Array Intro

itemsAr

0	1	2
~~5~~	~~10~~	~~15~~

index	sum
✓0	✓5
✓1	✓15
✓2	✓30

```
int main()
{
    Const int MAX_AR=3;

    int itemsAr[MAX_AR]={5,10,15};
    int Sum;
    int index;

    Sum=0;

    for (index =0; index < MAX_AR; index++)
     {
        Cout<<"Enter an integer: ";
        Cin  >> itemsAR[index];

         Sum= Sum + itemsAr[index];
     }
                                                    30
    Cout << "The sum of the numbers = " << sum << endl;
    Cout << "The numbers in reverse are: ";

    for(index=MAX_AR-1 ; index >-1; index --)
     {
        Cout<< itemsAr[index] <<", ";
     }

     return 0;
                              15, 10, 5
}
```

E27 part 2 1/2

```
int itemsAr[AR_SIZE] = {0};
//INPUT - read input from a file into the array
index = 0;

while(inFile&& index< AR_SIZE)
{
        inFile >> itemsAr[index];

        index++;
}
// SEARCH - for searchItem in the array
searchItem = 10;
index      = 0;
found      = false;
             ~~0~~,1,2   6,6,6    F,F,T
while(index<AR_SIZE&& !found)
{
                        0,1,2         10
        if(itemsAr[index]== searchItem)
        {
            found = true;
        }

        else
        {
            index++;
        }
}
```

INPUT FILE

~~3~~ ~~7~~ 10 2 11

AR_SIZE	index	found	searchItem
6	0	false	10
6	1	false	10
6	2	true	10

itemsAr

0	1	2	3	4	5
3	7	10			

E27 part 2 [2/2]

```
Const int AR_SIZE = 6;

      int itemsAr[AR_SIZE] = {3, 7, 10, 2, 10, 12};
      int index;
      int searchItem;
      int instanceCount;

     searchItem     = 10;
     instanceCount = 0;

     for(index = 0; index < AR_SIZE; index ++)
      {
            if (itemsAr[index] == searchItem)
            {
                ++ instanceCount;
            }
      }
```

itemsAr

		1		2	
~~0~~	~~1~~	2	~~3~~	4	~~5~~
3	7	10	2	10	12

AR_SIZE	index	instanceCount	searchItem
6	0	0	10
6	1	0	10
6	2	1	10
6	3	1	10
6	4	2	10
6	5	2	10

Looping in Arrays

How do we know when to use a For loop vs a While loop?

If we need to check every element of our array, we use a for loop.

If we need to check our array for 1 Seecific value, we use a while loop to search for it until it is found once.

What is the one check we have in every loop when using arrays?

	[0]	[1]	[2]	[3]	[4]	[5]	[6]
ageTrain	85	79	69	63	36	41	28

So if I am calculating the average for the array do I have to inspect every element? Yes, because you need to add every index value into a total in order to divide by the count (using a for loop)

```
/*******************************************************************************
 * AUTHOR     : Blake Allard
 * STUDENT ID : 358888
 * EXR #27    : Arrays 4/4
 * CLASS      : CS1A
 * SECTION    : MW 8am
 * DUE DATE   : 11/18/24
 ******************************************************************************/

#ifndef HEADER_H_
#define HEADER_H_

#include <iostream>
#include <iomanip>
#include <string>
#include <fstream>
#include <sstream>
using namespace std;



void ReadArrayFromFile (int numberArray[], const int AR_SIZE);

string ArrayToStringOss(int numberArray[], const int AR_SIZE);

void OutputArrayToFile (int numberArray[], const int AR_SIZE);

double AverageOfArray  (int numberArray[], const int AR_SIZE);



#endif /* HEADER_H_ */

