**CS – Information Security (FALL 2022)**

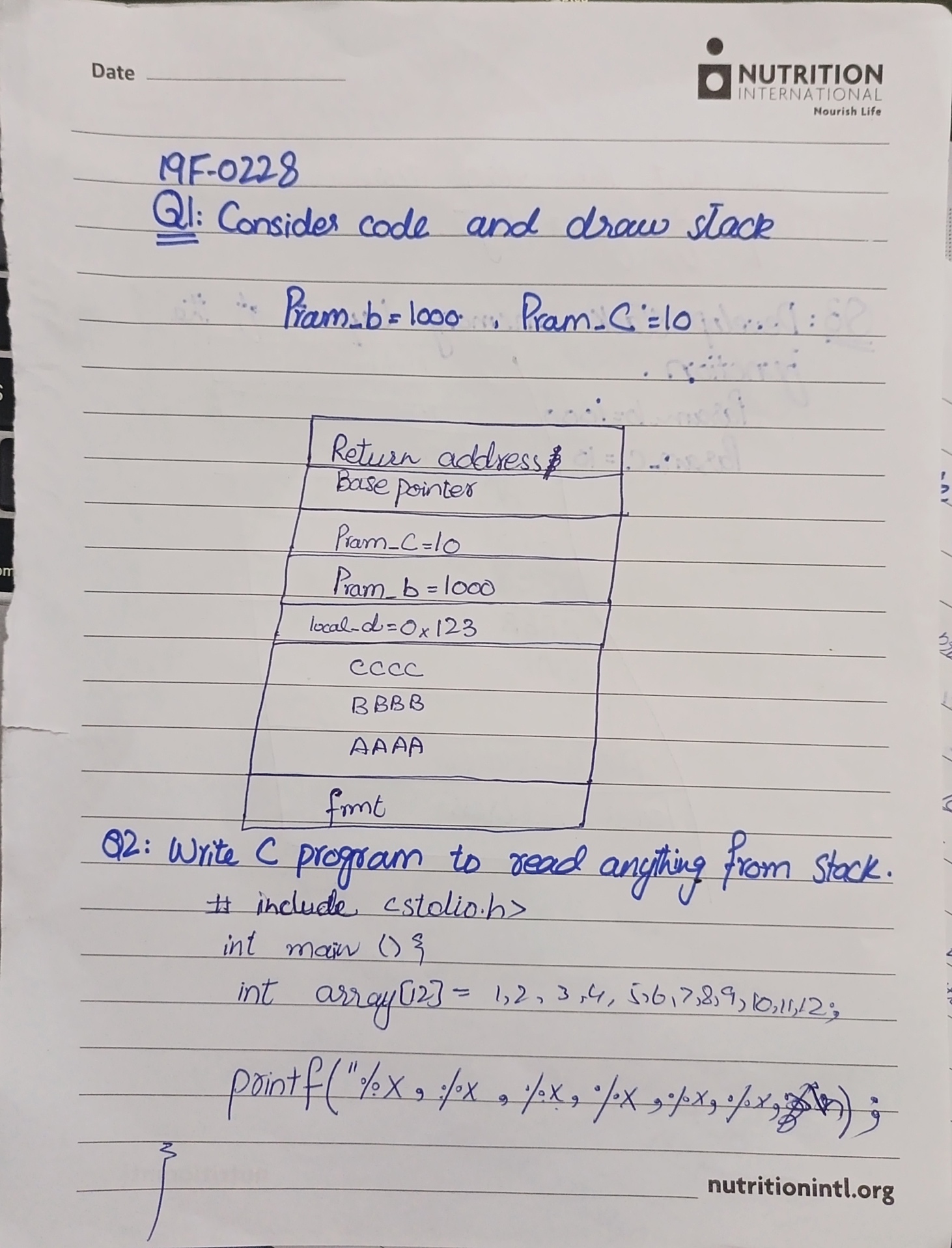
**19F-0228**

**Muhammad Zain**

# Question#01 (Format String)

Consider the above code and draw the stack memory layout of the stack of the give function with format String. (2)

