

## Desenvolvimento de um Programa com Dinâmica de Jogo

### JOGO "Contrabandista Espacial"

#### A Ideia e o Objetivo do Jogo

"Contrabandista Espacial" é um jogo de simulação e risco. O jogador assume o papel de um piloto de uma nave cargueira tentando lucrar transportando mercadorias (algumas nem tão legais assim) por uma perigosa rota espacial.

**Objetivo:** O objetivo principal é sobreviver à jornada por 5 setores espaciais e terminar com a maior quantidade de créditos possível. O jogador perde se a integridade do casco da sua nave chegar a zero ou se seus créditos se esgotarem. Cada viagem é única, pois os eventos encontrados são aleatórios.

#### Como Funciona (Regras e Interações)

O jogo é baseado em turnos (ou "setores"), onde a cada turno um evento aleatório acontece.

- **Estado Inicial:** O jogador começa com:
  - **1000 Créditos:** A moeda do jogo, usada para subornos e reparos.
  - **100 de Integridade do Casco:** Os "pontos de vida" da nave.
- **A Jornada:** A viagem tem um total de **5 setores**. O jogo avança em um loop, um setor por vez.
- **Eventos Aleatórios:** No início de cada setor, um de três eventos possíveis é sorteado:
  1. **Carga à Deriva:** O jogador encontra uma carga valiosa flutuando no espaço. É uma oportunidade de lucro sem riscos.
  2. **Ataque de Piratas Espaciais:** Uma nave pirata exige seus créditos. O jogador tem duas opções:
    - **Lutar:** Enfrentar os piratas. O resultado (vitória ou derrota) é decidido aleatoriamente. Vencer pode gerar lucro, mas perder causa um dano severo ao casco.
    - **Pagar:** Entregar uma quantia de créditos para evitar o confronto.
  3. **Inspeção da Patrulha Galáctica:** Uma patrulha policial ordena que o jogador pare para uma inspeção de carga. As opções são:
    - **Tentar Persuadir:** Usar a lábia para evitar a multa. Há uma chance de sucesso e uma chance de falha, que resulta em uma multa pesada.
    - **Subornar:** Pagar uma quantia menor para que o guarda "faça vista grossa".
- **Condições de Fim de Jogo:**
  - **Vitória:** O jogador completa a travessia dos 5 setores com créditos e integridade do casco acima de zero. A pontuação final são os créditos restantes.

- **Derrota:** O jogo termina imediatamente se a **integridade do casco** chegar a **0 ou menos**, ou se os **créditos chegarem a 0 ou menos**.

---

### 3. Código Fonte do Jogo

O código abaixo foi escrito em C. Ele é amplamente comentado para explicar cada passo da lógica.

```
////////////////////////////////////
```

```
// Online C compiler to run C program online
```

```
#include <stdio.h>
```

```
#include <time.h>
```

```
#include <stdlib.h>
```

```
int main() {
```

```
    // --- Declaração de Variáveis ---
```

```
    int credits_jogador;
```

```
    int integridade_casco;
```

```
    int setor_atual;
```

```
    int total_setores;
```

```
    int evento_aleatorio; // Armazena o número do evento sorteado
```

```
    int chance_sucesso; // Usado para determinar o resultado de ações
```

```
    int escolha_jogador; // Armazena a decisão do jogador (1 ou 2)
```

```
    // --- Inicialização do Jogo ---
```

```
    // A função 'srand(time(NULL))' usa o tempo atual como "semente" para o gerador
```

```
    // de números aleatórios. Isso garante que os eventos sejam diferentes a cada
```

```
    // vez que o jogo é executado.
```

```

srand(time(NULL));

creditos_jogador = 1000;
integridade_casco = 100;
setor_atual = 1;
total_setores = 5;

// --- Mensagem de Boas-Vindas e Contexto ---

printf("=====\n");
printf("===  CONTRABANDISTA ESPACIAL  ===\n");
printf("=====\n\n");
printf("Voce e um piloto tentando a sorte na rota Kessel.\n");
printf("Sua missao: sobreviver a 5 setores e maximizar seus lucros.\n");
printf("Voce comeca com %d creditos e sua nave com %d%% de integridade.\n",
creditos_jogador, integridade_casco);
printf("Boa sorte, piloto!\n\n");
printf("Pressione ENTER para iniciar sua jornada...");
getchar(); // Espera o jogador pressionar Enter

// --- Laço Principal do Jogo (while) ---

// O jogo continua enquanto o jogador não tiver perdido (casco e créditos > 0)
// e enquanto não tiver chegado ao final da rota (setor_atual <= total_setores).
while (setor_atual <= total_setores && integridade_casco > 0 && creditos_jogador > 0) {

    // Mostra o status atual no início de cada setor
    printf("-----\n");
    printf("Entrando no Setor %d de %d...\n", setor_atual, total_setores);
    printf("Status: [Creditos: %d] - [Casco: %d%%]\n", creditos_jogador, integridade_casco);
    printf("-----\n\n");
}

```

```

// Sorteia um evento para o setor atual (1, 2 ou 3)

// A expressão 'rand() % 3' gera um número entre 0 e 2. Somamos 1 para ter o intervalo
[1, 3].

evento_aleatorio = (rand() % 3) + 1;


// --- Estrutura Condicional para Tratar os Eventos (if / else if / else) ---


// Evento 1: Carga à Deriva
if (evento_aleatorio == 1) {
    printf(">> EVENTO: Voce encontrou uma carga de pecas raras a deriva!\n");
    printf(">> Lucro de 250 creditos obtido.\n\n");
    creditos_jogador += 250; // Forma abreviada de: creditos_jogador = creditos_jogador +
250
}

// Evento 2: Ataque de Piratas
else if (evento_aleatorio == 2) {
    printf(">> EVENTO: Piratas Espaciais surgem e exigem 300 creditos!\n");
    printf(" 1 - Lutar contra os piratas.\n");
    printf(" 2 - Pagar 300 creditos para evitar o conflito.\n");
    printf("Qual a sua decisao? ");
    scanf("%d", &escolha_jogador);


    if (escolha_jogador == 1) { // Escolheu lutar
        printf("\n>> Voce decide lutar! Os canhoes sao acionados...\n");
        chance_sucesso = (rand() % 10) + 1; // Sorteia um número de 1 a 10

        if (chance_sucesso > 5) { // 50% de chance de ganhar (6, 7, 8, 9, 10)
            printf(">> VITORIA! Voce derrotou os piratas e saqueou 150 creditos dos
destrocos!\n\n");
            creditos_jogador += 150;
        } else {

```

```

        printf(">> DERROTA! Sua nave foi atingida! O casco perdeu 40 de
integridade!\n\n");

        integridade_casco -= 40; // Forma abreviada
    }
} else { // Escolheu pagar

    printf("\n>> Voce paga os piratas e eles partem em paz.\n\n");

    creditos_jogador -= 300;

}

}

// Evento 3: Inspeção da Patrulha
else {

    printf(">> EVENTO: A Patrulha Galactica ordena que voce pare para uma inspecao!\n");

    printf(" 1 - Tentar persuadir o guarda a te liberar.\n");

    printf(" 2 - Oferecer um 'incentivo' de 150 creditos (suborno).\n");

    printf("Qual a sua decisao? ");

    scanf("%d", &escolha_jogador);


    if (escolha_jogador == 1) { // Tentar persuadir

        printf("\n>> Voce tenta usar sua labia com o guarda...\n");

        chance_sucesso = (rand() % 10) + 1;


        if (chance_sucesso > 6) { // 40% de chance de sucesso (7, 8, 9, 10)

            printf(">> SUCESSO! O guarda acredita na sua historia e te libera!\n\n");

        } else {

            printf(">> FALHA! O guarda nao gostou da sua atitude e aplicou uma multa de 400
creditos!\n\n");

            creditos_jogador -= 400;

        }

    } else { // Escolheu subornar

        printf("\n>> Voce discretamente passa os creditos para o guarda, que sorri e te deseja
'boa viagem'.\n\n");
    }
}

```

```

        credits_jogador -= 150;
    }
}

// Pausa para o jogador ler o que aconteceu
printf("Pressione ENTER para continuar para o proximo setor...");
getchar();

// Avança para o próximo setor
setor_atual++;
}

// --- Fim do Jogo: Verificação de Vitória ou Derrota ---
printf("\n\n=====\\n");
printf("===      FIM DA JORNADA      ===\\n");
printf("=====\\n\\n");

if (integridade_casco <= 0) {
    printf("DERROTA: Sua nave foi destruida no espaco profundo.\\n");
    printf("Fim de jogo.\\n");
} else if (credits_jogador <= 0) {
    printf("DERROTA: Voce esta sem credits e nao pode continuar.\\n");
    printf("Fim de jogo.\\n");
} else {
    printf("VITORIA! Voce completou a rota com sucesso!\\n");
    printf("Pontuacao Final (Credits): %d\\n", credits_jogador);
    printf("Integridade do Casco Restante: %d%%\\n", integridade_casco);
}

printf("\\n\\n");

```

```
    return 0; // Indica que o programa terminou com sucesso  
}
```