```

```
1 18 25 32 18 16 54 25 37 18 64
```

```
1 18 25 32 18 16 54 25 37 18 64
```

```
Enter an integer: 5
Enter an integer: 10
Enter an integer: 15
The sum of the numbers is = 30
The numbers in reverse are: 15, 10, 5,
```

```cpp
/**************************************************************************
 * AUTHOR     : Blake Allard
 * STUDENT ID : 358888
 * EXR #27    : Arrays 4/4
 * CLASS      : CS1A
 * SECTION    : MW 8am
 * DUE DATE   : 11/18/24
 *************************************************************************/

#include "header.h"


int main()
{
    //CONSTANT
    const int AR_SIZE = 10;

    //VARIABLES
    int numArray[AR_SIZE] = {0};
    int index;
    string arrayStringIn;
    string arrayStringOut;
    double arrayAvg;


    //PROCESSING - calling functions
    ReadArrayFromFile(numArray, AR_SIZE);

    arrayStringIn = ArrayToStringOss(numArray, AR_SIZE);

    arrayAvg = AverageOfArray(numArray, AR_SIZE);


    //OUTPUT - output converted string to console & file
    cout << arrayStringIn;

    OutputArrayToFile(numArray, AR_SIZE);

    index = FindFirstInstance(numArray, AR_SIZE, 10);

    if (index != -1)
    {
        cout << "\nValue 10 found at index: " << index << endl;
    }
    else
    {
        cout << "\nValue 10 not found! \n";
    }



    cout << setprecision(3);
    cout << "\n\nThe average of the provided array is: " << arrayAvg << endl;
    cout << setprecision(6);




    return 0;
}

```

```
/*******************************************************************************
 * AUTHOR     : Blake Allard
 * STUDENT ID : 358888
 * EXR #27    : Arrays 4/4
 * CLASS      : CS1A
 * SECTION    : MW 8am
 * DUE DATE   : 11/18/24
 ******************************************************************************/

#include "header.h"


void ReadArrayFromFile(int numArray[], const int AR_SIZE)
{
    ifstream inFile;
    int index;

    inFile.open("E27_Input.txt");

    //load the array from a file
    index = 0;


    while (inFile && index < AR_SIZE)
    {
        inFile >> numArray[index];

        index++;
    }

    inFile.close();

}

```

```cpp
/**************************************************************************
 * AUTHOR     : Blake Allard
 * STUDENT ID : 358888
 * EXR #27    : Arrays 4/4
 * CLASS      : CS1A
 * SECTION    : MW 8am
 * DUE DATE   : 11/18/24
 *************************************************************************/

#include "header.h"

string ArrayToStringOss(int numArray[], const int AR_SIZE)
{
    ostringstream outOss;
    int           index;

    index = 0;

    while (index < AR_SIZE)
    {
        outOss << numArray[index] << " ";

        index++;
    }

    return outOss.str();

}

```

```
/********************************************************************************
 * AUTHOR      : Blake Allard
 * STUDENT ID : 358888
 * EXR #27     : Arrays 4/4
 * CLASS       : CS1A
 * SECTION     : MW 8am
 * DUE DATE    : 11/18/24
 *******************************************************************************/

#include "header.h"


void OutputArrayToFile(int numArray[], const int AR_SIZE)
{
    ofstream outFile;
    int      index;

    outFile.open("E27_Output.txt");

    index = 0;

    while (outFile && index < AR_SIZE)
    {
        outFile << numArray[index] << " ";

        index++;
    }

    outFile.close();
}
```

```cpp
/*******************************************************************************
 * AUTHOR     : Blake Allard
 * STUDENT ID : 358888
 * EXR #27    : Arrays 4/4
 * CLASS      : CS1A
 * SECTION    : MW 8am
 * DUE DATE   : 11/18/24
 ******************************************************************************/

#include "header.h"


double AverageOfArray(int numArray[], const int AR_SIZE)
{
    int    index;
    double arraySum;

    arraySum = 0.0;
    index    = 0;

    while (index < AR_SIZE)
    {
        arraySum = arraySum + numArray[index];

        index++;
    }

    return arraySum / AR_SIZE;
}
```

```cpp
/*******************************************************************************
 * AUTHOR     : Blake Allard
 * STUDENT ID : 358888
 * EXR #27    : Arrays 4/4
 * CLASS      : CS1A
 * SECTION    : MW 8am
 * DUE DATE   : 11/18/24
 ******************************************************************************/

#ifndef HEADER_H_
#define HEADER_H_

#include <iostream>
#include <iomanip>
#include <string>
#include <fstream>
#include <sstream>
using namespace std;


void ReadArrayFromFile (int numberArray[], const int AR_SIZE);

string ArrayToStringOss(int numberArray[], const int AR_SIZE);

void OutputArrayToFile (int numberArray[], const int AR_SIZE);

double AverageOfArray  (int numberArray[], const int AR_SIZE);

int FindFirstInstance  (int numberArray[], const int AR_SIZE,
                        int &elementValue);

int FindAllInstances   (int numberArray[], const int AR_SIZE,
                        int &elementValue);

#endif /* HEADER_H_ */