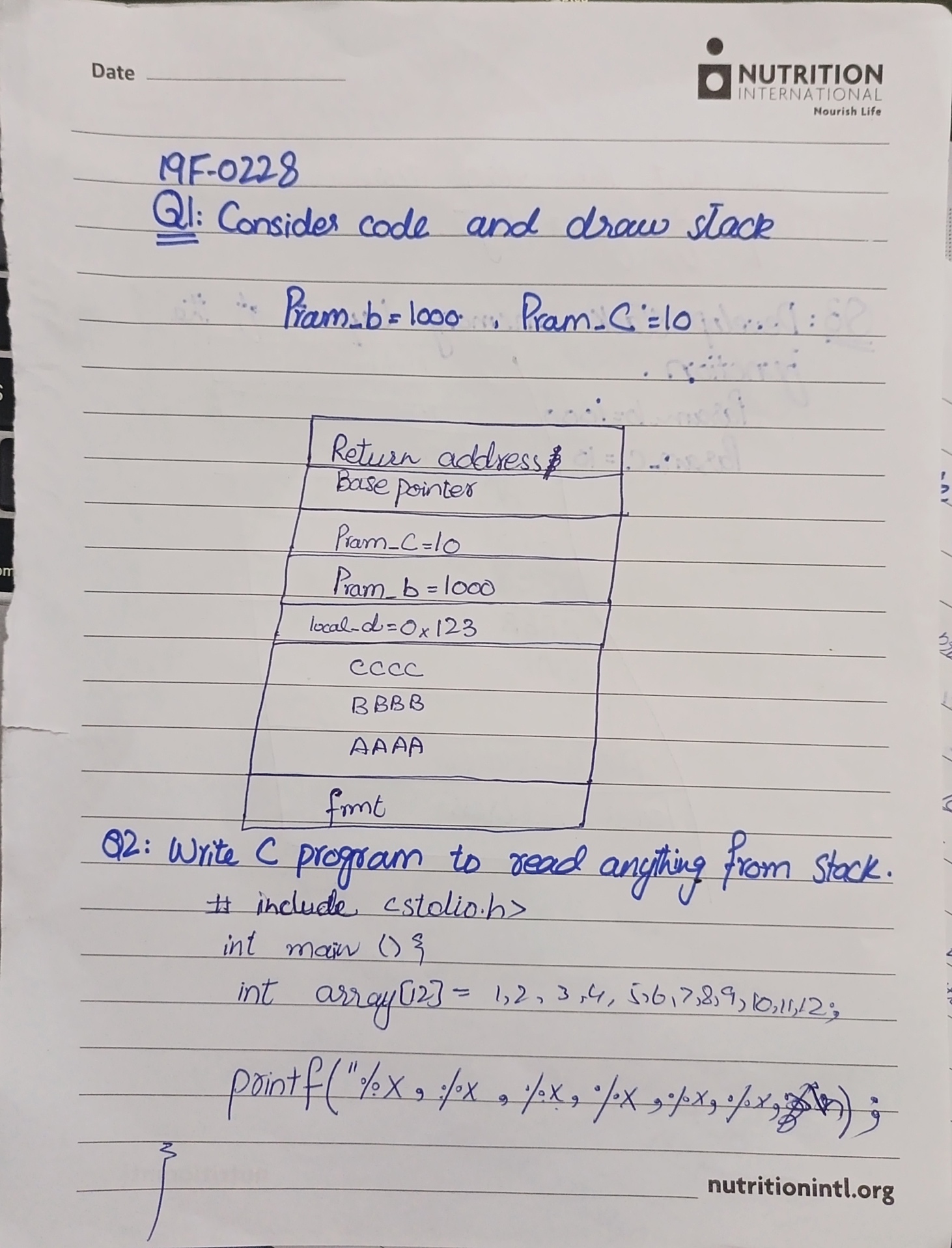
Hint: Param\_b = 1000 and Param\_c = 10 and draw memory stack when prinf is called

