EXPERIMENT 6

IPC USING SHARED MEMORY - SENDER & RECEIVER PROCESSES

AIM

To implement IPC using shared memory with sender and receiver processes.

ALGORITHM FOR SENDER PROCESS

STEP 0: START

STEP 1: The variables are declared
'id' - stores the identifier for shared memory segment
'sm' - represents a pointer to shared memory segment.
'buf' - A buffer to store input data.

STEP 2: shmget() function creates a new shared memory segment or gets the identifier of an existing one
'(key_t)1222' - Unique key used to identify the shared memory segment.
'1024' - size of shared memory (bytes)
'0666 | IPC_CREAT' - are the permissions for the shared memory segment (octal)
'IPC_CREAT' flag indicates that the shared memory is created if it doesn't already exist.

STEP 3: Print the id of shared memory segment.

STEP 4: shmat() function attaches the shared memory segment identified 'id' to addres space of calling process.
STEP 5: Print the address where the shared memory segment is attached in the process's address
STEP 6: Prompt the user to enter the data and read it into the buffer 'buf' using the read() function
STEP 7: copy the data from 'buf' to the shared memory pointed by 'sm' using strcpy() function.
STEP 8: Print the data that was written to the shared memory segment.

ALGORITHM FOR RECEIVER PROCESS
STEP 0: START
STEP 1: Vaiables are declared
'id' - Identifier for shared memory segment
'sm' - Pointer to the shared memory segment
'buf' - Buffer to store data read from shared memory
'a' & 'b' are variables to store parsed integers for addition.
STEP 2: shmget() function retrieves the identifier of the existing shared memory segment.
STEP 3: shmat() attaches the shared memory segment to the address space

of the calling process.
STEP 4: Print the address where the shared memory segment is attached in the process's address space.
STEP 5: Read data from shared memory into the buffer 'buf' using 'strcpy()' and prints the data read.
STEP 6: a = buf[0] - '0' and b = buf[2] - '0' extracts the integers from buffer and converts character digits to integers
STEP 7: Sum is printed after adding a+b.
STEP 8: END

RESULT
Programs successfully executed and output obtained.