```

```
1 18 25 32 18 16 54 25 37 18 64
```

```
1 18 25 32 18 16 54 25 37 18 64
```

```
18 25 32 18 16 54 25 37 18 64

The average of the provided array is: 30.7

Value 32 located at index: 2

18 25 32 18 16 54 25 37 18 64

The average of the provided array is: 30.7

Value 64 located at index: 9

18 25 32 18 16 54 25 37 18 64

The average of the provided array is: 30.7

Value 16 located at index: 4

18 25 32 18 16 54 25 37 18 64

The average of the provided array is: 30.7

Value 2000 not found!


```

```
18 25 32 18 16 54 25 37 18 64

The average of the provided array is: 30.7

Value 18 located at index: 0

Value found at index/indices: 0 3 8

18 25 32 18 16 54 25 37 18 64

The average of the provided array is: 30.7

Value 64 located at index: 9

Value found at index/indices: 9

18 25 32 18 16 54 25 37 18 64

The average of the provided array is: 30.7

Value 25 located at index: 1

Value found at index/indices: 1 6

18 25 32 18 16 54 25 37 18 64

The average of the provided array is: 30.7

Value 25 located at index: 1

25 was found at index/indices: 1 6

18 25 32 18 16 54 25 37 18 64

The average of the provided array is: 30.7

Value 8000000 not found!
```

```cpp
/*******************************************************************************
 * AUTHOR      : Blake Allard
 * STUDENT ID  : 358888
 * EXR #27     : Arrays 4/4
 * CLASS       : CS1A
 * SECTION     : MW 8am
 * DUE DATE    : 11/18/24
 ******************************************************************************/

#include "header.h"


int main()
{
    //CONSTANT
    const int AR_SIZE = 10;

    //VARIABLES
    int numArray[AR_SIZE] = {0};
    int index;
    string arrayStringIn;
    string arrayStringOut;
    double arrayAvg;
    int elementValue;
    int allInstances;

    //INITIALIZE
    elementValue = 8000000;


    //PROCESSING - calling functions
    ReadArrayFromFile(numArray, AR_SIZE);

    arrayStringIn = ArrayToStringOss(numArray, AR_SIZE);

    arrayAvg = AverageOfArray(numArray, AR_SIZE);

    index = FindFirstInstance(numArray, AR_SIZE, elementValue);


    //OUTPUT - output converted string to console & file
    cout << arrayStringIn;

    OutputArrayToFile(numArray, AR_SIZE);

    cout << setprecision(3);

    cout << "\n\nThe average of the provided array is: " << arrayAvg << "\n\n";

    cout << setprecision(6);


    if (index != -1)
    {
        cout << "Value " << elementValue << " located at index: " << index;
        cout << "\n\n";
    }
    else
    {
        cout << "Value " << elementValue << " not found!\n";
        cout << "\n\n";
    }

    allInstances = FindAllInstances(numArray, AR_SIZE, elementValue);

    cout << "\n\n";


    return 0;
}

