Question 2C

QUESTION 2C

1. Meaning of each of the column:

1. **Address** :- The starting address of the given mapping in the virtual address space of the process.

2. **Perm** :- The permissions that apply to this mapping.
   Example:
   r-read
   w- write
   x- execute
   p- private mapping
   s - shared
3. **Offset** :- Offset of the mapping within the file (In case the mappings are file based)

4. **Device** :- Device major/minor number

5. **Inode** :- File System inode number of the file to which the mapping refers to

6. **Size** :- Size of mapped address range in virtual address space(in kilobytes)

7. **Rss** :- Resident Set Size i.e. amount of memory that is currently in RAM (in kilobytes)

8. **Pss** :- **Proportional Set Size** (**PSS**) is the portion of main memory (RAM) occupied by a process and is composed by the private memory of that process plus the proportion of shared memory with one or more other processes. Unshared memory including the proportion of shared memory is reported as the PSS.

9. **Refernced** :- It indicates the amount of memory currently marked as referenced or accessed

10. **Anonymous** :- It shows the amount of memory that does not belong to any file

11. **Lazyfree** :- The amount of memory that has been used for lazy allocation of memory
(deferring the allocation until necessary and avoiding allocating physical pages
which will never actually be used). This is the amount of memory which is
marked by madvise(MADV_FREE). The memory isn't freed immediately with
madvise(). It's freed in memory pressure if the memory is clean.

12**. ShmemPmdMapped** :- The amount of shared (shmem/tmpfs) memory backed by huge
pages.

13**. Shared Hugetlb** :- Amount of shared memory backed by hugetlbfs page which is not
counted in "RSS" or "PSS"

14**. Private Hugetlb** :- Amount of memory private to the process backed by hugetlbfs page
which is not counted in "RSS" or "PSS"

15**. Swap** :- Amount of anonymous memory which is swapped out.

16**. Swappss** :- We have to divide Swap value by number of process that are sharing this
Virtual Memory Area.

17**. Locked** :- The number of pages locked within the mapping.

18**. Mapping** :- we see there are executable files and shared libraries that are

mapped privately and that different parts of the same file are mapped
differently (some parts are even mapped more than once). This is because
executable files contain different sections: text, data, rodata, bss . . . each
has a different meaning and is mapped differently.

2. Why are there three entries corresponding to a.out with different permissions?

The a.out format is the original file format for Unix. It consists of three sections: text, data, and bss, which are for program code, initialized data, and uninitialized data, respectively. This format is so simple that it doesn't have any reserved place for debugging information. The only debugging format for a.out is stabs, which is encoded as a set of normal symbols with distinctive attributes.

2.1 .text

This section contains the executable instruction codes and is shared among every process running the same binary. **This section usually has READ and EXECUTE permissions only.** This section is the one most affected by optimization

2.2 .bss

BSS stands for 'Block Started by Symbol'. It holds uninitialized global and static variables. Since the BSS only holds variables that don't have any values yet, it doesn't actually need to store the image of these variables. The size that BSS will require at runtime is recorded in the object file, but the BSS (unlike the data section) doesn't take up any actual space in the object file.**This section usually has READ permission only.**

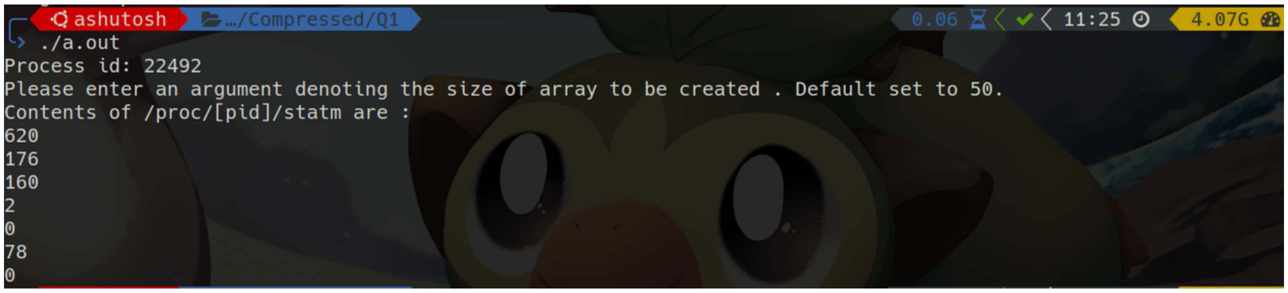
2.3 .data

Contains the initialized global and static variables and their values. It is usually the largest part of the executable. **It usually has READ/WRITE permissions.**

