**Baku Higher Oil School**

**Process Automation Engineering Department**

**Programming in C**

**Laboratory 8 – Character and String**

**P.S** Add comment for each task; submit the file in LMS before the deadline.

1. Write a single statement to accomplish each of the following. Assume that variables c (which stores a character), x, y and z are of type int, variables d, e and f are of type double, variable ptr is of type char \* and arrays s1[100] and s2[100] are of type char.
2. Convert the character stored in variable c to an uppercase letter. Assign the result to variable c.
3. Determine whether the value of variable c is a digit. Use the conditional to print " is a " or " is not a " when the result is displayed.
4. Determine whether the value of variable c is a control character. Use the conditional operator to print " is a " or " is not a " when the result is displayed.
5. Read a line of text into array s1 from the keyboard. Do not use scanf.
6. Print the line of text stored in array s1. Do not use printf.
7. Assign ptr the location of the last occurrence of c in s1.
8. Print the value of variable c. Do not use printf.
9. Determine whether the value of c is a letter. Use the conditional operator to print " is a " or " is not a " when the result is displayed.
10. Read a character from the keyboard and store the character in variable c.
11. Assign ptr the location of the first occurrence of s2 in s1.
12. Determine whether the value of variable c is a printing character. Use the conditional operator to print " is a " or " is not a " when the result is displayed.
13. Read three double values into variables d, e and f from the string "1.27 10.3 9.432".
14. Copy the string stored in array s2 into array s1.
15. Assign ptr the location of the first occurrence in s1 of any character from s2.
16. Compare the string in s1 with the string in s2. Print the result.
17. Assign ptr the location of the first occurrence of c in s1
18. Use sprintf to print the values of integer variables x, y and z into array s1. Each value should be printed with a field width of 7.
19. Append 10 characters from the string in s2 to the string in s1.
20. Determine the length of the string in s1. Print the result.
21. Assign ptr to the location of the first token in s2. Tokens in the string s2 are separated

by commas (,).

1. Show two different ways to initialize character array vowel with the string of vowels "AEIOU"
2. **(Displaying Strings in Alternating Uppercase and Lowercase)** Write a program that inputs a line of text into char array s[100]. Output the line in alternate uppercase letters and lowercase letters.
3. **(Converting Strings to Integers for Calculations)** Write a program that inputs six strings that represent integers, converts the strings to integers, and calculates the sum and average of the six values.
4. **(Converting Strings to Floating Point for Calculations)** Write a program that inputs six strings that represent floating-point values, converts the strings to double values, stores the values into a double array and calculates the sum, and average of the values.
5. Write What, if anything, prints when each of the following C statements is performed? If the statement contains an error, describe the error and indicate how to correct it. Assume the following variable definitions:

char s1[50] = "jack", s2[50] = "jill", s3[50];

* 1. printf("%c%s", toupper(s1[0]), &s1[1]);
  2. printf("%s", strcpy(s3, s2));
  3. printf("%s", strcat(strcat(strcpy(s3, s1), " and "), s2));
  4. printf("%u", strlen(s1) + strlen(s2));
  5. printf("%u", strlen(s3)); // using s3 after part (c) executes

1. Find the error in each of the following program segments and explain how to correct it:
2. char s[10];

strncpy(s, "hello", 5);

printf("%s\n", s);

1. printf("%s", 'a');
2. char s[12];

strcpy(s, "Welcome Home");

1. if (strcmp(string1, string2)) {

puts("The strings are equal"); }

1. Write a program to find the first and last occurrence of character ‘e’ in this string:

“Teaching is an instruction or delivering a skill or subject”

1. Write a program to check whether a substring “teach” is present in the string:

“Educational methods include teaching, training and discussion”

1. Write a program to find the number of times a given word 'or' appears in the string.

“Teaching is an instruction or delivering a skill or subject”

1. Write a program to tokenize string using delimiter ‘@’

“Educational@ methods include tea@ching, tra@ining @nd discussion”

1. **(Concatenating Strings)** Write a program that uses function *strcat* to concatenate two strings provided by the user. The program should print the strings before and after concatenating as well as the length of the concatenated string.
2. **(Appending Part of a String)** Write a program that uses function strncat to append part of a string to another string. The program should input the strings, and the number of characters to be appended, then display the first string and its length after the second string was appended.
3. **(Pig Latin)** Write a program that encodes English-language phrases into pig Latin. Pig Latin is a form of coded language often used for amusement. Many variations exist in the methods used to form pig-Latin phrases. For simplicity, use the following algorithm:

To form a pig-Latin phrase from an English-language phrase, tokenize the phrase into words with function strtok. To translate each English word into a pig-Latin word, place the first letter of the English word at the end of the English word and add the letters "ay". Thus the word "jump" becomes "umpjay", the word "the" becomes "hetay" and the word "computer" becomes "omputercay". Blanks between words remain as blanks. Assume the following: The English phrase consists of words separated by blanks, there are no punctuation marks, and all words have two or more letters. Function printLatinWord should display each word. [Hint: Each time a token is found in a call to strtok, pass the token pointer to function printLatinWord, and print the pig-Latin word. Note: We’ve provided simplified rules for converting words to pig Latin here. For more detailed rules and variations, visit en.wikipedia.org/wiki/Pig\_latin.]

1. **(Tokenizing Telephone Numbers)** Write a program that inputs a telephone number as a string in the form (555) 555-5555. The program should use function strtok to extract the area code as a token, the first three digits of the phone number as a token and the last four digits of the phone number as a token. The seven digits of the phone number should be concatenated into one string.
2. **(Displaying a Sentence with Its Words Reversed)** Write a program that inputs a line of text, tokenizes the line with function strtok and outputs the tokens in reverse order.
3. **(Removing a Particular Word From a Given Line of Text)** Write a program that inputs a line of text and a given word. The program should use string library functions strcmp and strcpy to remove all occurrences of the given word from the input line of text. The program should also count the number of words in the given line of text before and after removing the given word using the strtok function.
4. **(Counting the Number of Words in a String)** Write a program that inputs several lines of text and uses strtok to count the total number of words. Assume that the words are separated by either spaces or newline characters.
5. **(Strings Starting with "Th")** Write a program that reads a series of strings and prints only those beginning with the letters "Th".
6. **(Write Your Own String-Copy and String-Concatenation Functions)** Write two versions of each of the string-copy and string-concatenation functions. The first version should use array indexing, and the second should use pointers and pointer arithmetic
7. **(Printing Dates in Various Formats)** Dates are commonly printed in several different formats in business correspondence. Two of the more common formats are

Write a program that reads a date in the first format and prints it in the second format.