GEBZE TECHNICAL UNIVERSITY

CSE344
SYSTEMS PROGRAMMING COURSE

# HOMEWORK2 REPORT

Baran Solmaz
1801042601

## Solution Method:

Child process receives 30 characters and output file name from parent process to calculate covariance matrix. When child calculates, writes to output file.

Parent process creates a child processes when it read 30 characters from input file and sends to the child process to calculate covariance matrix then waits for the childs to finish. When all childs finish,parent process reads output file to calculate frobenius norms of childs' covariance matricies. After calculation of frobenius norms,finds the closest pair.

When parent process receives SIGINT,sends to all childs to stop the process and waits for childs to stop.When all childs stop,parent process deletes output file and frees all allocated memory then exits.

When child process receives SIGINT,frees all allocated memory and exits.

## Design Decisions/Algorithm:

Read 10 x 3-character,
Create child process and send characters,
Wait for child to finish,
Read file and calculate norm,
Find closest pair.

## Function Explanation:

Child Process:

```
void display(float arr[][COL], int r, int c);                    Prints Matrix
void matrix_initialize(char **mat, float arr[ROW][COL]); Converts char matrix to float matrix
void fileOP(float result[COL][COL],char *fileName,char *number); Prints File
void handler(int sig_number);Signal handler
void sig_check(); Checks signal
int lockFile(char *filepath);  Locks file
void unlockFile(int fd);Unlocks file
```

```
void matrix_product(float first[][ROW], float second[][COL], float result[][COL], int r1, int c1, int r2, int c2);
void matrix_transpose(float first[ROW][COL], float result[COL][ROW], int r1, int c1);
void matrix_extraction(float first[ROW][COL], float second[ROW][COL], float result[ROW][COL], int r1, int c1);
void matrix_divide_n(float first[][COL], float result[][COL], int r1, int c1, int n);
void matrix_covariance(float first[ROW][COL], float result[COL][COL], int r1, int c1);
```

Parent Process: