ME5470: Intro parallel scientific computing

HW1

T Naga Vaishnavi

CO22BTECH11014

Q1:

a) When the code was executed with n = 4000, the following file sizes were observed:

ASCII (format 0)	: array_004000_asc.out	327 MB
Binary (format 1)	: array_004000_bin.out	125 MB

b)

The size of the array in memory is calculated as:
Array size = (size of one double) × (total elements in the array)
= 8 bytes × 4000 × 4000
= 122.07 MB

Observations:

1. Memory vs. Disk Usage :
   - The ASCII file is much larger (327 MB) compared to the binary file (125 MB) because numbers stored as text require additional characters for formatting and precision.
   - The size of the binary file (125 MB) closely matches the in-memory size of the array (122.07 MB), as it directly stores the raw binary representation of the data.
2. Efficient Format for Large Data :
   - Binary format is more efficient for storing large datasets because it is compact and faster to read or write compared to ASCII format.

array_004000_asc.out	✓	22-01-2025 21:57	OUT File	3,27,640 KB
array_004000_bin.out	✓	22-01-2025 21:57	OUT File	1,25,078 KB

Q2:

For Modularity wrote the functions :

- dot_prod : For calculating dot product
- mat_dot_prod : For computing matrix multiplied by a vector
- sum_vec : For computing sum of elements in the vector

LOGIC:

If y is a eigenvector of matrix A, then;

$$Ay = vy$$

Where v is the corresponding eigenvalue, then;

Dot product with some vector x on both sides;

$$(Ay).x = v(y.x)$$

If x = [1 1 ·····1], then;

$$\sum(Ay) = v\sum y$$

This implies;

$$\text{Error} = \sum(Ay) - v\sum y$$

If |Error| < some tolerance,

then y is the eigen vector of A

(Eigen value v can be computed by v = (A[i].y)/y[i] for some i where y[i] is non−zero value)

```
PS C:\Users\thiri\OneDrive\Desktop\PSC\hw1-nagavaishnavi260
iles\" ; if ($?) { gcc Q2.c -o Q2 } ; if ($?) { .\Q2 }
vec_000003_000001.in : Yes : -6.000000
vec_000003_000002.in : Yes : -6.000000
vec_000003_000003.in : Yes : -1.000000
vec_000003_000004.in : Not an Eigenvector
PS C:\Users\thiri\OneDrive\Desktop\PSC\hw1-nagavaishnavi260
iles\" ; if ($?) { gcc Q2.c -o Q2 } ; if ($?) { .\Q2 }
vec_000005_000001.in : Yes : 0.268098
vec_000005_000002.in : Not an Eigenvector
vec_000005_000003.in : Yes : 0.986875
vec_000005_000004.in : Yes : 1.399039
PS C:\Users\thiri\OneDrive\Desktop\PSC\hw1-nagavaishnavi260
iles\" ; if ($?) { gcc Q2.c -o Q2 } ; if ($?) { .\Q2 }
vec_000050_000001.in : Not an Eigenvector
vec_000050_000002.in : Yes : 0.479628
vec_000050_000003.in : Yes : 1.337887
vec_000050_000004.in : Not an Eigenvector
PS C:\Users\thiri\OneDrive\Desktop\PSC\hw1-nagavaishnavi260
iles\" ; if ($?) { gcc Q2.c -o Q2 } ; if ($?) { .\Q2 }
vec_000080_000001.in : Yes : 0.333018
vec_000080_000002.in : Yes : 0.493142
vec_000080_000003.in : Yes : 0.939275
vec_000080_000004.in : Not an Eigenvector
PS C:\Users\thiri\OneDrive\Desktop\PSC\hw1-nagavaishnavi260
```