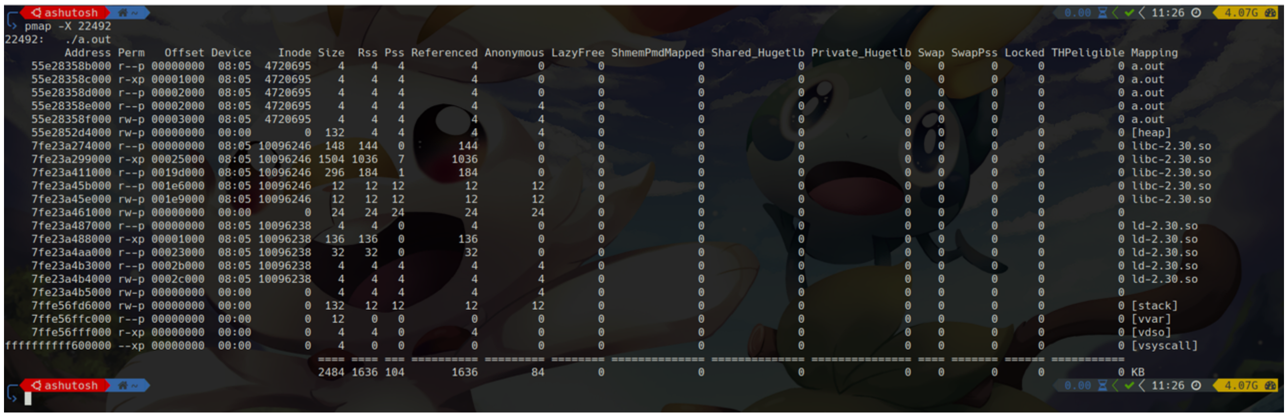
3 Comment on the pmap -X outputs of the C program written for Q1(b).

We can see there are 6 entries corresponding to a.out where they are divided into 2.text, 2.bss, 2.data sections respectively. Which implies we are executing 2 times reading 2 times and writing 2 times.

The following are the outputs for a smaller size array for Q1(b)

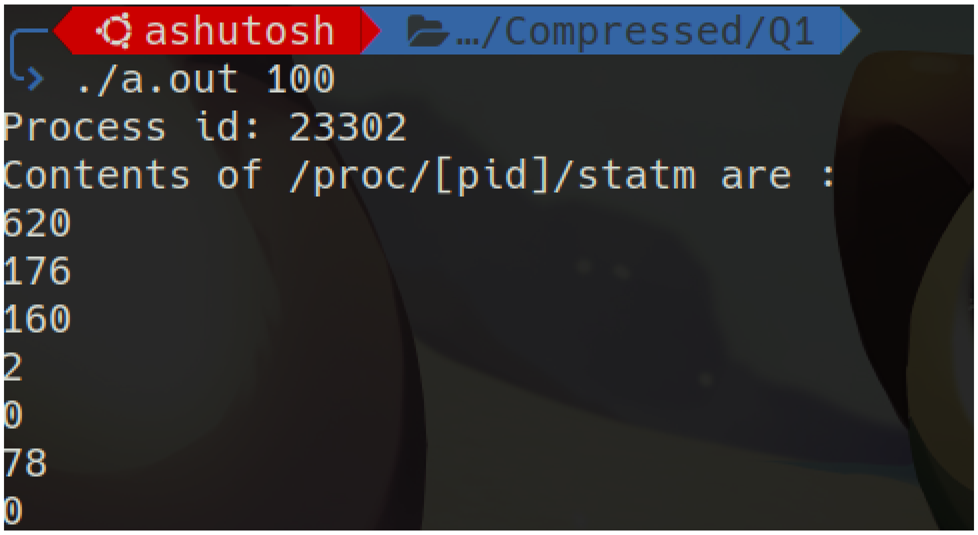


```
 ashutosh  …/Compressed/Q1                                          0.06    11:25    4.07G
> ./a.out
Process id: 22492
Please enter an argument denoting the size of array to be created . Default set to 50.
Contents of /proc/[pid]/statm are :
620
176
160
2
0
78
0
```

