 2009. Disponível em: <<https://processolinux.wordpress.com/2009/11/26/c-lendo-de-e-para-strings-com-sscanf-sprintf/>>. Acesso em: 27 nov. 2016.