README

Author - Kshitij 2022256 , Roshan Kumar Mahto 2022418

MeMS - Memory Management System

Features
1. Initialize the MeMS system using `mems_init` before using any memory allocation functions.

2. Allocate memory using `mems_malloc` and deallocate it using `mems_free`.

3. To print memory usage statistics, use the `mems_print_stats` function.

4. Clean up the MeMS system using `mems_finish` at the end of your program.

Function Descriptions

- `void mems_init()`: Initializes the MeMS system. Call this function at the beginning of your program.

- `void mems_finish()`: Cleans up the MeMS system by releasing all allocated memory. Call this function at the end of your program.

- `void *mems_malloc(size_t size)`: Allocates memory of the specified size and returns a MeMS virtual address.

- `void mems_free(void *v_ptr)`: Frees memory pointed to by a MeMS virtual address, making it available for reuse.

- `void mems_print_stats()`: Prints memory usage statistics, including the number of pages utilized and unused memory.

- `void *mems_get(void *v_ptr)`: Returns the MeMS physical address mapped to a MeMS virtual address.

Example Usage

Here is an example of how to use MeMS:

```c
#include "mems.c"

int main() {
    mems_init();
```

```
    // Allocate memory
    void *mem_ptr = mems_malloc(1024);

    // Use the allocated memory

    // Free memory
    mems_free(mem_ptr);

    // Print memory usage statistics
    mems_print_stats();

    mems_finish();
    return 0;
}
```

Assumption:
While free memory allocation , the holes will only be merged if two holes are adjacent to each other . Code will stop if incorrect input is given.