

ENGR 0012 – Engineering Problem Solving

Goals for this week:

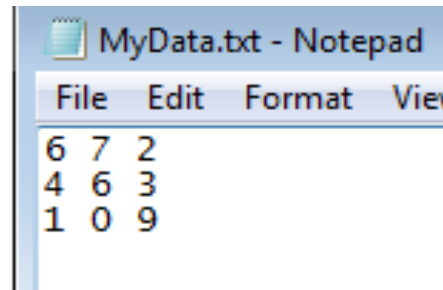
- Review

Please submit your HW!
(Old and new submission systems)

Review:

- Functions
- Reading from file / EOF
- Arrays

Review:



- Ask user for file name and check if it exists
- Read the data into arrays x, y, z
- Use function to give user the choice to add or multiply
- Use function to add or multiply the arrays
- From the main, print the first and last elements of the resulting array

```
File name:
MyData.txt
x      y      z
6      7      2
4      6      3
1      0      9
Pick 1 to add arrays.  Pick 2 to multiply arrays:
1
First and last elements of ResultArray are 15 and 10
Press any key to continue . . . _
```

```

#include <stdio.h>
#include <stdlib.h>

int Pick(void);
void Operation(int, int, int[], int[], int[], int[]);

int main()
{
    char filename[50]; //Declare file name as a string

    FILE *infile; //Create pointer

    do
    {
        printf("File name: \n"); //Ask for file name
        scanf("%s", filename);

        infile = fopen(filename, "r"); //Open file
    } while (infile == NULL); //Check if file exists

    //Read in data
    int x[10], y[10], z[10]; //Declare arrays
    int status = 3, i = 0;
    printf("x \t y \t z \n");
    while (status != EOF)
    {
        status = fscanf(infile, "%d %d %d", &x[i], &y[i], &z[i]);
        if (status==EOF)
        {
            break;
        }
        printf("%d \t %d \t %d \n", x[i], y[i], z[i]);
        i++;
    }

    //Use funtion to get user choice
    int choice;
    choice = Pick();

    //Call function to perform operation
    int ResultArray[10];
    Operation(choice, i, x, y, z, ResultArray);

    //Print results
    printf("First and last elements of ResultArray are %d and %d \n", ResultArray[0], ResultArray[i-1]);

}

```

```

//Function 1: Pick
int Pick(void)
{
    int mychoice;
    //Ask user for choice
    printf("Pick 1 to add arrays. Pick 2 to multiply arrays: \n");
    scanf("%d", &mychoice);

    return(mychoice); //Need to return variable
}

//Function 2: Operation
void Operation(int choice, int i, int x[], int y[], int z[], int ResultArray[])
{
    //Use if statement to perform correct operation
    if (choice == 1)
    {
        for (int j = 0; j <= i - 1; j++)
        {
            ResultArray[j] = x[j] + y[j] + z[j];
        }
    }
    else if (choice == 2)
    {
        for (int j = 0; j <= i - 1; j++)
        {
            ResultArray[j] = x[j] * y[j] * z[j];
        }
    }
}

```

```
#include <string.h>
#include <stdio.h>
#include <stdlib.h>
```

```
double func1(int, double, int[]);
```

```
void main(void)
```

```
{
```

```
double aa = 15, cc = 8, result;
int xx = 3, yy = 1, i, num, Div;
int Array[4] = { 2, 1, 4, 3 };
```

```
for (i = 4; i > 1; i--)
{
```

```
    num = 15 % i;
```

```
    switch (num)
    {
```

```
    case 1:
```

```
        result = func1(xx, aa, Array);
        printf("result=%7.2lf\n", result);
```

```
    case 2:
```

```
        Div = xx / cc;
        printf("Div=%5d\n", Div);
        result = func1(yy, cc, Array);
        printf("result = %.3lf\n", result);
        break;
```

```
    case 3:
```

```
        result = yy / xx;
        printf("result=%lf\n", result);
        break;
```

```
    default:
```

```
        printf("Math practice\n");
```

```
    }
```

```
}
```

```
printf("The array is Array = [%d %d %d %d]\n", Array[0], Array[1], Array[2], Array[3]);
```

```
printf("Done!\n");
```

```
}
```

Review. What is the output?

