



EDUARDO OLIVEIRA GOMES - 10A
MARCO ANTÔNIO MARTINS RIBEIRO DE JESUS - 10A
VITOR MELO ASSUNÇÃO - 10A

GAC 104 - COMPUTAÇÃO GRÁFICA - 2022/2
DOCUMENTAÇÃO DO PROJETO FINAL

LAVRAS – MG
2022

1. INTRODUÇÃO

Na disciplina de Computação Gráfica, de acordo com a ementa disponibilizada pelo curso, devemos aprender tópicos importantes para uma formação básica no tema. São eles:

- Representação vetorial e matricial
- Transformações geométricas
- Algoritmos de projeção
- Determinação de superfícies visíveis
- Representação de cores
- Métodos de iluminação
- Algoritmos de conversão matricial de primitivas gráficas
- Técnicas anti-serrilhado e texturização

Para aplicar tais conhecimentos de forma prática, é sugerida a utilização da ferramenta OpenGL como técnica de ensino. Para a persistência de tais conhecimentos adquiridos ao decorrer do curso faz-se necessário a implementação de um projeto prático a fim de consolidar e demonstrar o mesmo. O objetivo deste projeto se dá na criação de um mini game descrito na atividade de proposta de projeto.

O mini game proposto se trata de um jogo no qual é possível misturar substâncias e compostos químicos a fim de gerar reações e criar novos compostos a partir dos disponibilizados inicialmente e dos que podem ser adicionados a um arquivo de configuração que permitirá novas combinações. A dinâmica do jogo se faz presente na liberdade de criação do usuário, podendo o mesmo selecionar os líquidos e compostos que se deseja misturar, além de como já dito anteriormente a possibilidade de criar novas combinações até então não propostas pelo jogo.

O jogo não possui um objetivo de conclusão em si (início, meio e fim), como normalmente são os jogos, o intuito do mesmo se encontra na possibilidade de aprender um pouco sobre química e suas reações em uma representação gráfica, sendo ela fiel às leis e resultados observados no mundo real. Sendo o mesmo de jogabilidade single player e tendo como controle o mouse.

A estética do jogo é simples, focando mais nas reações e produtos que podem ser gerados com os compostos. São dois frascos cilíndricos, sendo o primeiro em uma planície, onde ocorre as reações químicas e o composto é gerado, já o segundo possuindo interatividade com o usuário devido a possibilidade de seleção do mesmo por meio do mecanismo de entrada do mouse, além da possibilidade de selecionar a substância química que estará presente no segundo frasco através do clique nas substâncias presentes em uma estante, mudando então a substância que poderá vir a ser misturada. O visual do jogo se dá na

seleção do segundo frasco e no derrame do mesmo no líquido presente no primeiro frasco, logo em seguida uma reação química ocorre, gerando então um novo composto químico, no caso de ambos os compostos reagirem entre si.

Como dito anteriormente, as reações iniciais já são predefinidas, mas também contamos com a feature que permite a possibilidade de criação de novos compostos e substâncias químicas o que pode gerar novas reações, a partir da configuração do arquivo de reações, sendo necessário as entradas do que é o composto adicionado e quais as possibilidades de reações ele possuirá com os elementos pré existentes e os demais adicionados pelo usuário.

2. REQUISITOS

2.1 Requisitos Funcionais (RF)

Descrição de como as funcionalidade do sistema devem se comportar.

RF 001	FECHAMENTO DA JANELA
DESCRIÇÃO	Ao pressionar a tecla “ESC” o fechamento da janela acontece.
MUTÁVEL	NÃO
IMPORTÂNCIA	ALTA

RF 002	PAUSAR ANIMAÇÃO *
DESCRIÇÃO	Ao pressionar a tecla “SPACE” a animação é pausada.
MUTÁVEL	NÃO
IMPORTÂNCIA	MÉDIA

RF 003	SELEÇÃO DO FRASCO
---------------	--------------------------

DESCRIÇÃO	Ao pressionar o botão direito do mouse em cima do frasco que não está na planície ocorre o início da animação de mistura.
MUTÁVEL	NÃO
IMPORTÂNCIA	ALTA

