Project: - Defect Programmer Assignments [Sprint-1]

Story: - The story is about the Clients of software development company report defects in software they purchased. These defects are assigned to programmers depending on functional area they are handling.

getDefect.c

main.c

Employee Details File

EX: EmployeeDetails.txt

Multiple files of defects

EX: Defect1.txt, Defect2.txt

Input Files

**Story Starts**

Input Files

Process Flow

assign.c

Output Files

Inside main.c file

Inside main function

-- Start –

Take location of input files as Command Line arguments.

Validate Command Line Arguments

If insufficient arguments are passed display proper usage

Create separate threads for each input file passed

Declare an array to strore all thread IDs

Display proper error in case thread is not created

Pass file location with the thread to getdefect function

Wait for all child threads to join

-- End --

Inside getdefect.c file

Inside getdefect function

-- Start –

Takes the location of file as input

Open the file with a file pointer

if the pointer is NULL

print "cannot open file" and exit

read each line of the file

if the line returns NULL

return as we have reached the end of file

else

pass the content of the line to checkValidity function

-- End --

Inside checkValidity function

-- Start –

Takes defect line as argument

Tokenises the line in separate parts.

If all required information is given then consider it as a valid entry.

Else Consider it as invalid entry and pass it to InvalidDefect Function

For valid entries, create store values in proper data structure.

Add all defects inside an array of used data structure.

Pass this array with valid entries to assignProgrammer Function.

-- End --

Inside InvalidDefect function

-- Start –

Takes Invalid Defect ID and Defect String/Line as Input

// Display Defect ID and Unvalible Informations

PRINT "Unvalible Info of <Defect ID>"

// Append Invalid Entry to File "invaliddefect.txt"

IF file is not created

CREATE "invaliddefect.txt"

APPEND entire invalid defect line

ELSE

APPEND entire invalid defect line

-- End --

Inside assign.c file

Inside assignProgrammer function

-- Start –

Takes array of valid entries as an argument

Declare an data structure to store employee data

Create an array to store employee data.

Call function getEmployee and pass this array.

Checks for status of all elements inside array.

IF status is not open ignore it

IF status is open then call seachProgrammer Function,

pass array of employee structure and array of valid defect structure to it.

-- End --

Inside getEmployee function

-- Start –

Takes array of employee structure as an argument

Open input file with employee data using File pointer

Display error if file can't be opened for any reason

Read all lines and tokenise it into employee information

Store these information in employee structure

Store all employee in the array

-- End --

Inside displayAssignemt function

-- Start –

Takes array of valid defects entry as argument

Checks for status = assigned.

Defects assigned to programmers to be displayed along with defect description, module name, functional area, filed-on date type, Emp ID and EMP Name.

-- End --

Inside searchProgrammer function

-- Start –

Takes array of employee structure and array of valid defect entry structure as argument

maps programmer to the defect if there functional requrirement is matched

If programmer is assigned, call changeStatus Function and displayAssignemt function

IF multiple programmer is found assign to first one.

After assigning change defect status to assigned.

Call createEmployeeFIle Function, pass Employee and defect sturcuture.

IF programmer is not found, call unsignedDefect Function, pass defect structure to it.

-- End --

**Finished**