Principais partes do código:

Inicialização da memória

```
void initializeMemory(int memSize, int fSize) {
    physicalMemorySize = memSize;
    frameSize = fSize;
    numFrames = physicalMemorySize / frameSize;
    physicalMemory = (unsigned char *)malloc(physicalMemorySize);
    memset(physicalMemory, 0, physicalMemorySize);

// Initialize the free frame list
    freeFrames = NULL;
    for (int i = numFrames - 1; i >= 0; i--) {
        FrameNode *node = (FrameNode *)malloc(sizeof(FrameNode));
        node->frameNumber = i;
        node->next = freeFrames;
        freeFrames = node;
    }
}
```

- configura o tamanho de memória e dos quadros
- aloca e inicializa a memória física
- cria a lista encadeada de quadros livres

Criação de um processo

```
void createProcess(int processID, int processSize) {
   if (processSize > maxProcessSize) {
      printf("Erro: Tamanho do processo excede o máximo
permitido.\n");
      return;
   }
...
```

- Verifica se o tamanho do processo é válido.
- Calcula o número de páginas necessárias.
- Inicializa a memória lógica com valores aleatórios.
- Aloca quadros e configura a tabela de páginas.
- Copia os dados da memória lógica para a memória física.
- Adiciona o processo à lista de processos.

Mostrar a memória e tabela de páginas

- Conta e exibe a porcentagem de memória livre.
- Exibe o conteúdo de cada quadro na memória física.
- Procura pelo processo e exibe sua tabela de páginas.
- Se o processo não for encontrado, exibe uma mensagem de erro.

Instruções para compilação

Via terminal:

- 1. Se não tiver o GCC instalado, realizar a instalação para o SO utilizado
 - a. sudo apt-get install gcc > Linux
 - b. brew install gcc > MacOS
 - c. Instalar o pacote MINGW > Windows
- 2. Compilar o programa utilizando o GCC com o seguinte comando: gcc main.c -o main
- 3. Executar o programa usando:
 - a. Linux: ./main
 - b. Windows: main.exe

Via IDE ou Compilador Online:

- 1. **IDE**: Rodar o arquivo main.c
- Compilador Online: Copiar o código do arquivo main.c > colar no compilador >
 executar. Testado no: Online C Compiler online editor (onlinegdb.com)

Casos de Teste executados:

Configuração inicial:

- Tamanho memória física: 512 bytes
- Tamanho Quadro/Página em bytes: 64 bytes
- Tamanho máximo de um processo: 128 bytes

```
> $ ./main
Digite o tamanho da memoria fisica (em bytes): 512
Digite o tamanho do quadro/pagina (em bytes): 64
Digite o tamanho maximo de um processo (em bytes): 128
```

Menu do programa:

Menu:

- 1. Visualizar memoria
- 2. Criar processo
- 3. Visualizar tabela de paginas
- 4. Sair

Escolha uma opcao:

Visualizar memória:

```
| Memória live: 100 00% | Memoria live: 100 00% | Memo
```

Criando um processo com tamanho menor que o máximo:

```
Escolha uma opcao: 2
Digite o ID do processo: 1
Digite o tamanho do processo (em bytes): 64
Processo 1 criado com sucesso.
```

Criando um processo com tamanho maior que o maximo:

```
Digite o ID do processo: 2
Digite o tamanho do processo (em bytes): 256
Erro: Tamanho do processo excede o máximo permitido.
```

Visualizar memória:

```
        Memorita livre:
        87.50%
        Frame
        0:
        74
        5C
        A1
        2D
        A9
        1C
        4C
        A7
        66
        F4
        80
        52
        EC
        DE
        94
        1C
        A1
        2D
        A9
        1C
        A1
        2D
        A9
        1C
        A1
        A0
        CE
        A7
        66
        F4
        80
        52
        EC
        DE
        50
        50
        6A
        B0
        A0
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00
        00<
```

Visualizar tabela de páginas de um processo existente:

```
Escolha uma opcao: 3
Digite o ID do processo: 1
Tabela de páginas do processo 1 (Tamanho: 64 bytes):
Página 0 -> Quadro 0
```

Visualizar tabela de páginas de um processo não existente:

```
Escolha uma opcao: 3
Digite o ID do processo: 5
Erro: Processo 5 não encontrado.
```

Criar um novo processo, passando o tamanho limite:

```
Escolha uma opcao: 2
Digite o ID do processo: 2
Digite o tamanho do processo (em bytes): 256
Processo 2 criado com sucesso.
```

Visualizar memória:

```
Frame 0: E9 E2 D0 SC FD C9 E6 A7 B6 C5 DD 9A D2 92 42 3A 8F C4 66 E9 25 8B F4 0E 2F 10 7D 3C 52 7B EE 57 F6 AD 44 45 F9 48 D3 D6 15 69 FA C8 93 2A C8 91 7C C8 98 C5 B1 7F 4E 05 F3 C2 D4 EA 7F 09 C6 3D Frame 1: 64 21 DF 84 9E F4 DF 96 25 BA 9A F2 F9 0C 59 DC C3 A1 23 2E 8D 71 44 5C B2 8B EE 23 40 DF BF CA A1 44 47 96 58 FB A7 7B EB E6 A1 FD A5 18 6C 6C 9A 7F 87 59 68 DA 57 41 E5 F2 71 25 AD 52 6E FA Frame 2: 3D 40 A1 CA 9B 2E 81 13 0A 40 36 52 10 BB 15 C6 48 5C 56 0B AB 96 CB E6 4C 73 E9 DD 0F EF D9 01 49 A3 A7 6A 70 0B 09 64 32 0D 70 AD D1 E4 4B A7 14 E2 79 84 CC 18 B9 57 F7 AF 12 F0 F7
```

Criar outro processo com tamanho de 256 bytes:

```
Escolha uma opcao: 2
Digite o ID do processo: 3
Digite o tamanho do processo (em bytes): 256
Erro: Memória insuficiente durante a alocação.
```