RF 004	SELEÇÃO DO CONTEÚDO DO FRASCO
DESCRIÇÃO	Ao clicar na substância presente na estante do fundo altera a substância presente no frasco clicável.
MUTÁVEL	SIM
IMPORTÂNCIA	ALTA

RF 005	CRIAÇÃO DE NOVAS SUBSTÂNCIAS
DESCRIÇÃO	Ao alterar o arquivo de reações será possível criar novas substâncias e novos compostos a serem misturadas e sofrerem reações.
MUTÁVEL	NÃO
IMPORTÂNCIA	ALTA

RF 006	CRIAÇÃO DE NOVAS REAÇÕES
DESCRIÇÃO	Ao alterar o arquivo de reações será possível criar novas reações químicas com as substâncias já presentes ou às anteriormente adicionadas
MUTÁVEL	NÃO

IMPORTÂNCIA	ALTA
--------------------	-------------

RF 007	REVERTER REAÇÃO QUÍMICA
DESCRIÇÃO	Ao clicar na tecla “r” no teclado a última reação química será desfeita, se limitando apenas a última, não sendo a tecla recursiva.
MUTÁVEL	NÃO
IMPORTÂNCIA	ALTA

RF 008	RETIRAR SUBSTÂNCIA DO 1 FRASCO
DESCRIÇÃO	Ao pressionar a tecla “1” do teclado a substância do frasco ,onde as reações ocorrem, irá sumir.
MUTÁVEL	NÃO
IMPORTÂNCIA	ALTA

RF 009	RETIRAR SUBSTÂNCIA DO 2 FRASCO
DESCRIÇÃO	Ao pressionar a tecla “2” do teclado a substância presente no frasco que joga os compostos irá sumir.
MUTÁVEL	NÃO
IMPORTÂNCIA	ALTA

RF 010	EXCLUIR SUBSTÂNCIA CRIADA
---------------	----------------------------------

DESCRIÇÃO	Ao apagar a substância criada e descrita no arquivo de reações a mesma irá sumir da estante.
MUTÁVEL	NÃO
IMPORTÂNCIA	ALTA

RF 011	EXCLUIR REAÇÃO CRIADA
DESCRIÇÃO	Ao apagar a reação descrita no arquivo de reações a mesma não irá mais acontecer.
MUTÁVEL	NÃO
IMPORTÂNCIA	ALTA

2.2 Requisitos não funcionais

Descrição da qualidade da aplicação e características gerais presentes no sistema.

RF 001	AMBIENTE
DESCRIÇÃO	O desenvolvimento de ambiente 2D ou 3D em OpenGL.
IMPORTÂNCIA	ALTA

RF 002	CORES
DESCRIÇÃO	Deve conter colorização diferente para objetos e substâncias distintas.
IMPORTÂNCIA	ALTA

RF 003	INTERAÇÃO
DESCRIÇÃO	Deve conter interatividade com teclado e mouse.
IMPORTÂNCIA	ALTA

RF 004	ANIMAÇÃO
DESCRIÇÃO	Conterá a animação do frasco a ser virado e também das reações durante as possíveis trocas de cores que podem ocorrer.
IMPORTÂNCIA	ALTA

RF 005	TEXTURA
DESCRIÇÃO	Conterá textura em objetos presentes
IMPORTÂNCIA	ALTA

RF 006	SOFTWARE (LINUX)
DESCRIÇÃO	O programa deve funcionar em qualquer ambiente com sistema operacional Linux com a biblioteca GLUT instalada.
IMPORTÂNCIA	ALTA

RF 007	SOFTWARE (WINDOWS)
DESCRIÇÃO	Em casos de sistema operacional Windows, deve conter o WSL instalado com uma distro Linux e com a biblioteca GLUT instalada.
IMPORTÂNCIA	ALTA

RF 008	HARDWARE (MONITOR)
DESCRIÇÃO	O computador deve conter um monitor com resolução mínima de 1280x720.
IMPORTÂNCIA	ALTA