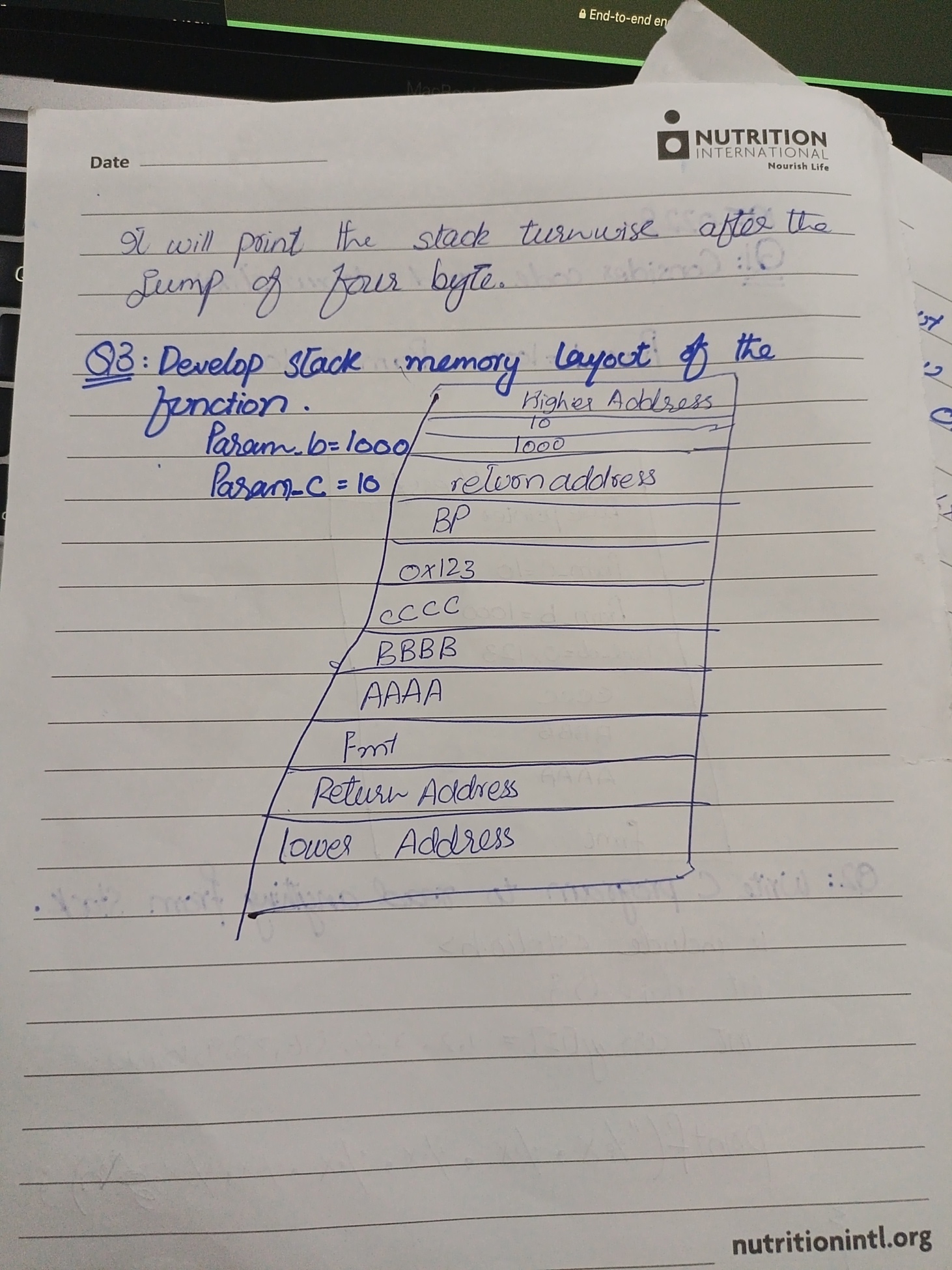


# Question#02 (Format String)

Write a C program to read anything on the stack using format String, compile the code and show your output. Discuss your observations?

