

First time:

	j	K	l
Child 1	0x7ffeeec45a4c	0x7ffeeec45a48	0x7ffeeec45a44
Child 2	0x7ffeeec45a4c	0x7ffeeec45a48	0x7ffeeec45a44
Child 3	0x7ffeeec45a4c	0x7ffeeec45a48	0x7ffeeec45a44

	G_first	G_second	G_third
Child 1	0x100fbb048	0x100fbb04c	0x100fbb050
Child 2	0x100fbb048	0x100fbb04c	0x100fbb050
Child 3	0x100fbb048	0x100fbb04c	0x100fbb050

	parent
a	0x7ffeeec45b18
b	0x7ffeeec45b14
c	0x7ffeeec45b10
G_first	0x100fbb048
G_second	0x100fbb04c
G_third	0x100fbb050

Second time:

	j	k	l
Child1	0x7ffeef4afa4c	0x7ffeef4afa48	0x7ffeef4afa44
Child2	0x7ffeef4afa4c	0x7ffeef4afa48	0x7ffeef4afa44
Child3	0x7ffeef4afa4c	0x7ffeef4afa48	0x7ffeef4afa44

	G_first	G_second	G_third
Child1	0x100751048	0x10075104c	0x100751050
Child2	0x100751048	0x10075104c	0x100751050
Child3	0x100751048	0x10075104c	0x100751050

	parent
a	0x7ffeef4afb18
b	0x7ffeef4afb14
c	0x7ffeef4afb10
G_first	0x100751048
G_second	0x10075104c
G_third	0x100751050

Observations:

1. All the processes(both child and parent) have same address for the global variables.
2. All the child processes have the same address for local variables j,k,l

Explanation:

When child processes are created an exact copy of the memory map (virtual memory mapping) of the parent is made for each of the child processes. . Each process gets it's own memory space in the physical memory. MMU maps the virtual address to the physical address. The above addresses we got are the virtual addresses of the variables.