RF 009	HARDWARE (MOUSE)
DESCRIÇÃO	O computador deverá ter como dispositivo de entrada um mouse para que as operações com cliques sejam possíveis
IMPORTÂNCIA	ALTA

RF 010	HARDWARE (TECLADO)
DESCRIÇÃO	O computador deverá conter como dispositivo de entrada um teclado com as teclas que contém as funcionalidades descritas funcionando.
IMPORTÂNCIA	ALTA

3. CODIFICAÇÃO

Toda biblioteca possui muitas funções e métodos prontos que são disponibilizados no intuito de reduzir o retrabalho, padronizar códigos e facilitar a introdução de outros desenvolvedores em um projeto já em andamento ou na atualização de código legado, na biblioteca **GLUT** não é diferente, a estruturação de objetos já anteriormente idealizados, assim como texturas, iluminação e padrões, podendo ser facilmente identificadas as funções pertencentes a essa biblioteca pelo prefixo **glut** antes de qualquer função pertencente a mesma. Ao longo da disciplina, foram utilizados uma boa parte dessas funções e foram também utilizados para o desenvolvimento do nosso projeto.

Algumas das principais funções da GLUT incluem:

1. Criação de janelas: A GLUT permite criar janelas de desenho com facilidade e oferece uma interface para definir o tamanho da janela, o título e o tipo (janela normal ou tela cheia).
2. Gerenciamento de eventos de entrada: A GLUT fornece funções para gerenciar eventos de entrada, como teclado e mouse, para que você possa detectar interações com o usuário e reagir a elas.
3. Loop de eventos: A GLUT oferece um loop de eventos para o aplicativo, o que significa que ele é responsável por detectar eventos de entrada, atualizar a tela e garantir que o aplicativo seja executado de maneira suave e eficiente.
4. Callbacks: A GLUT oferece funções para definir callbacks, que são funções executadas automaticamente quando um evento ocorre, como o redimensionamento da janela ou a movimentação do mouse.
5. Desenho de figuras primitivas: A GLUT fornece funções para desenhar primitivas básicas, como linhas, círculos e retângulos, para que você possa começar a desenhar rapidamente sem precisar escrever muito código.
6. Interação com o usuário: a GLUT fornece funções para criar menus e diálogos de mensagem, permitindo que os usuários interajam com o aplicativo.
7. Suporte a animações: a GLUT fornece suporte para animações através de funções de atualização de tela e redisplay

As transformações geométricas são essenciais para a construção de um programa interativo e animado em OpenGL. Por meio dos métodos **glRotatef()**, **glTranslatef()** e o **glScalef()** é possível aplicar rotações, translações e operações de escala. Apoiados por variáveis de controle que são atualizadas nas funções de interação por teclado e mouse como por exemplo **glutKeyboardFunc()** e **glutMouseFunc()**, estes métodos permitiram as animações e seleção de itens implementados no projeto, além do suporte às operações para renderizar a cena.

Duas importantes funções do OpenGL que são responsáveis por empilhar e desempenhar matrizes de desenho são as **glPushMatrix()** e **glPopMatrix()**. Essas chamadas permitem isolar transformações geométricas. E para a implementação das cores dos objetos será utilizado o método de definição de cor **glColor3f()**, que recebe três parâmetros do tipo float que integra a cor no formato RGB.

Funções de Desenhos de Figuras Primitivas para criarmos os frascos os quais as reações químicas reagem utilizamos de funções de criação de cilindros, com a possibilidade de passagem de parâmetros como o tamanho da base, do topo, altura entre outros, tendo a função o seguinte escopo:

```
gluCylinder (GLUquadric* quad, GLdouble base, GLdouble top,  
             GLdouble height, GLint slices, GLint stacks);
```

Para concretizar criação dos frascos foi utilizado de funções criadoras de discos, sendo então implementado um disco no topo e um disco na base do cilindro, imitando então um frasco, sendo os parâmetros definidos de acordo com a base e o topo do respectivo cilindro o qual ela servirá de base ou topo. Sendo o escopo da função de criação de disco:

```
gluDisk (GLUquadric* quad, GLdouble inner, GLdouble outer,  
         GLint slices, GLint loops);
```

