CSC3320 System Level Programming
Lab Assignment 9 - Post-Lab

Due at 11:59 pm on Sunday, March 21, 2021

Purpose: Learn how to use array in C. Understand the basic memory address in C.

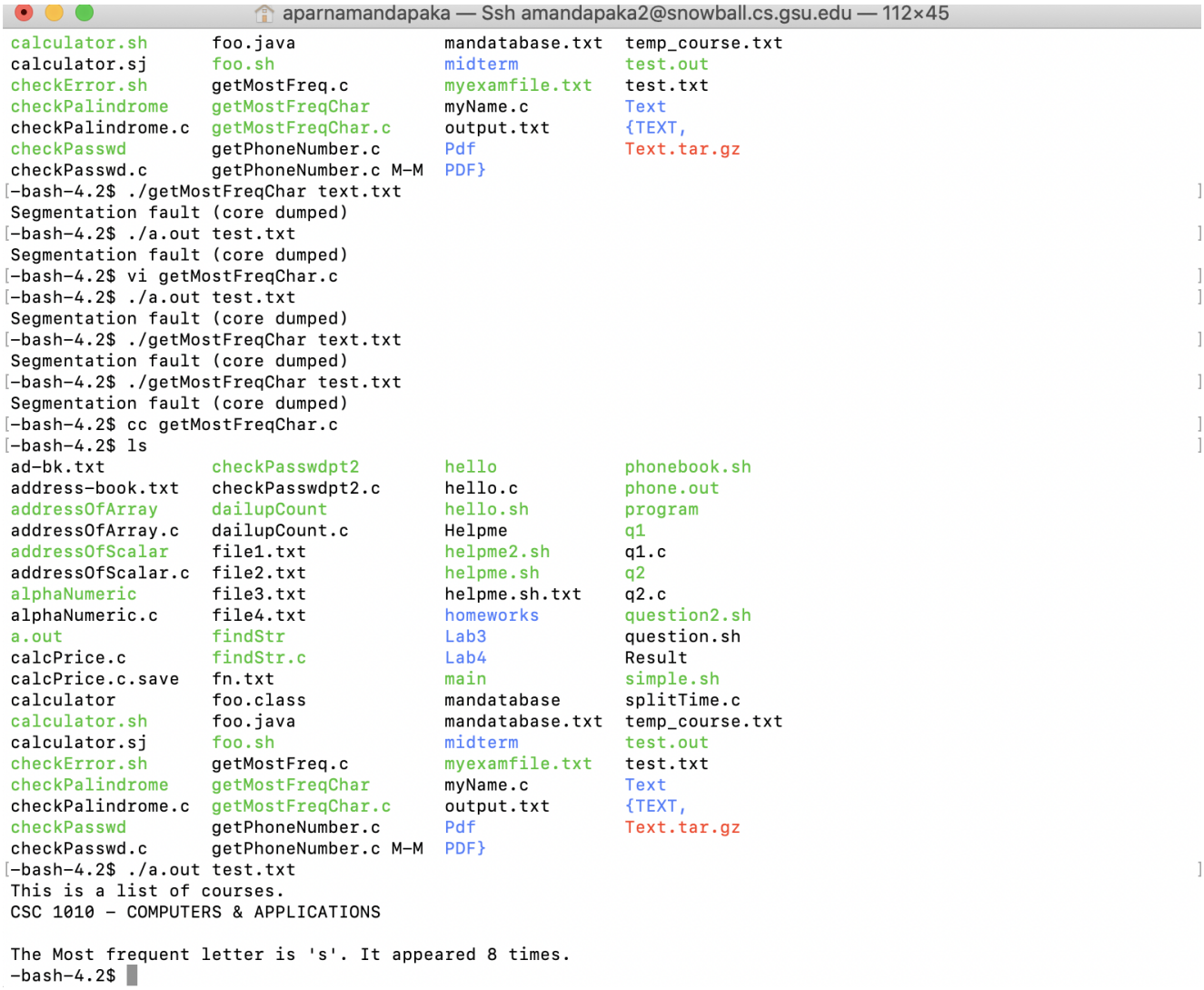
Part 1:

Write a C program named as getMostFreqChar.c that finds the most frequent letter from the input via ignoring the case sensitive and prints out its frequency. For example, sample outputs could be like below

```
$cat test.txt
This is a list of courses.
CSC 1010 - COMPUTERS & APPLICATIONS

$./getMostFreqChar test.txt
The most frequent letter is 's'. It appeared 8 times. Run the C program,
```

attach a screenshot of the output in the answer sheet.