```

```
/*******************************************************************************
 * AUTHOR     : Blake Allard
 * STUDENT ID : 358888
 * EXR #27    : Arrays 4/4
 * CLASS      : CS1A
 * SECTION    : MW 8am
 * DUE DATE   : 11/18/24
 ******************************************************************************/

#include "header.h"


void ReadArrayFromFile(int numArray[], const int AR_SIZE)
{
    ifstream inFile;
    int index;

    inFile.open("E27_Input.txt");

    //load the array from a file
    index = 0;


    while (inFile && index < AR_SIZE)
    {
        inFile >> numArray[index];

        index++;
    }

    inFile.close();

}

```

```cpp
/*******************************************************************************
 * AUTHOR     : Blake Allard
 * STUDENT ID : 358888
 * EXR #27    : Arrays 4/4
 * CLASS      : CS1A
 * SECTION    : MW 8am
 * DUE DATE   : 11/18/24
 ******************************************************************************/

#include "header.h"

string ArrayToStringOss(int numArray[], const int AR_SIZE)
{
    ostringstream outOss;
    int           index;

    index = 0;

    while (index < AR_SIZE)
    {
        outOss << numArray[index] << " ";

        index++;
    }

    return outOss.str();

}

```

```cpp
/*******************************************************************************
 * AUTHOR     : Blake Allard
 * STUDENT ID : 358888
 * EXR #27    : Arrays 4/4
 * CLASS      : CS1A
 * SECTION    : MW 8am
 * DUE DATE   : 11/18/24
 ******************************************************************************/

#include "header.h"


void OutputArrayToFile(int numArray[], const int AR_SIZE)
{
    ofstream outFile;
    int      index;

    outFile.open("E27_Output.txt");

    index = 0;

    while (outFile && index < AR_SIZE)
    {
        outFile << numArray[index] << " ";

        index++;
    }

    outFile.close();
}





```

```cpp
/***************************************************************************
 * AUTHOR     : Blake Allard
 * STUDENT ID : 358888
 * EXR #27    : Arrays 4/4
 * CLASS      : CS1A
 * SECTION    : MW 8am
 * DUE DATE   : 11/18/24
 **************************************************************************/

#include "header.h"


double AverageOfArray(int numArray[], const int AR_SIZE)
{
    int    index;
    double arraySum;

    arraySum = 0.0;
    index    = 0;

    while (index < AR_SIZE)
    {
        arraySum = arraySum + numArray[index];

        index++;
    }

    return arraySum / AR_SIZE;
}
```

```cpp
/*******************************************************************************
 * AUTHOR     : Blake Allard
 * STUDENT ID : 358888
 * EXR #27    : Arrays 4/4
 * CLASS      : CS1A
 * SECTION    : MW 8am
 * DUE DATE   : 11/18/24
 ******************************************************************************/

#include "header.h"

int FindFirstInstance(int numArray[], const int AR_SIZE, int &elementValue)
{
    int index;
    int result;

    result = -1;


    for (index = 0; index < AR_SIZE; index++)
    {
        if (numArray[index] == elementValue && result == -1)
        {
            result = index;
        }
    }


    return result;
}
```

```
/******************************************************************************
 * AUTHOR     : Blake Allard
 * STUDENT ID : 358888
 * EXR #27    : Arrays 4/4
 * CLASS      : CS1A
 * SECTION    : MW 8am
 * DUE DATE   : 11/18/24
 *****************************************************************************/

#include "header.h"

int FindAllInstances(int numArray[], const int AR_SIZE, int &elementValue)
{
    int index;
    int count;

    count  = 0;


    for (index = 0; index < AR_SIZE; index++)
    {

        if (numArray[index] == elementValue)
        {
            count++;
        }
    }

    if (count > 0)
    {
        cout << elementValue << " was found at index/indices: ";

        for (index = 0; index < AR_SIZE; index++)
        {

            if (numArray[index] == elementValue)
            {
                cout << index << " ";
            }
        }
    }

    return count;
}

```