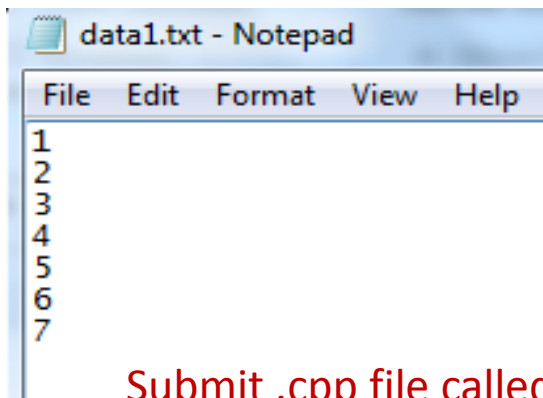
Submit sheet of paper
with your results!

```
double func1(int aa, double xx, int Var[])
{
    double num;
    int i;
    num = xx / aa;

    for (i = 0; i <= 3; i++)
    {
        Var[i] = 2 * Var[i];
    }
    return num;
}
```

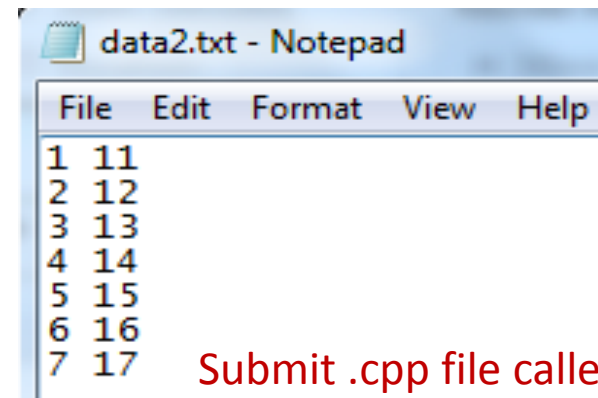
Practice Problems (EOF)

- Part 1:
- Create data1.txt
- Ask user for file name and check if it exists
- Read the data points as an array from the file and print to the screen and to a file



Submit .cpp file called
"Mena_Time_Part1Team#"
("Mena_10am_Part1L01")

- Part 2:
- Create data2.txt
- Ask user for file name and check if it exists
- Read the data points as an array from the file and print to the screen and to a file



Submit .cpp file called
"Mena_Time_Part2Team#"
("Mena_10am_Part2L01")

Review: What is the output?

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    //Declare variables
    int a = 3, b = 6, c = 9, d = 12, e, f;
    double alfa = 2, beta = 3, gamma = 3.3, delta, epsilon;

    //Math operations
    delta = alfa*(a*b / c);
    e = (d%c)*b;

    epsilon = (alfa*alfa / beta)*(c / b);
    f = alfa*gamma;

    //Print results
    printf("Hello!");
    printf("\ndelta=%5.1lf \ne=%7d", delta, e);
    printf("\nepsilon=%1f \nf=%d", epsilon, f);
    printf("\nalfa=%4.2lf", alfa);

    printf("\n");
}
```

Review: What is the output?

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    //Declare variables
    int a = 3, b = 6, c = 9, d = 12, e, f;
    double alfa = 2, beta = 3, gamma = 3.3, delta, epsilon;

    //Math operations
    delta = alfa*(a*b / c);
    e = (d%c)*b;

    epsilon = (alfa*alfa / beta)*(c / b);
    f = alfa*gamma;

    //Print results
    printf("Hello!");
    printf("\ndelta=%5.1lf \ne=%7d", delta, e);
    printf("\nepsilon=%1f \nf=%d", epsilon, f);
    printf("\nalfa=%4.2lf", alfa);

    printf("\n");
}
```

```
Hello!
delta=  4.0
e=      18
epsilon=1.333333
f=6
alfa=2.00
Press any key to continue . . .
```


