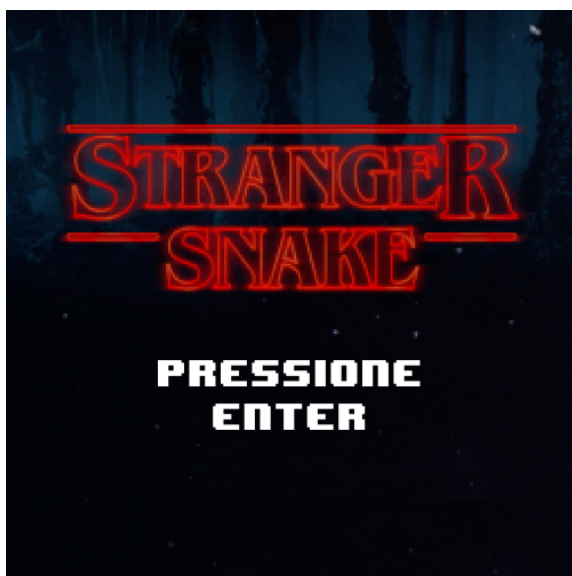


---

**Discentes:** Matheus Otavio Lisboa - 108190  
Pedro Henrique Lopes de Almeida - 108203  
Vitória Maria Ferreira de Oliveira - 108211

### **Relatório Snake Game**

Jogo tematizado no universo da série da Netflix Stranger Things, "Stranger Snake" permite que o usuário navegue entre a nostalgia do famoso jogo de arcade ambientado em uma das séries mais famosas da atualidade, de tal maneira a agradar públicos de todas as idades.



*Imagem 1: Tela inicial*



*Imagem 2: Frame do jogo*

Algumas das funcionalidades implementadas no snake game são:

#### **1. Frutinhas e inseto (Pedro)**

Assim como no original, nesta versão a snake se alimenta de cerejas e abelhas, que aumentam o tamanho de seu corpo em 1 e 3, respectivamente. Sendo a abelha uma funcionalidade bônus, que surge no mapa com menos frequência em comparação à cereja.



Imagem 3: Ilustração da cereja e da abelha

Entendendo o código:

- Declaração das variáveis da fruta e abelha(especial) no allegro:

```
55 ALLEGRO_BITMAP *fruta = NULL;
56 ALLEGRO_BITMAP *fruta = NULL; //cria a frutiha
57 ALLEGRO_BITMAP *especial = NULL; //cria o inseto
58
```

- Desenhando a cereja por meio de carregamento de imagem:

```
121 fruta = al_load_bitmap("cereja.tga");
122 if(!fruta)
123 {
124     al_destroy_display(display);
125     al_destroy_timer(timer);
126     return -1;
127 }
```

- Desenhando a abelha:

```
376 especial = al_load_bitmap ( "abelha.tga" ); //cria o inseto
377 al_draw_bitmap (especial,jespecial*q,iespecial*q, 0 );
378 if (!jespecial || !iespecial || !especial)
```

Durante o percorrer da snake na malha do jogo são oferecidas possibilidades de frutas(cerejas), de maneira aleatória, para que o corpo da cobrinha aumente junto à pontuação do usuário. Assim, quanto mais frutinhas forem coletadas a pontuação aumenta. Além disso, existe a possibilidade de especiais(abelhas) surgirem no decorrer do jogo, essas abelhas possuem uma pontuação maior em comparação às cerejas, o triplo.

```
if(comeu){
    pontuacao++; //aumenta a pontuação
    corpo++; //aumenta o tamanho da cobra
    bool teste=true;
    int k;
    while(teste){
        teste=false;
        ifruta=(rand()%23)+1; //gera posicao aleatoria para fruta
        jfruta=(rand()%23)+1;

        for(k=cont*2+1;k>=1;k=k-2){
            if(ifruta==vet[k]&&jfruta==vet[k-1]){ //posicao da fruta == posicao da cobrinha
                teste=true;
            }
        }
    }
}
```