Para a simulação do líquido contido nos frascos foi utilizado de técnicas de textura, foram criadas texturas lisas e reflexivas, imitando então os possíveis líquidos e substâncias que podem vir a ser criadas dado as misturas que podem vir a ser realizadas. Além da utilização de texturas para simular os elementos químicos já existentes além dos que podem vir a ser criados no arquivo de reações. Sendo então idealizado da seguinte forma:

```

void display() {
    glClear(GL_COLOR_BUFFER_BIT | GL_DEPTH_BUFFER_BIT);

    // Desenhar o copo
    GLUQuadraticObj *quadratic = gluNewQuadric();
    gluQuadricNormals(quadratic, GLU_SMOOTH);
    gluCylinder(quadratic, 0.3, 0.3, 0.5, 32, 32);

    // Desenhar a base do copo
    glPushMatrix();
    glTranslatef(0.0f, 0.0f, -0.25f);
    gluDisk(quadratic, 0.0, 0.3, 32, 32);
    glPopMatrix();

    // Desenhar o líquido no copo
    glPushMatrix();
    glTranslatef(0.0f, 0.0f, 0.1f);
    gluDisk(quadratic, 0.0, 0.3, 32, 32);
    glPopMatrix();

    // Aplicar reflexão no líquido
    glPushMatrix();
    glScalef(1.0f, -1.0f, 1.0f);
    glTranslatef(0.0f, 0.0f, -0.2f);
    gluDisk(quadratic, 0.0, 0.3, 32, 32);
    glPopMatrix();

    glutSwapBuffers();
}

```

Legenda: protótipo do objeto de um copo com um líquido

Seguindo a mesma ideologia do protótipo apresentado, objetos semelhantes foram criados para criar a estante e os objetos contidos na mesma, além de é claro o outro copo que irá jogar as substâncias no copo presente na base.

Além disso, como já citado anteriormente, as funções “glutKeyboardFunc” e “glutMouseFunc” irão auxiliar em toda a interatividade do mini-game cumprindo todos os requisitos funcionais previamente estabelecidos, como voltar reações, pausar animações, retirar o líquido e seleção de novos líquidos através de cliques com o mouse.

Quanto a animação que pode ou não ocorrer em determinadas reações químicas, serão chamadas recursivas de novas renderizações de telas por meio da função de **glutDisplayFunc()** chamando um nova cena, onde o líquido pode mudar de cor de forma súbita ou de forma morosa, de um modo que seja possível observar e ter uma ideia de como seriam algumas reações.

4.RESULTADOS ESPERADOS

4.1 Inspiração

O sistema de Frascos Cilíndricos tem como principal fonte de inspiração o jogo Bartender The Right Mix. Nesse jogo, o suposto Bartender recebe comandos do usuário, que monta um drink a partir da prateleira de bebidas fazendo combinações de líquidos e, assim que o Bartender faz a mistura para gerar o drink, ele pode obter uma reação positiva ou negativa, que, em alguns casos, pode resultar até em explosão ou enjoo.



Legenda: jogo Bartender The Right Mix (inspiração do projeto)

Pensando nisso, tivemos a ideia de desenvolver um sistema mais educacional, cujo a proposta seria baseada no Bartender The Right Mix, porém, seria focado em gerar reações químicas alcançadas por meio da mistura de elementos químicos escolhidos pelo usuário.

4.1 O que esperamos visualmente

O sistema deve ser limpo, fluido, e, que consiga trazer a experiência de ensinar e demonstrar algumas reações químicas.

