

Figure 1 is a line graph showing the size of various data types in bytes as a function of array size. The x-axis is labeled 'Array size' and ranges from 0 to 2,000,000. The y-axis is labeled 'Size [bytes]' and ranges from 0 to 60,000,000. The legend identifies six data types: byte[] (solid line with '+' markers), Byte[] (dashed line with 'x' markers), int[] (dotted line with '*' markers), Integer[] (dash-dot line with square markers), long[] (long-dashed line with solid square markers), and Long[] (short-dashed line with open circle markers). All data types show a linear increase in size as the array size increases. Long[] has the highest size, followed by Integer[], long[], Byte[], int[], and byte[] has the lowest size.

Array size	byte[]	Byte[]	int[]	Integer[]	long[]	Long[]
0	0	0	0	0	0	0
200,000	~1,000,000	~2,000,000	~3,000,000	~4,000,000	~5,000,000	~6,000,000
400,000	~2,000,000	~4,000,000	~6,000,000	~8,000,000	~10,000,000	~12,000,000
600,000	~3,000,000	~6,000,000	~9,000,000	~12,000,000	~15,000,000	~18,000,000
800,000	~4,000,000	~8,000,000	~12,000,000	~16,000,000	~20,000,000	~24,000,000
1,000,000	~5,000,000	~10,000,000	~15,000,000	~20,000,000	~25,000,000	~30,000,000
1,200,000	~6,000,000	~12,000,000	~18,000,000	~24,000,000	~30,000,000	~36,000,000
1,400,000	~7,000,000	~14,000,000	~21,000,000	~28,000,000	~35,000,000	~42,000,000
1,600,000	~8,000,000	~16,000,000	~24,000,000	~32,000,000	~40,000,000	~48,000,000
1,800,000	~9,000,000	~18,000,000	~27,000,000	~36,000,000	~45,000,000	~54,000,000
2,000,000	~10,000,000	~20,000,000	~30,000,000	~40,000,000	~50,000,000	~60,000,000

Array size	byte	Byte	int	Integer	long	Long
0	~1e6	~1e6	~1e6	~1e6	~1e6	~1e6
200,000	~1e6	~1e6	~1e6	~5e6	~2e6	~5e6
400,000	~1e6	~1e6	~1e6	~1e7	~3e6	~1e7
600,000	~1e6	~1e6	~1e6	~1.2e7	~4e6	~1.5e7
800,000	~1e6	~1e6	~1e6	~1.5e7	~5e6	~2e7
1,000,000	~1e6	~1e6	~1e6	~2e7	~7e6	~2.5e7
1,200,000	~1e6	~1e6	~1e6	~2.5e7	~9e6	~3.5e7
1,400,000	~1e6	~1e6	~1e6	~3e7	~11e6	~4.5e7
1,600,000	~1e6	~1e6	~1e6	~3.5e7	~13e6	~5e7
1,800,000	~1e6	~1e6	~1e6	~4e7	~15e6	~5.5e7
2,000,000	~1e6	~1e6	~1e6	~4e7	~18e6	~6e7

Number of Elements	byte[]	Byte[]	int[]	Integer[]	long[]	Long[]
0	0	0	0	0	0	0
2000	~1e+05	~1e+06	~1e+06	~1e+06	~1.5e+06	~1.5e+06
4000	~2e+05	~2e+06	~2e+06	~2e+06	~3e+06	~3e+06
6000	~3e+05	~3e+06	~3e+06	~3e+06	~4.5e+06	~4.5e+06
8000	~4e+05	~4e+06	~4e+06	~4e+06	~6e+06	~6e+06
10000	~5e+05	~5e+06	~5e+06	~5e+06	~8e+06	~8e+06
12000	~6e+05	~6e+06	~6e+06	~6e+06	~9.5e+06	~9.5e+06
14000	~7e+05	~7e+06	~7e+06	~7e+06	~1.1e+07	~1.1e+07
16000	~8e+05	~8e+06	~8e+06	~8e+06	~1.25e+07	~1.25e+07
18000	~9e+05	~9e+06	~9e+06	~9e+06	~1.4e+07	~1.4e+07
20000	~1e+06	~1e+07	~1e+07	~1e+07	~1.6e+07	~1.6e+07

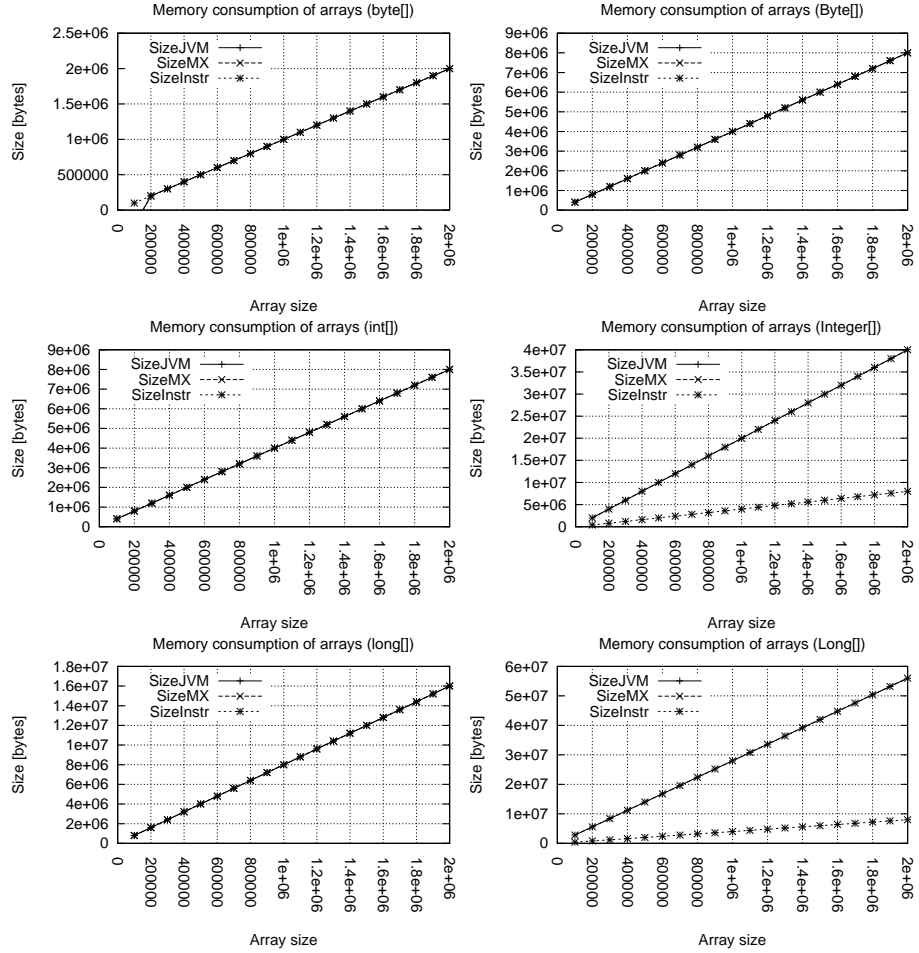


Figure 2: Memory consumption of arrays