No código acima é detalhado o que ocorre quando a cobra come uma fruta. Assim, a pontuação aumenta e o corpo também no contador. Após isso, é feito um teste para uma nova posição da fruta para verificar a posição de uma nova fruta em comparação a atual posição da cobrinha.

```

if (contabelha==99){//quando o contador atinge o valor de 99 que ocorre quando o jogo da 99 "vo
    bool teste=true;
    while (teste){
        teste=false;
        iabelha=(rand()%23)+1; // gera posicao aleatória para a abelha
        jabelha=(rand()%23)+1;
        for (int k=cont*2+1;k>=1;k--2) //mesmo caso da cereja
            if((iabelha==vet[k]&&jabelha==vet[k-1])||(iabelha==ifruta && jabelha==jfruta))
                //caso teste para a mesma posicao da fruta e inseto
                teste=true;
        }
        contabelha++; //contador para o inseto
    }
    else if (contabelha== 100 ){ //quando o contador atinge o valor de 100 que ocorre quando o jogo
        abelha = al_load_bitmap ( "abelha.tga" ); //desenha a abelha
        al_draw_bitmap (abelha,jabelha*q,iabelha*q, 0 );//desenha a abelha na malha do jogo
        if (j==jabelha&&i==iabelha){ //se a posicao da cabeca da cobra for a mesma da abelha
            al_play_sample(bonus, 1.0, 0.0, 1.0, ALLEGRO_PLAYMODE_ONCE, NULL);
            pontuacao=pontuacao + 50; //cada inseto equivale a 5 frutas
            corpo+=3 ;//o corpo da cobra aumenta 3 unidades
            velocidade *= 10;//caso coma uma abelha a velocidade aumenta 10x
            contabelha=0;//zera o contador das abelhas
        }
    }
}

```

No caso dos insetos ocorre o mesmo processo que acontece quando uma frutinha, mas existe a diferença de que um inseto equivale a 3 frutinhas, portanto a pontuação se torna ainda maior e o corpo da snake aumenta de maneira proporcional. Além disso, para cada abelha coletada a velocidade do jogo aumenta. No caso das abelhas existe um contador para o inseto que é acrescido de 1 a cada vez que o código é passado, assim quando o código é passado 99 vezes uma posição para da abelha é criada e na 100 é desenhada no display. Logo após, esse mesmo contador é zerado para que esse caso inicie novamente, como um looping.

## 2. Game Over (Matheus, Pedro e Vitória)

Existem duas maneiras da snake morrer durante o jogo: encostar no próprio corpo e encostar na parede.

### 2.1. Encostar no próprio corpo (Matheus)

```

if (compara>contpassou){ //a cobrinha encostou no seu proprio corpo, se a
    al_destroy_audio_stream (musica); //inicializacao para musica de game over
    al_play_sample(gameover, 1.0, 0.0, 1.0, ALLEGRO_PLAYMODE_ONCE, NULL);
    al_draw_bitmap(telafinal,0,0,0);
    al_flip_display();
    al_rest(3);
    sair = true;
    break;
}

compara=contpassou;
al_flip_display();

```

No caso da snake encostar no próprio corpo ocorre quando dada determinada quantidade de frutas coletadas, seu corpo aumenta de tal maneira que torna possível a cabeça da cobra encostar no próprio corpo. No código acima é exibido o condicional que faz o jogo fechar quando isso ocorre, no qual compara é o tamanho do corpo da

cobra diminuída de 2 quantidades for maior que o tamanho atual da cobrinha ela encostou no próprio corpo.

## 2.2. Encostar na parede (Matheus)

```
if(cima && MAPA[i][j] == '1')
{
    sair = true;
    break;
}

if(baixo && MAPA[i+1][j] != '1')
{
    i++;
    posy = i*q;
}

if(baixo && MAPA[i][j] == '1')
{
    sair = true;
    break;
}

if(esq && MAPA[i][j-1] != '1')
{
    j--;
    posx = j*q;
}

if(esq && MAPA[i][j] == '1')
{
    sair = true;
    break;
}
```

No código acima é feito uma comparação para cada direção que a snake for, deste modo, se a posição dela for igual a um dos quadrados da parede o jogo é automaticamente fechado.

## 3. Movimentação da snake (Pedro)

```
switch(ev.keyboard.keycode)
{
    case ALLEGRO_KEY_UP: //setinha pra cima
        if(!baixo)
        {
            cima = true;
            baixo = false;
            esq = false;
            dir = false;
        }
        else
        {
            cima=false;
            baixo=true; //nao deixa o corpo voltar
            esq=false;
            dir=false;
        }
        break;

    case ALLEGRO_KEY_DOWN: //setinha pra baixo
        if(!cima)
        {
            cima = false;
            baixo = true;
            esq = false;
            dir = false;
        }
        else
        {
            cima = true;
            baixo = false;
            esq = false;
            dir = false;
        }
        break;
}
```

Para o usuário poder movimentar a snake no plano do jogo é preciso que o mesmo indique a direção que deseja encaminhar a cobrinha. Uma vez selecionada, por meio das teclas de seta do próprio teclado, a cobra mudará de direção somente se for pressionada outra tecla de direção. Para evitar que a cabeça da cobra fique indo e

voltando, como um movimento de pêndulo, foi criado um else para que isso não ocorra, que torna “false” todas as outras direções que não foram indicadas.

