C PROGRAM THAT TAKES, AS A COMMAND LINE ARGUMENT, THE NUMBER OF MEGABYTES OF MEMORY IT WILL USE AND DURING EXECUTION IT SHOULD CONSUME THAT MUCH MEMORY. OBSERVE MEMORY USAGE DURING PROGRAM EXECUTION USING FREE COMMAND.

```
  GNU nano 6.2
#include<stdio.h>
#include<stdlib.h>
#include<time.h>
#include<unistd.h>


int main(int argc, char* argv[]){
    printf("Current Process ID = %d\n",getpid());
    long long int size = ((long long int)atoi(argv[1]))*1024*1024; //in bytes
    int* buffer = (int*)malloc(size);

    //run the while loop for given amount of time.
    time_t endwait, seconds, start;
    seconds=atoi(argv[2]);
    start=time(NULL);
    endwait = start + seconds;

    while(start<endwait){
        printf(".");
        fflush(stdout);
        for(long long int i=0; i<size/sizeof(int); i++){
            buffer[i] = i;
        }
        start = time(NULL);
    }
    printf("(done)\n");
    return 0;
}
```

Before executing the program

```
ayushmaan@ASJ: ~/Music/6
Every 1.0s: free

               total        used        free      shared  buff/cache   available
Mem:         2023460      632352     1059160       17948      331948     1217844
Swap:        3287036      402096     2884940
```

AFTER EXECUTING THE PROGRAM

```
ayushmaan@ASJ: ~/Music/6
Every 1.0s: free

               total        used        free      shared  buff/cache   available
Mem:         2023460      647644     1074960       17816      300856     1202680
Swap:        3287036      422628     2864408
```

```
ayushmaan@ASJ:~/Music/6$ ./a.out 1000 20
Current Process ID = 4888
.................(done)
```