Deve conter nele, uma ideia muito semelhante ao que se vê no jogo que foi inspiração do projeto. Terá uma prateleira com os componentes a serem utilizados na mistura, como ilustrado na imagem a seguir:



Legenda: ilustração da prateleira com os componentes químicos

E, também, na frente da prateleira deve ter uma bancada, que, nela, serão realizadas as misturas dos componentes a fim de gerar as reações químicas esperadas, como é feito na imagem a seguir:



4.2 Ambiente

Inicialmente, é preciso instalar as dependências necessárias para preparar o ambiente de desenvolvimento. A configuração é feita no WSL devido à questão de desenvolvimento inconsistente em ambientes separados do Windows e Linux. O WSL é utilizado para evitar esse tipo de problema, pois simula um ambiente Linux dentro do sistema operacional Windows, permitindo que todos da equipe e usuários Linux possam executar o programa com os requisitos de hardware e software definidos.

O WSL inicialmente suporta apenas o terminal, mas é possível instalar suporte para aplicativos de GUI do Linux com configurações adicionais. É necessário ter o Windows 11 Build 22000 ou posterior e o driver de vGPU instalado para executar aplicativos de GUI do Linux. É preciso instalar o driver de GPU correspondente ao seu sistema (Intel, AMD ou NVIDIA). Para instalar o WSL 2, basta escolher a distribuição desejada e instalá-la diretamente na Microsoft Store ou via terminal.

Espera-se que o usuário siga os seguintes passos para instalação e configuração do ambiente de reprodução do programa:

Passo 1: Instalar o WSL 2

- Pode ser instalado diretamente da Microsoft Store ou via terminal;
- Basta escolher a distribuição de preferência e fazer a instalação;

Pelo PowerShell no modo **administrador** digite os comandos:

- `wsl --install`
- `wsl --set-default-version 2`
- `wsl -l -v`
- Aqui deve aparecer uma imagem Ubuntu, caso não apareça, você precisa baixar, digite `wsl --install -d Ubuntu`
- `wsl --setdefault <Nome_da_distro_Linux>`

Existe a opção de instalar o Ubuntu pelo Microsoft Store, caso não funcione, portanto, lembre-se de alterar a distribuição do WSL.

- Configurar o uso do WSL:
- No seu diretório `C:/Users/nome_usuario/` crie um arquivo chamado `.wslconfig`
- Configure o arquivo da seguinte maneira:

[wsl2]

memory=4GB # Limits VM memory in WSL 2

processors=1 # Makes the WSL 2 VM use 4 virtual processors

localhostForwarding=true # Boolean specifying if ports bound to wildcard or localhost in the WSL 2 VM should be connectable from the host via localhost:port.

Passo 2: Instalar aplicativos X11

- X11 é o sistema de janelas do Linux, incluindo uma variedade de aplicativos e ferramentas, tais como xclock, xcalc, xclipboard e xev. Mais detalhes podem ser encontrados nos documentos da x.org. Para usar, basta abrir o terminal do WSL já instalado, e executar o seguinte comando:

```
sudo apt install x11-apps -y
```

É possível testar se deu certo, executando algumas ferramentas nativas do Linux.

Passo 3: Configurar variável DISPLAY

- No terminal do WSL abra o arquivo de configuração do seu shell. ex .bashrc, .zshrc...

```
$ nano .zshrc
```

- Adicione o export abaixo:

```
GNU nano 6.2 .zshrc *  
export DISPLAY=$(grep -m 1 nameserver /etc/resolv.conf | awk '{print$2}':0.0
```

4.3 Resultados a serem alcançados

O sistema de Frascos Cilíndricos deve ser uma ferramenta interativa e intuitiva que permite ao usuário controlar e criar soluções químicas de forma virtual.

Com dois frascos cilíndricos disponíveis para uso, o usuário pode selecionar o frasco na estante no fundo e controlar o conteúdo do frasco escolhido.

4.4 Funcionamento do Sistema

Ao clicar no frasco selecionado, o frasco deve se levantar e começar a derramar líquido para o outro frasco. A solução criada deve ser baseada na lista de reações pré-estabelecidas e mudar de cor e composição de acordo com essas reações. As novas reações devem poder ser criadas pelo usuário ao editar o arquivo de reações.