Review: What is the output?

```
#include <string.h>
#include <stdio.h>
#include <stdlib.h>

void main(void)
{
    int i = 4, j = 2;
    char A1[40] = "Today is a good day";
    char A2[40] = "I love Engineering";
    char A3[40] = "Hi";
    char NEW[30] = { A1[i], A2[i + 1], A1[i + 3] };

    printf("%s\n", A3);
    printf("%s\n", NEW);
    for (i = 6; i < 15; i++)
    {
        printf("%c\n", A2[i]);
    }
}
```

Review: What is the output?

```
#include <string.h>
#include <stdio.h>
#include <stdlib.h>

void main(void)
{
    int i = 4, j = 2;
    char A1[40] = "Today is a good day";
    char A2[40] = "I love Engineering";
    char A3[40] = "Hi";
    char NEW[30] = { A1[i], A2[i + 1], A1[i + 3] };

    printf("%s\n", A3);
    printf("%s\n", NEW);
    for (i = 6; i < 15; i++)
    {
        printf("%c\n", A2[i]);
    }
}
```

```
Hi
yes
E
n
g
i
n
e
e
r
Press any key to continue . . .
```

Submit .cpp file called
"Mena_Time_Review1Team#"
("Mena_10am_ReviewL01")

Review. Write a program that will:

- Call Function1: Display a header



```
Name, date, etc
```

Submit .cpp file called
"Mena_Time_Review1Team#"
("Mena_10am_ReviewL01")

Review. Write a program that will:

- Call Function2: Ask for data file name, check that file exists, load data into 2-dimensional array (look at data file) by using EOF, print data to file "ReviewPrint.txt"

```
Name, date, etc
Enter file name: tr
Enter file name: tr
Enter file name: tr
Enter file name: ReviewData.txt
```

ReviewPrint.txt - Notepad

File	Edit	Format	View	Help
11.000000	15.000000	48.000000		
47.000000	38.000000	48.000000		
26.000000	1.000000	46.000000		
42.000000	48.000000	5.000000		
1.000000	11.000000	49.000000		
7.000000	14.000000	42.000000		
36.000000	8.000000	17.000000		
15.000000	31.000000	40.000000		
34.000000	41.000000	29.000000		
34.000000	31.000000	30.000000		
44.000000	17.000000	16.000000		
46.000000	23.000000	18.000000		
18.000000	40.000000	37.000000		
49.000000	28.000000	4.000000		
5.000000	3.000000	48.000000		
32.000000	45.000000	35.000000		
8.000000	14.000000	19.000000		
10.000000	37.000000	25.000000		
41.000000	22.000000	22.000000		
24.000000	9.000000	15.000000		

Submit .cpp file called
"Mena_Time_Review1Team#"
("Mena_10am_ReviewL01")

Review. Write a program that will:

- In Main: Ask user to provide an integer between 0 and 50, and check for user error

```
Name, date, etc
Enter file name: tr
Enter file name: tr
Enter file name: tr
Enter file name: ReviewData.txt
Please provide a number between 0 and 50
-2
Please provide a number between 0 and 50
98
Please provide a number between 0 and 50
9
```

Submit .cpp file called
"Mena_Time_Review1Team#"
("Mena_10am_ReviewL01")

Review. Write a program that will:

- Call Function3: Perform 60%Num and return the number to the main
- In Main: Print Mod to screen and file

```
Name, date, etc
Enter file name: tr
Enter file name: tr
Enter file name: tr
Enter file name: ReviewData.txt
Please provide a number between 0 and 50
-2
Please provide a number between 0 and 50
98
Please provide a number between 0 and 50
9
Mod = 6
```