***HINT: DON’T” pass arguments in the Printf function***



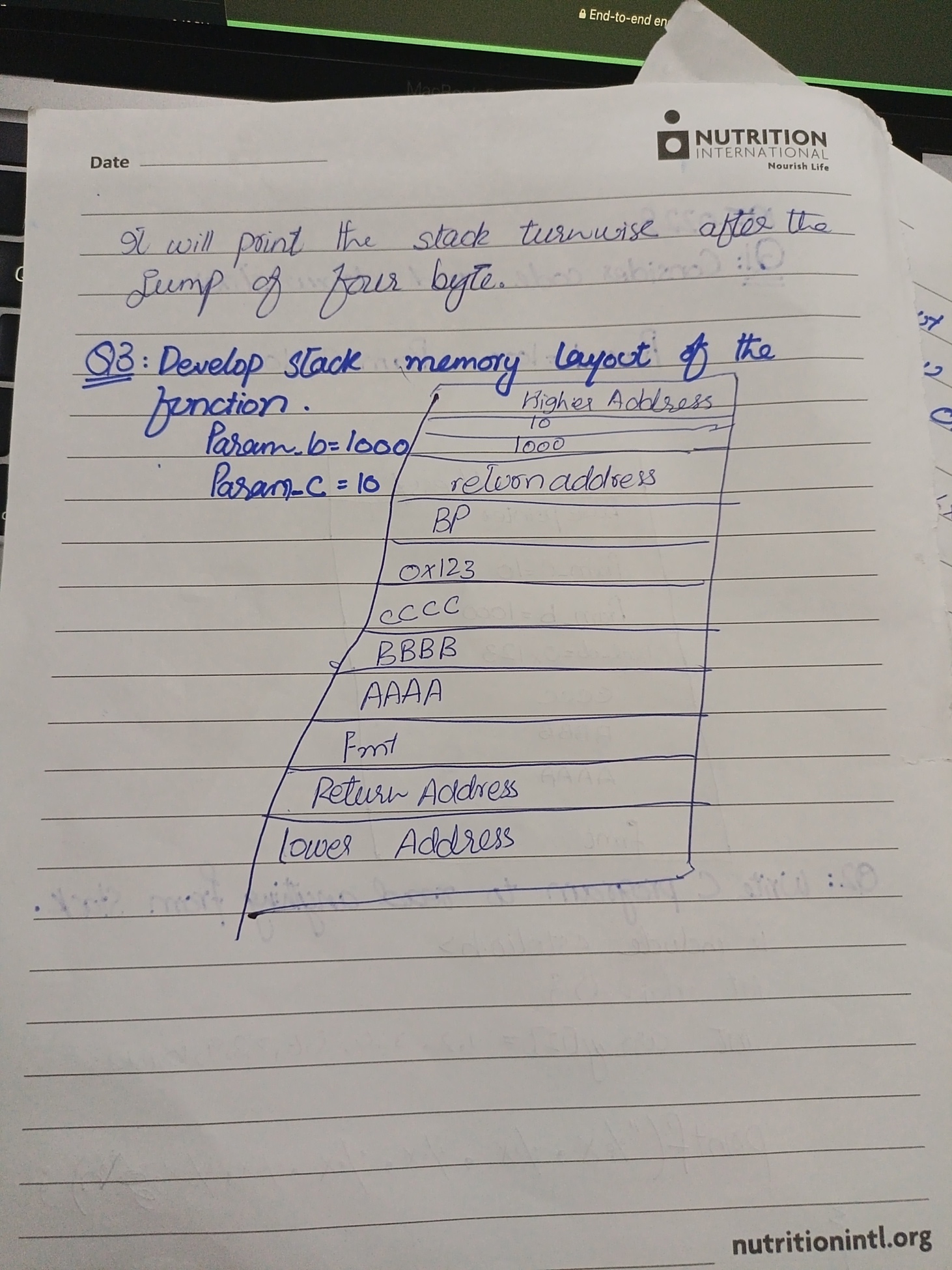


# Question#03 (Buffer overflow)

|  |
| --- |
| int func\_a(int param\_b, int param\_c)  { int local\_d = 0x123;  char local\_e[12] = "AAAABBBBCCCC";    printf("%x %x %x %x %x %x %x\n");  } |

Consider the above code and draw the stack memory layout of the stack of the give function. (3)