## 4. Adicionais (Vitória)

Como adicionais, o grupo propôs a inclusão de efeitos sonoros nas ações da snake, um placar, uma tela inicial, junto a opção de pressionar uma tecla para iniciar o jogo e uma tela de game over.

### 4.1. Efeitos Sonoros

Os efeitos sonoros disponíveis no jogo incluem: música temática da série “Stranger Things”, sons de comida da cereja e um som característico de bônus caso a cobrinha coma alguma abelha, além do som de game over.

```
// instalando som
if (!al_install_audio())
{
    fprintf(stderr, "Falha ao inicializar áudio.\n");
    return -1;
}

if (!al_init_acodec_addon())
{
    fprintf(stderr, "Falha ao inicializar codecs de áudio.\n");
    return -1;
}

if (!al_reserve_samples(3))
{
    fprintf(stderr, "Falha ao alocar canais de áudio.\n");
    return -1;
}

comida = al_load_sample("gameover.wav");
```

No trecho de código demonstrado acima é feita a instalação da funcionalidade de áudio no Allegro.

```
gameover= al_load_sample("gameover.wav");

if (!gameover)
{
    fprintf(stderr, "Falha ao carregar gameover.\n");
    return -1;
}

bonus = al_load_sample("bonus.wav");

if (!bonus)
{
    fprintf(stderr, "Falha ao carregar bonus.\n");
    return -1;
}

musica = al_load_audio_stream("mus.ogg", 4, 1024);
if (!musica)
{
    fprintf(stderr, "Falha ao carregar audio.\n");
}
```

Já nesta parte do código é feita a declaração dos sons para cada situação que ocorra com a cobra durante o jogo.

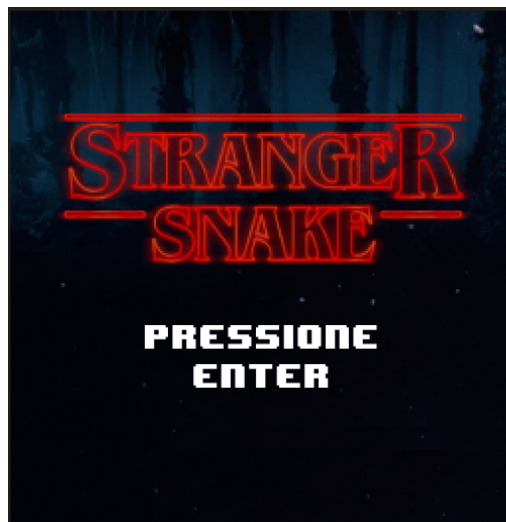
## 4.2. Placar

Para cada frutinha ou inseto que a snake comeu uma pontuação é atribuída ao score do jogador, de forma dinâmica e em tempo real ao jogo.

```
scoretotal = score + pontuacao;  
texto = "Score " + std::to_string(scoretotal);  
al_draw_text(font20, al_map_rgb(4,250, 196), 250, 480, ALLEGRO_ALIGN_CENTER ,texto.c_str());
```

No trecho acima é demonstrado como ocorre a exibição do “score” do jogador, coletando informações de pontuação já explicitadas acima.

## 4.3. Telas Iniciais e finais



Ao executar o jogo é exibido ao usuário uma tela inicial interativa ao pressionamento da tecla “ENTER” do teclado para iniciar o game.

```
if (ev.type == ALLEGRO_EVENT_KEY_DOWN) {  
    if (ev.keyboard.keycode == ALLEGRO_KEY_R || ev.keyboard.keycode == ALLEGRO_KEY_PAD_ENTER) {  
        intro = false;
```

Na parte do código acima é demonstrado o que ocorre quando a tecla é pressionada e o jogo iniciado.

```
if(compara>contpassou){ //a cobrinha encostou no seu proprio corpo, se a  
    al_destroy_audio_stream (musica); //inicializacao para musica de game over  
    al_play_sample(gameover, 1.0, 0.0, 1.0, ALLEGRO_PLAYMODE_ONCE, NULL);  
    al_draw_bitmap(telafinal,0,0,0);  
    al_flip_display();  
    al_rest(3);  
    sair = true;  
    break;  
}
```

No exemplo da cobrinha encostar no corpo é detalhado como os efeitos sonoros são inicializados e a tela de game over é exibida.