5) (5 pts) Write a Fortran variable declaration that declares a two dimensional real array to store the state of 400,000 planets, where each planet can have x, y, and z coordinates and Vx, Vy, Vy velocities all in this array. The dimensions are ordered such that it minimizes computer cache misses. (Fixed size array here, do not use allocatable memory.)

real*8 :: Planet(400 000, 6)

6) (5 pts) How many bytes away from each other are the x-coordinates of two planets that are next to each other in the array? Assume the array is of type **real**.

400,000 bytes

7) (5 pts) In Big $O()$ notation, what are the computational complexities of Bubble Sort and Quick Sort?

Bubble Sort : $O(n^2) \Rightarrow t_2 = t_1 \cdot \left(\frac{n_2}{n_1}\right)^2$

Quick sort : $O(n \log_2(n)) \Rightarrow t_2 = t_1 \cdot \left(\frac{n_2 \log_2(n_2)}{n_1 \log_2(n_1)}\right)$

8) (10 pts) QuickSort takes 2.1 seconds to sort 60,000 numbers. Bubble Sort takes 1.1 minutes. How long would it take to do 12 million numbers with each algorithm? Answer in appropriate units (one that a normal human does not have to stop and think how long this is).

$$\frac{12000,000}{60,000} = \frac{12,00}{6} = 200$$

$$n_2 = 40,000 \cdot n_1$$

$$2,640,000 = 40,000 \cdot 66$$

$$t_2 = 2.1 \cdot \frac{12,000,000}{60,000} \cdot \frac{\log_2(12000000)}{\log_2(60000)}$$

$$622 = 10$$

Quick sort ⇒ 10 minutes 22 seconds

Bubble Sort → 30 days 13 hours