Hint: Param\_b = 1000 and Param\_c = 10



# Question#04 (Format String)

**Provide the input string which can change help you write “0xA551” at the place of 0xBB42 on the stack:**

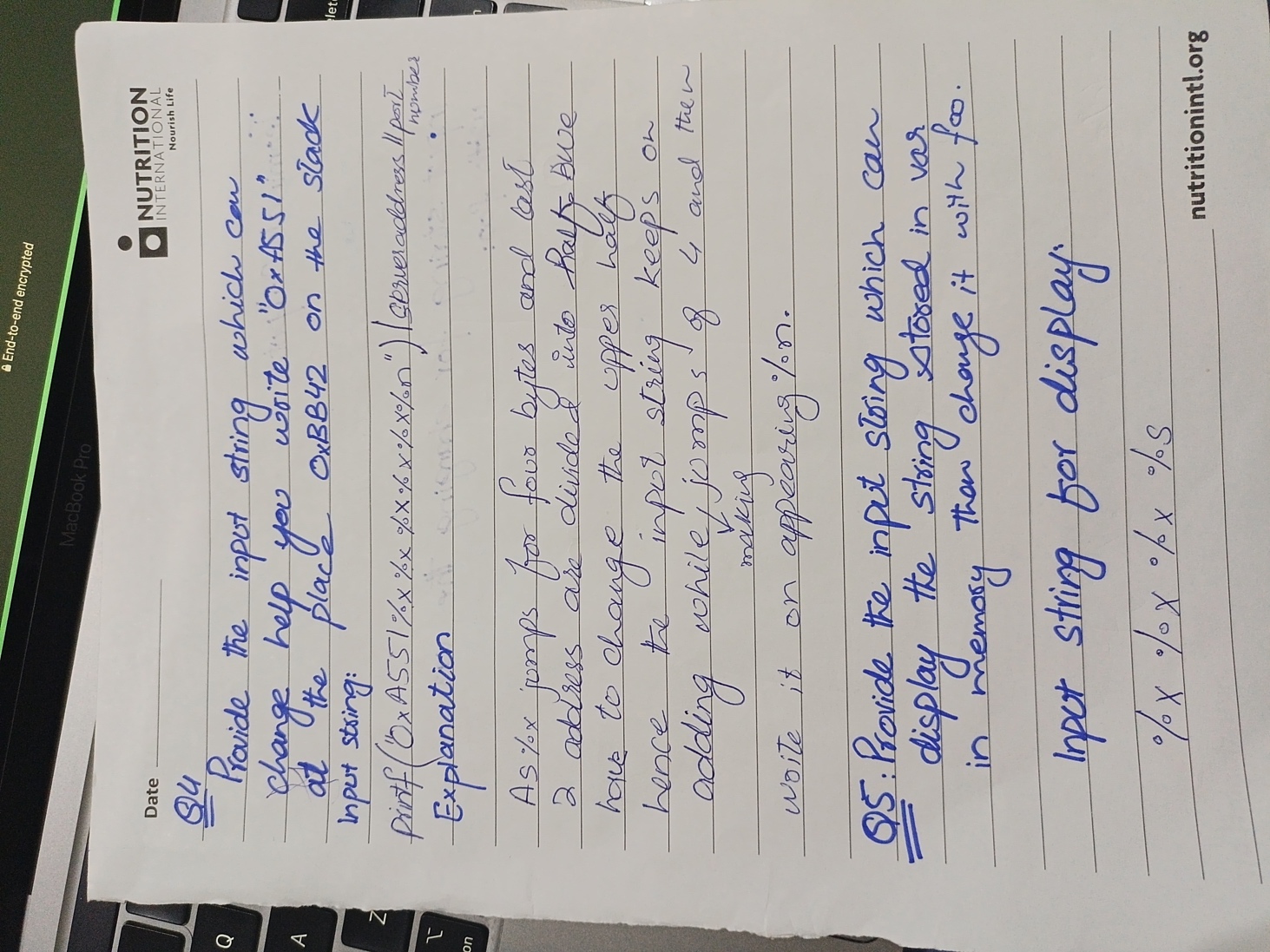
Diagram, table

Description automatically generated

Note: these are half addresses, each having 2 bytes.

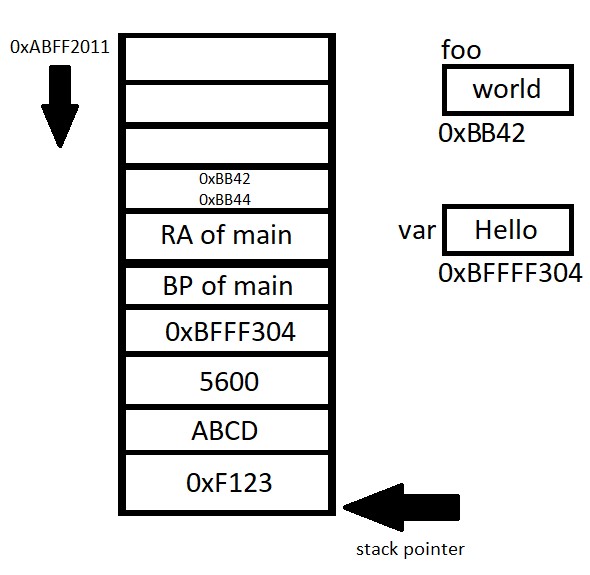
**Input string:**

**Explanation:**



# Question#05 (Format String)

**Provide the input string which can display the string stored in var in the memory block and then change it with foo:**



**Input string for display:**

**Terminal Output:**

**Input string for changing the var with foo:**