4.5 Controles do Usuário

O usuário tem o controle total sobre o frasco selecionado, podendo clicar nele com o mouse para fazê-lo derramar líquido para o outro frasco. Além disso, a solução criada no outro frasco muda de cor e composição de acordo com uma lista de reações previamente estabelecida. No entanto, o usuário também tem a capacidade de criar novas reações ao editar o arquivo de reações.

4.6 Interação do Usuário com o Sistema

A interação do usuário com o sistema é fácil e intuitiva. O usuário pode selecionar o frasco desejado na estante no fundo, clicar nele com o mouse para fazê-lo derramar líquido para o outro frasco, e ver a solução criada mudar de cor e composição de acordo com as reações estabelecidas ou criadas pelo usuário. Além disso, o usuário pode editar o arquivo de reações a qualquer momento para criar novas soluções químicas.

4.7 Câmera

A câmera deverá ser posicionada a uma distância fixa do da exibição da cena que compõe o projeto, e o usuário poderá movimentar para visualizar melhor o experimento, a fim de se atentar a reação gerada, além de poder voltar à posição inicial.

4.8 Iluminação

Para a iluminação esperamos uma iluminação base que possibilite ver cada cor dos elementos e uma luz que incida em cada uma dos recipientes que contenham elementos químicos, para que a textura possa ser realçada.

4.9 Texturas

Um dos recursos que esperamos implementar é a troca de texturas, como o projeto tem o objetivo de modelar de forma realista as reações químicas que podem vir a ocorrer, também temos o objetivo de representar a variedade de cores e padrões que estas substâncias possuem no meio físico, já que cada um tem a sua particularidade em específico, seja cor, opacidade, composição, diluição, espessura, etc. Contudo, em contraste com a sua forma física a textura e forma não representa nenhuma vantagem em relação às outras.

5. CONCLUSÃO

Em conclusão, o sistema em questão será composto por dois frascos de formato cilíndrico, sendo que um deles pode ser controlado pelo usuário. O conteúdo do frasco escolhido pelo usuário pode ser trocado na estante no fundo, e clicando com o mouse no frasco, ele começará a derramar seu conteúdo para outro frasco. Neste outro frasco, uma solução será criada e sua cor e composição mudarão de acordo com uma lista de reações. Além disso, o usuário poderá criar novos compostos químicos que podem vir a gerar novas reações, que por sua vez também necessitam que sejam editadas o arquivo de reações. Em resumo, esse sistema permite ao usuário controlar e simular reações químicas com substâncias que não se possui acesso, oferecendo possibilidades para a aprendizagem e descoberta de novas reações.

Para representar o conteúdo dos frascos, é necessário criar objetos que representem diferentes tipos de líquidos, e alterar a cor e a textura do líquido de acordo com uma lista de reações. A animação do derramamento do líquido também é importante para tornar a simulação mais realista, além de é claro o processo entre os reagentes, de forma que mudanças de cores rápidas ou morosas e de texturas simulem como seria o acontecido.

A lista de reações químicas pode ser editada pelo usuário, permitindo a criação de novas reações, substâncias e a personalização da simulação. Essa flexibilidade torna o projeto mais interessante e educacional, permitindo ao usuário explorar diferentes reações químicas e suas consequências.

Em resumo, para desenvolver esse projeto é importante ter conhecimento de desenho em OpenGL, animações e interações com o usuário, além de estruturas de dados para representar objetos e listas de reações químicas. Ao seguir essas estratégias, é possível criar uma simulação interativa e educacional de reações químicas em OpenGL.

Dessa forma, o tema escolhido é adequado para o trabalho e fala sobre outros assuntos abordados em sala de aula, incluindo tópicos de Computação Gráfica e outras diversas disciplinas que oferecem conceitos complementares atrelados ao desenvolvimento desse projeto.

A realização deste trabalho permitiu a aplicação prática dos conhecimentos adquiridos ao longo da formação na disciplina, nos desafiando em uma área que até então era apenas teórica para a maioria de nós. Além de nos desafiar e incentivar a

criar um projeto do 0, não só pela parte da codificação mas também da criação do documento e de todas as suas etapas, além de, é claro, permitir a apresentação do mesmo.