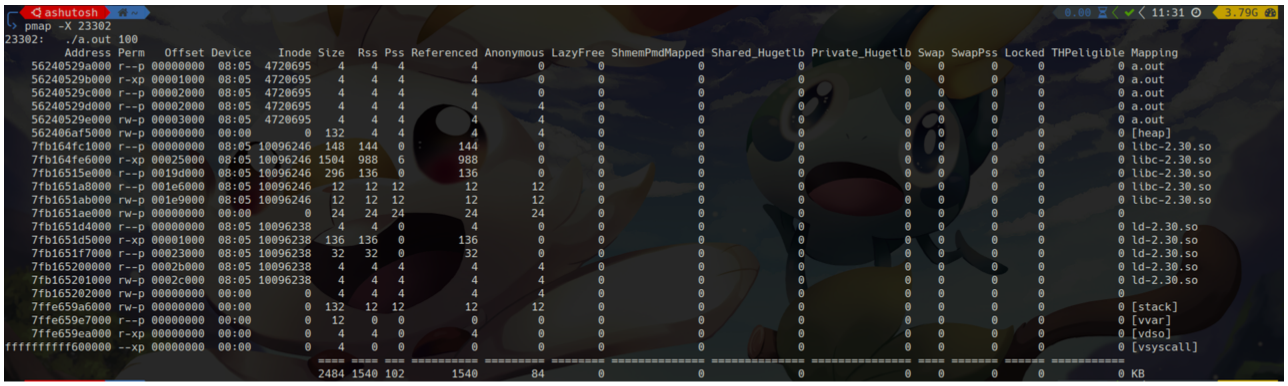


```
 ashutosh  ~                                                        0.00    11:26    4.07G
> pmap -X 22492
22492:   ./a.out
         Address Perm   Offset Device    Inode Size  Rss Pss Referenced Anonymous LazyFree ShmemPmdMapped Shared_Hugetlb Private_Hugetlb Swap SwapPss Locked THPeligible Mapping
    55e28358b000 r--p 00000000  08:05  4720695    4    4   4          4         0        0              0              0               0    0       0      0           0 a.out
    55e28358c000 r-xp 00001000  08:05  4720695    4    4   4          4         0        0              0              0               0    0       0      0           0 a.out
    55e28358d000 r--p 00002000  08:05  4720695    4    4   4          4         0        0              0              0               0    0       0      0           0 a.out
    55e28358e000 r--p 00002000  08:05  4720695    4    4   4          4         4        0              0              0               0    0       0      0           0 a.out
    55e28358f000 rw-p 00003000  08:05  4720695    4    4   4          4         4        0              0              0               0    0       0      0           0 a.out
    55e2852d4000 rw-p 00000000  00:00        0  132    4   4          4         4        0              0              0               0    0       0      0           0 [heap]
    7fe23a274000 r--p 00000000  08:05 10096246  148  144   0        144         0        0              0              0               0    0       0      0           0 libc-2.30.so
    7fe23a299000 r-xp 00025000  08:05 10096246 1504 1036   7       1036         0        0              0              0               0    0       0      0           0 libc-2.30.so
    7fe23a411000 r--p 0019d000  08:05 10096246  296  184   1        184         0        0              0              0               0    0       0      0           0 libc-2.30.so
    7fe23a45b000 r--p 001e6000  08:05 10096246   12   12  12         12        12        0              0              0               0    0       0      0           0 libc-2.30.so
    7fe23a45e000 rw-p 001e9000  08:05 10096246   12   12  12         12        12        0              0              0               0    0       0      0           0 libc-2.30.so
    7fe23a461000 rw-p 00000000  00:00        0   24   24  24         24        24        0              0              0               0    0       0      0           0 
    7fe23a487000 r--p 00000000  08:05 10096238    4    4   0          4         0        0              0              0               0    0       0      0           0 ld-2.30.so
    7fe23a488000 r-xp 00001000  08:05 10096238  136  136   0        136         0        0              0              0               0    0       0      0           0 ld-2.30.so
    7fe23a4aa000 r--p 00023000  08:05 10096238   32   32   0         32         0        0              0              0               0    0       0      0           0 ld-2.30.so
    7fe23a4b3000 r--p 0002b000  08:05 10096238    4    4   4          4         4        0              0              0               0    0       0      0           0 ld-2.30.so
    7fe23a4b4000 rw-p 0002c000  08:05 10096238    4    4   4          4         4        0              0              0               0    0       0      0           0 ld-2.30.so
    7fe23a4b5000 rw-p 00000000  00:00        0    4    4   4          4         4        0              0              0               0    0       0      0           0 
    7ffe56fd6000 rw-p 00000000  00:00        0  132   12  12         12        12        0              0              0               0    0       0      0           0 [stack]
    7ffe56ffc000 r--p 00000000  00:00        0   12    0   0          0         0        0              0              0               0    0       0      0           0 [vvar]
    7ffe56fff000 r-xp 00000000  00:00        0    4    4   0          4         0        0              0              0               0    0       0      0           0 [vdso]
ffffffffff600000 --xp 00000000  00:00        0    4    0   0          0         0        0              0              0               0    0       0      0           0 [vsyscall]
                                             ==== ==== === ========== ========= ======== ============== ============== =============== ==== ======= ====== ===========
                                             2484 1636 104       1636        84        0              0              0               0    0       0      0           0 KB
 ashutosh  ~                                                        0.00    11:26    4.07G
>
```

