12/13/2024

**HANAN MAJEED**

**519166**

FUNDAMENTAL OF COMPUTER PROGRAMMING

LAB\_14

**Task A:**

Write a C++ program to test the following code snippet and show the output.

string course = "FoCP Course";

int ary[] = { 1, 3, 5, 7, 9, 11, 13 };

string\* ptr1 = &course;

int\* ptr2 = ary;

cout << ptr1 << endl;

cout << \*ptr1 << endl;

cout << (\*ptr1)[0] << endl;

cout << \*ptr2 << endl;

cout << (\*ptr1)[2] - \*ptr2 << endl;

cout << (\*ptr1+=3)[2] + \*ptr2+3;

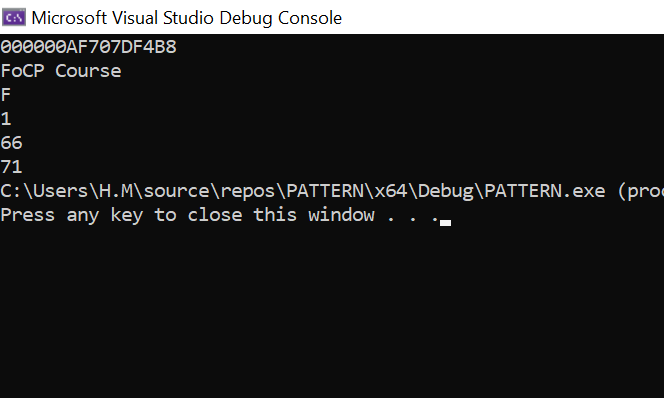
Provide your justification on why the specific output was for each cout statement

**Input:**

A screen shot of a computer program

Description automatically generated

**Output:**

****

**Justification:**

* **ptr1**:

ptr1 is a pointer to the string course. When we print ptr1, it shows the **memory address** of the string.

i.e:  
**000000AF707DF4B8**

* **\*ptr1**:

as the ptr1 is Dereferencing so it points to the actual string course, which is **"FoCP Course"**

* **(\*ptr1)[0]**:

(\*ptr1) is the string course, so (\*ptr1)[0] access the first character of the string. The first character of "FoCP Course" is **'F'.**

* **\*ptr2**:

ptr2 is a pointer to the first element of the array ary, which is 1. Dereferencing ptr2 (\*ptr2) gives the value **1.**

* **(\*ptr1)[2] - \*ptr2**:

(\*ptr1)[2] access the third character of the string course, which is 'C'. The ASCII value of 'C' is 67. \*ptr2 is the first element of ary, which is 1.

The difference between the ASCII value of 'C' (67) and 1 is 66. Hence, the output is **66.**

* **(\*ptr1 += 3)[2] + \*ptr2 + 3**:

(\*ptr1 += 3) increments ptr1 by 3. So, ptr1 now points to the string "P Course".

(\*ptr1)[2] access the third character of the new string "P Course", which is 'C'.as we know that the ASCII value of 'C' is 67.

\*ptr2 is still 1, and adding 3 to the result of dereferencing ptr2 gives 1 + 3 = 4.now, the final expression is 67 + 4, which equals **71.**

**Task B:**

Complete the following program by writing definitions of the rotateArray and printArray functions and then use them in the main program to show the required output.

// Function to rotate the array by k positions in both directions.

// Positive number should rotate to right and negative number to left

void rotateArray(int\* arr, int size, int k);

void printArray(const int\* arr, int size);

int main() {

int list[] = { 1, 2, 3, 4, 5, 6, 7, 8, 9 };

}

**Input:**

****

**Output:**

**A screenshot of a computer

Description automatically generated**

**Task C:**

A linked list is a dynamic data structure consisting of nodes, where each node contains:

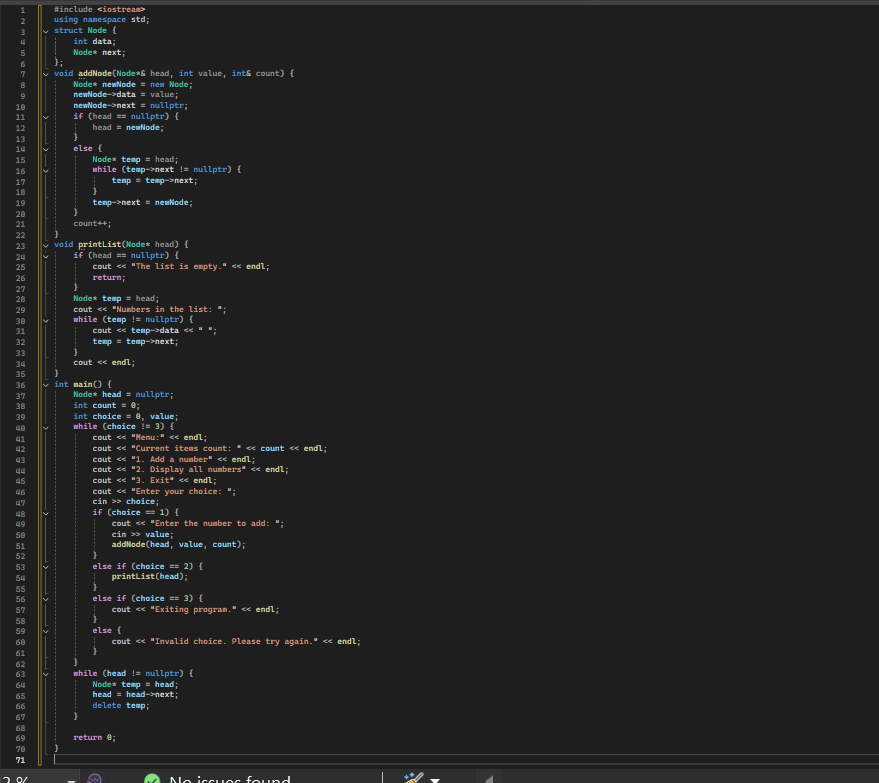
* Data: The value or information stored in the node.
* Pointer: A reference to the next node in the sequence

A diagram of a data flow

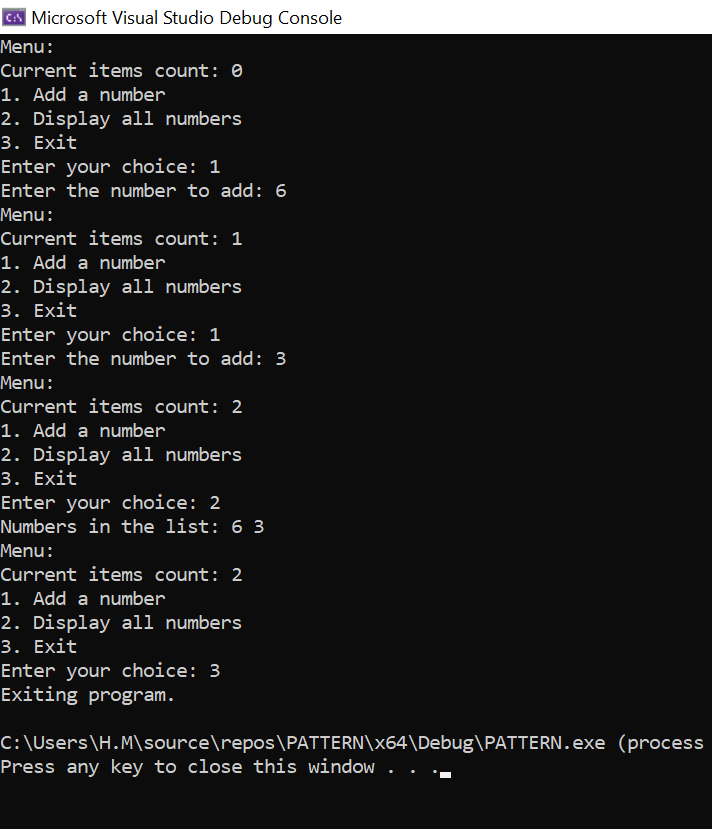
Description automatically generated

Write a program that stores the numbers entered by the user in a linked list. Each node of the list should be implemented as a structure, that should have integer data and a pointer to the next node. Use nullptr keyword to represent NULL. Initially, the list should be empty. The program should show a menu that allows the user to enter a number. Each number should be stored as a single node in the list. The second option should print the list.

**Input:**

****

**Output:**

****