aparnamandapaka — Ssh amandapaka2@snowball.cs.gsu.edu — 112×45
calculator.sh foo.java mandatabase.txt temp_course.txt
calculator.sj foo.sh midterm test.out
checkError.sh getMostFreq.c myexamfile.txt test.txt
checkPalindrome getMostFreqChar myName.c Text
checkPalindrome.c getMostFreqChar.c output.txt {TEXT,
checkPasswd getPhoneNumber.c Pdf Text.tar.gz
checkPasswd.c getPhoneNumber.c M-M PDF}
-bash-4.2$ ./getMostFreqChar text.txt
Segmentation fault (core dumped)
-bash-4.2$ ./a.out test.txt
Segmentation fault (core dumped)
-bash-4.2$ vi getMostFreqChar.c
-bash-4.2$ ./a.out test.txt
Segmentation fault (core dumped)
-bash-4.2$ ./getMostFreqChar text.txt
Segmentation fault (core dumped)
-bash-4.2$ ./getMostFreqChar test.txt
Segmentation fault (core dumped)
-bash-4.2$ cc getMostFreqChar.c
-bash-4.2$ ls
ad-bk.txt checkPasswdpt2 hello phonebook.sh
address-book.txt checkPasswdpt2.c hello.c phone.out
addressOfArray dailupCount hello.sh program
addressOfArray.c dailupCount.c Helpme q1
addressOfScalar file1.txt helpme2.sh q1.c
addressOfScalar.c file2.txt helpme.sh q2
alphaNumeric file3.txt helpme.sh.txt q2.c
alphaNumeric.c file4.txt homeworks question2.sh
a.out findStr Lab3 question.sh
calcPrice.c findStr.c Lab4 Result
calcPrice.c.save fn.txt main simple.sh
calculator foo.class mandatabase splitTime.c
calculator.sh foo.java mandatabase.txt temp_course.txt
calculator.sj foo.sh midterm test.out
checkError.sh getMostFreq.c myexamfile.txt test.txt
checkPalindrome getMostFreqChar myName.c Text
checkPalindrome.c getMostFreqChar.c output.txt {TEXT,
checkPasswd getPhoneNumber.c Pdf Text.tar.gz
checkPasswd.c getPhoneNumber.c M-M PDF}
-bash-4.2$ ./a.out test.txt
This is a list of courses.
CSC 1010 – COMPUTERS & APPLICATIONS
The Most frequent letter is 's'. It appeared 8 times.
-bash-4.2$

Part 2:

When a variable is stored in memory, it is associated with an address. To obtain the address of a variable, the & operator can be used. For example, &a gets the memory address of variable a. Let's try some examples.
Write a C program addressOfScalar.c by inserting the code below in the main function.

Questions:
1) Run the C program, attach a screenshot of the output in the answer

sheet.

```
-bash-4.2$ ./addressOfScalar
address of charvar = 0x7ffc6baaecdf
address of charvar - 1 = 0x7ffc6baaecde
address of charvar + 1 = 0x7ffc6baaece0
address of intvar = 0x7ffc6baaecd8
address of intvar - 1 = 0x7ffc6baaecd4
address of intvar + 1 = 0x7ffc6baaecdc
-bash-4.2$
```

2) Attach the source code in the answer sheet

aparnamandapaka — Ssh amandapaka2@snowball.cs.gsu.edu — 93×31

```c
#include <stdio.h>
int main(){
char charvar = '\0';
printf("address of charvar = %p\n", (void *)(&charvar));
printf("address of charvar - 1 = %p\n", (void *)(&charvar - 1));
printf("address of charvar + 1 = %p\n", (void *)(&charvar + 1));
int intvar = 1;
printf("address of intvar = %p\n", (void *)(&intvar));
printf("address of intvar - 1 = %p\n", (void *)(&intvar - 1));
printf("address of intvar + 1 = %p\n", (void *)(&intvar + 1));
}
```

2) Then explain why the address after intvar is incremented by 4 bytes instead of 1 byte.

1 // intialize a char variable, print its address and the next address 2 char charvar = '\0';
3 printf("address of charvar = %p\n", (void *)(&charvar));
4 printf("address of charvar - 1 = %p\n", (void *)(&charvar - 1)); 5 printf("address of charvar + 1 = %p\n", (void *)(&charvar + 1)); 6
7 // intialize an int variable, print its address and the next address 8 int intvar = 1;
9 printf("address of intvar = %p\n", (void *)(&intvar));
10 printf("address of intvar - 1 = %p\n", (void *)(&intvar - 1)); 11 printf("address of intvar + 1 = %p\n", (void *)(&intvar + 1)); 12

This intvar would be incremented by 4 bytes instead of 1 because the intvar takes 4

bytes of memory, whereas the char takes 1 byte of memory.

Part 3:

Write a C program addressOfArray.c by inserting the code below in the main function.

```
1 // initialize an array of ints
2 int numbers[5] = {1,2,3,4,5};
3 int i = 0;
4
5 // print the address of the array variable
6 printf("numbers = %p\n", numbers);
7
8 // print addresses of each array index
9 do {
10 printf("numbers[%u] = %p\n", i, (void *)(&numbers[i]));
11 i++;
12 } while(i < 5);
        // print the size of the array
        printf("sizeof(numbers) = %lu\n", sizeof(numbers));
```

Questions:

1) Run the C program, attach a screenshot of the output in the answer sheet.

```
-bash-4.2$ vi addressOfArray.c
-bash-4.2$ gcc -o addressOfArray -g addressOfArray.c
-bash-4.2$ ./addressOfArray
numbers = 0x7fff55ff8030
numbers[0] = 0x7fff55ff8030
numbers[1] = 0x7fff55ff8034
numbers[2] = 0x7fff55ff8038
numbers[3] = 0x7fff55ff803c
numbers[4] = 0x7fff55ff8040
sizeof(numbers)= 20
length(numbers)= 5
-bash-4.2$
```

2) Check the address of the array and the address of the first element in the array. Are they the same?

Yes, they are same

3) Write down the statement to print out the length of the array by using sizeof

operator.

printf(“length of the (numbers) = %lu\n”, sizeof(numbers)/sizeof(numbers[0]);

Submission:

- Upload an electronic copy (pdf) of your answer sheet to the folder named “Lab 9” in Google Classroom
- Please add the lab assignment number and your name at the top of your answer sheet.
- Upload the C files getMostFreqChar.c, addressOfArray.c and addressOfScalar.c to the folder named named “Lab 9” in Google Classroom
- Name your file in the format of Lab9_ FirstnameLastname (e.g Lab9_FilRondel.pdf)