The following are the outputs for a larger size array for Q1(b)



```
ashutosh  …/Compressed/Q1
 ./a.out 100
Process id: 23302
Contents of /proc/[pid]/statm are :
620
176
160
2
0
78
0
```



```
ashutosh  ~                                                                                                  0.00  11:31  3.79G
 pmap -X 23302
23302:   ./a.out 100
         Address Perm   Offset Device    Inode Size  Rss Pss Referenced Anonymous LazyFree ShmemPmdMapped Shared_Hugetlb Private_Hugetlb Swap SwapPss Locked THPeligible Mapping
    56240529a000 r--p 00000000  08:05  4720695    4    4   4          4         0        0              0              0               0    0       0      0           0 a.out
    56240529b000 r-xp 00001000  08:05  4720695    4    4   4          4         0        0              0              0               0    0       0      0           0 a.out
    56240529c000 r--p 00002000  08:05  4720695    4    4   4          4         0        0              0              0               0    0       0      0           0 a.out
    56240529d000 r--p 00002000  08:05  4720695    4    4   4          4         4        0              0              0               0    0       0      0           0 a.out
    56240529e000 rw-p 00003000  08:05  4720695    4    4   4          4         4        0              0              0               0    0       0      0           0 a.out
    562406af5000 rw-p 00000000  00:00        0  132    4   4          4         4        0              0              0               0    0       0      0           0 [heap]
    7fb164fc1000 r--p 00000000  08:05 10096246  148  144   0        144         0        0              0              0               0    0       0      0           0 libc-2.30.so
    7fb164fe6000 r-xp 00025000  08:05 10096246 1504  988   6        988         0        0              0              0               0    0       0      0           0 libc-2.30.so
    7fb16515e000 r--p 0019d000  08:05 10096246  296  136   0        136         0        0              0              0               0    0       0      0           0 libc-2.30.so
    7fb1651a8000 r--p 001e6000  08:05 10096246   12   12  12         12        12        0              0              0               0    0       0      0           0 libc-2.30.so
    7fb1651ab000 rw-p 001e9000  08:05 10096246   12   12  12         12        12        0              0              0               0    0       0      0           0 libc-2.30.so
    7fb1651ae000 rw-p 00000000  00:00        0   24   24  24         24        24        0              0              0               0    0       0      0           0 
    7fb1651d4000 r--p 00000000  08:05 10096238    4    4   0          4         0        0              0              0               0    0       0      0           0 ld-2.30.so
    7fb1651d5000 r-xp 00001000  08:05 10096238  136  136   0        136         0        0              0              0               0    0       0      0           0 ld-2.30.so
    7fb1651f7000 r--p 00023000  08:05 10096238   32   32   0         32         0        0              0              0               0    0       0      0           0 ld-2.30.so
    7fb165200000 r--p 0002b000  08:05 10096238    4    4   4          4         4        0              0              0               0    0       0      0           0 ld-2.30.so
    7fb165201000 rw-p 0002c000  08:05 10096238    4    4   4          4         4        0              0              0               0    0       0      0           0 ld-2.30.so
    7fb165202000 rw-p 00000000  00:00        0    4    4   4          4         4        0              0              0               0    0       0      0           0 
    7ffe659a6000 rw-p 00000000  00:00        0  132   12  12         12        12        0              0              0               0    0       0      0           0 [stack]
    7ffe659e7000 r--p 00000000  00:00        0   12    0   0          0         0        0              0              0               0    0       0      0           0 [vvar]
    7ffe659ea000 r-xp 00000000  00:00        0    4    4   0          4         0        0              0              0               0    0       0      0           0 [vdso]
ffffffffff600000 --xp 00000000  00:00        0    4    0   0          0         0        0              0              0               0    0       0      0           0 [vsyscall]
                                             ==== ==== === ========== ========= ======== ============== ============== =============== ==== ======= ====== =========== 
                                             2484 1540 102       1540        84        0              0              0               0    0       0      0           0 KB 
```

The size of memory allocated increases because the size of heap+stack will increase as more memory is required for larger array. Also the resident set size may increase or decrease (usually doesn't change much) as this represents the amount of memory present in RAM. The number of shared pages and code size remain unchanged. As we can expect the anonymous column will not change as it shows the memory doesn’t belong to any file.