ReviewPrint.txt - Notepad

File	Edit	Format	View	Help
11.000000	15.000000	48.000000		
47.000000	38.000000	48.000000		
26.000000	1.000000	46.000000		
42.000000	48.000000	5.000000		
1.000000	11.000000	49.000000		
7.000000	14.000000	42.000000		
36.000000	8.000000	17.000000		
15.000000	31.000000	40.000000		
34.000000	41.000000	29.000000		
34.000000	31.000000	30.000000		
44.000000	17.000000	16.000000		
46.000000	23.000000	18.000000		
18.000000	40.000000	37.000000		
49.000000	28.000000	4.000000		
5.000000	3.000000	48.000000		
32.000000	45.000000	35.000000		
8.000000	14.000000	19.000000		
10.000000	37.000000	25.000000		
41.000000	22.000000	22.000000		
24.000000	9.000000	15.000000		
Mod = 6				

Submit .cpp file called
"Mena_Time_Review1Team#"
("Mena_10am_ReviewL01")

Review. Write a program that will:

- Call Function4: Find the min and sum of the first five rows of the data matrix
- In Main: Print min and sum to screen and file

```
Name, date, etc
Enter file name: tr
Enter file name: tr
Enter file name: tr
Enter file name: ReviewData.txt
Please provide a number between 0 and 50
-2
Please provide a number between 0 and 50
98
Please provide a number between 0 and 50
9
Mod = 6
The sum is = 436.000000
The min is = 1.000000
Press any key to continue . . . _
```

ReviewPrint.txt - Notepad

File	Edit	Format	View	Help
11.000000		15.000000		48.000000
47.000000		38.000000		48.000000
26.000000		1.000000		46.000000
42.000000		48.000000		5.000000
1.000000		11.000000		49.000000
7.000000		14.000000		42.000000
36.000000		8.000000		17.000000
15.000000		31.000000		40.000000
34.000000		41.000000		29.000000
34.000000		31.000000		30.000000
44.000000		17.000000		16.000000
46.000000		23.000000		18.000000
18.000000		40.000000		37.000000
49.000000		28.000000		4.000000
5.000000		3.000000		48.000000
32.000000		45.000000		35.000000
8.000000		14.000000		19.000000
10.000000		37.000000		25.000000
41.000000		22.000000		22.000000
24.000000		9.000000		15.000000
Mod = 6				
The sum is = 436.000000				
The min is = 1.000000				

A pointer is a type of variable that, rather than storing data, stores the memory address for another variable



(a value)



(so it points to that variable)

- To declare a variable as a pointer use * before the variable name
- Pointer variables should be of the same type as the variable to which they are pointing
- You can use them to return more than one variable from a function

Two important symbols: * (indicates that it will be a pointer variable) and & (means “the address of”¹)

Use * to declare a variable as a pointer and use & to assign it a non-pointer variable's address

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    double a = 5.0, *pointer_a = &a;

    printf("\n a=%lf \n", a);

    printf("\n pointer for a=%p \n\n", pointer_a);
}
```

Use %p to print the content of a pointer variable

So:

- Regular variable a has been assigned a value of 5.0
- Pointer variable pointer_a has been assigned the address of variable a

For example:

For example:

```
What is in c = 5, address of c = 0035F8E4
```

```
What is in ptr_c = 0035F8E4, address of ptr_c = 0035F8D8
```

```
What is stored at address stored in ptr_c (indirection-get data at address *ptr_c) = 5
```

```
Press any key to continue . . . _
```

Regular variable,
assigned a value

Pointer variable,
assigned a location

```
#include <stdio.h>
#include <stdlib.h>

int main()
{
    //Declare variables
    int c = 5, *ptr_c = &c;

    //Print what is stored in c and address of c
    printf("What is in c = %d, address of c = %p\n", c, &c);

    //Print what is stored in ptr_c and address of ptr_c
    printf("\nWhat is in ptr_c = %p, address of ptr_c = %p\n", ptr_c, &ptr_c);

    //Print data stored at address stored in ptr_c (indirection)
    printf("\nWhat is stored at address stored in ptr_c (indirection-get data at address *ptr_c) = %d\n", *ptr_c);
}
```

Indirection operation: get
data stored at that address