**1}** The differences between an array and a stack are:

* An array stores elements of the same data type in a continuous block of memory while stack stores memory linearly.
* We can access arrays through their index and insert, delete, traverse through their indexes, while a stack only supports push and pop operations.
* Arrays require a fixed amount of memory while a stack can grow or shrink dynamically.

**2}** Stack:

* Stack is a linear data structure that follows last in first out principle.

Which means that when an element is added to the stack it goes to the top and if an element is to be popped the top element is given the first priority.

One computer application for stack is:

* Stacks can be used to store information about the function calls in a program, and when a function returns.

**4} (**L1.1)

* The code may compile but theres not enough space in the message char.
* The loop is till SIZE which means it will loop 5 times where as the arr can only hold 4 elements.

(L1.2)

* The array has 5 elements but the SIZE is initialized as 4.
* Array1 is declared with a size of 2 but it is initialized with 5 values.

(L1.3)

CODE:

#include <stdio.h>

int main() {

int arr[5] = {1, 2, 3, 4, 5};

// Calculate the number of bytes occupied by a single int

size\_t int\_size = sizeof(int);

printf("Size of int: %zu bytes\n", int\_size);

// Calculate the size of the entire array

size\_t arr\_size = sizeof(arr);

printf("Size of array: %zu bytes\n", arr\_size);

//To print the address of the element inside the array-

printf(“Address of the arr[0] : %p”, &arr[0]);

return 0;

}

(L1.4)

No the code will not run because String is not a valid type in C programming instead we can use char.