Full Name: ______________________________ Net ID: __________________________________

CSO Honour Code
By turning in this exam for grading, I certify that I have produced the solution in accordance to the academic integrity policies of NYU Abu Dhabi as stated in `https://students.nyuad.nyu.edu/campus-life/community-standards/policies/academic-integrity`

Exam Instructions

1. You **must** follow all the x86-64 procedure call conventions.
2. You **must** show all steps of your work for full credit.
3. You **are** allowed to use a non-communicating calculator. You **cannot** use your phone, tablet, or laptop as a calculator. You **cannot** borrow a calculator from your colleagues.
4. You **are** allowed to have the reference sheet containing the summary of the x86-64 instructions, which I made available on Brightspace.
5. You **are** allowed one regular-size sheet of paper (front and back) on which you can write anything that you wish, as long as it is hand-written (i.e., not typed) by you (i.e., not by somebody else or a photocopy). You **are not** allowed to use any mechanical or electronic method of reproduction to create this sheet.

Question	Points	Score
Question 1	65	
Total:	65	

Question 1: (65 points)

You were recently hired by Xoople, a tech giant that has an x86-based search engine. It is your lucky day, and your team is responsible for improving the search engine by adding a feature that treats the search queries "weather today" and "today weather" equally. To achieve that, you must write x86-64 assembly code that implements the procedure `reverseWords` to reverse words in a given input phrase. For example, given the phrase "I know words I have the best words", the procedure will change it to "words best the have I words know I".

Since the search engine receives billions of queries every day, the procedure must process the input *in-place*, i.e., without using any additional memory or data structures. To solve this question, you must follow these instructions:

- A phrase is a null-terminated string that is represented by a character pointer.
- A word is a sequence of non-space characters.
- You can assume that the input phrase does not have any leading or trailing spaces, and that words are separated by exactly one space.
- To do the reversal in-place, your team must implement the 2-step algorithm that is shown below in the `C` declaration of the method. **Please note that you are not required to provide the `C` implementation of `reverseWords`.**

```
void reverseWords(char* phrase) {
  // Step 1: reverse the complete phrase

  // Step 2: reverse each word in the reversed phrase from Step 1
}
```