Inspiring Excellence

LAB ASSIGNMENT: 01

SPRING 2024

COURSE CODE: CSE321

COURSE TITLE: Operating System

PREPARED BY

Name: Nur-E-Jannat

ID: 21301744

Section: 12

Email: nur.e.jannat@g.bracu.ac.bd

1.

a. touch 21301744_1.txt 21301744_2.txt 21301744_3.txt 21301744_4.txt

mkdir jannat1 jannat2

cp 21301744_1.txt 21301744_2.txt jannat2/

mv 21301744_1.txt 21301744_2.txt 21301744_3.txt jannat1/

mkdir jannat3

cp -r jannat1 jannat3

cd jannat3

ls -l

chmod go=rx jannat1/*

chmod go=rx jannat1

cd ..

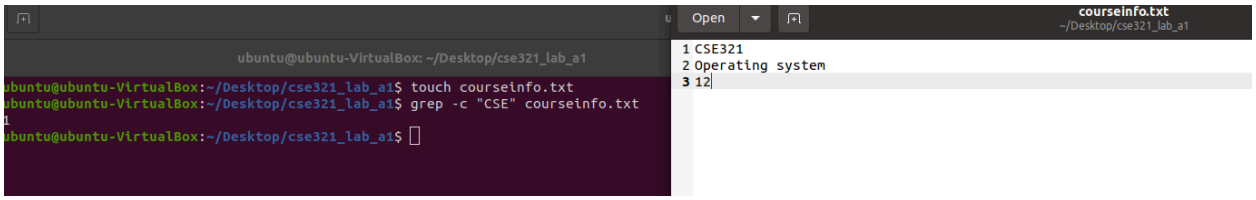
ls -la

mv jannat3/

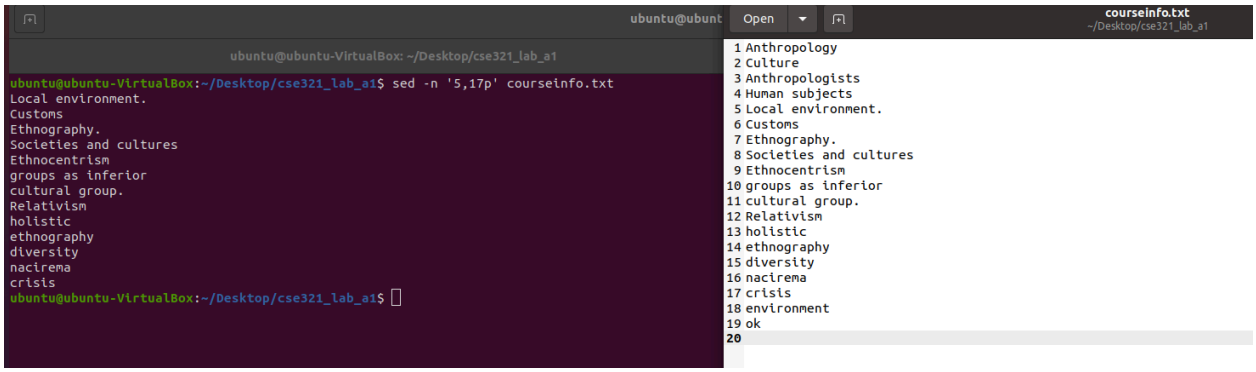
rm -rf *

```
ubuntu@ubuntu-VirtualBox:~$ cd Desktop
ubuntu@ubuntu-VirtualBox:~/Desktop$ mkdir cse321_lab_a1
ubuntu@ubuntu-VirtualBox:~/Desktop$ cd cse321_lab_a1
ubuntu@ubuntu-VirtualBox:~/Desktop/cse321_lab_a1$ touch 21301744_1.txt 21301744_2.txt 21301744_3.txt 21301744_4.txt
ubuntu@ubuntu-VirtualBox:~/Desktop/cse321_lab_a1$ mkdir jannat1 jannat2
ubuntu@ubuntu-VirtualBox:~/Desktop/cse321_lab_a1$ cp 21301744_1.txt 21301744_2.txt jannat2/
ubuntu@ubuntu-VirtualBox:~/Desktop/cse321_lab_a1$ mv 21301744_1.txt 21301744_2.txt 21301744_3.txt jannat1/
ubuntu@ubuntu-VirtualBox:~/Desktop/cse321_lab_a1$ mkdir jannat3
ubuntu@ubuntu-VirtualBox:~/Desktop/cse321_lab_a1$ cp -r jannat1 jannat3
ubuntu@ubuntu-VirtualBox:~/Desktop/cse321_lab_a1$ cd jannat3
ubuntu@ubuntu-VirtualBox:~/Desktop/cse321_lab_a1/jannat3$ ls -l
total 4
drwxrwxr-x 2 ubuntu ubuntu 4096 ফেব      29 00:40 jannat1
ubuntu@ubuntu-VirtualBox:~/Desktop/cse321_lab_a1/jannat3$ chmod go=rx jannat1/*
ubuntu@ubuntu-VirtualBox:~/Desktop/cse321_lab_a1/jannat3$ chmod go=rx jannat1
ubuntu@ubuntu-VirtualBox:~/Desktop/cse321_lab_a1/jannat3$ ls -l
total 4
drwxr-xr-x 2 ubuntu ubuntu 4096 ফেব      29 00:40 jannat1
ubuntu@ubuntu-VirtualBox:~/Desktop/cse321_lab_a1/jannat3$ cd ..
ubuntu@ubuntu-VirtualBox:~/Desktop/cse321_lab_a1$ ls -la
total 20
drwxrwxr-x 5 ubuntu ubuntu 4096 ফেব      29 00:39 .
drwxr-xr-x 4 ubuntu ubuntu 4096 ফেব      29 00:37 ..
-rw-rw-r-- 1 ubuntu ubuntu    0 ফেব      29 00:38 21301744_4.txt
drwxrwxr-x 2 ubuntu ubuntu 4096 ফেব      29 00:39 jannat1
drwxrwxr-x 2 ubuntu ubuntu 4096 ফেব      29 00:38 jannat2
drwxrwxr-x 3 ubuntu ubuntu 4096 ফেব      29 00:40 jannat3
ubuntu@ubuntu-VirtualBox:~/Desktop/cse321_lab_a1$ mv jannat3 /
mv: cannot move 'jannat3' to '/jannat3': Permission denied
ubuntu@ubuntu-VirtualBox:~/Desktop/cse321_lab_a1$ cd jannat3
ubuntu@ubuntu-VirtualBox:~/Desktop/cse321_lab_a1/jannat3$ mv jannat3 /
mv: cannot stat 'jannat3': No such file or directory
ubuntu@ubuntu-VirtualBox:~/Desktop/cse321_lab_a1/jannat3$ cd ..
ubuntu@ubuntu-VirtualBox:~/Desktop/cse321_lab_a1$ rm -rf *
ubuntu@ubuntu-VirtualBox:~/Desktop/cse321_lab_a1$
```

b. touch courseinfo.txt
grep -c "CSE" courseinfo.txt



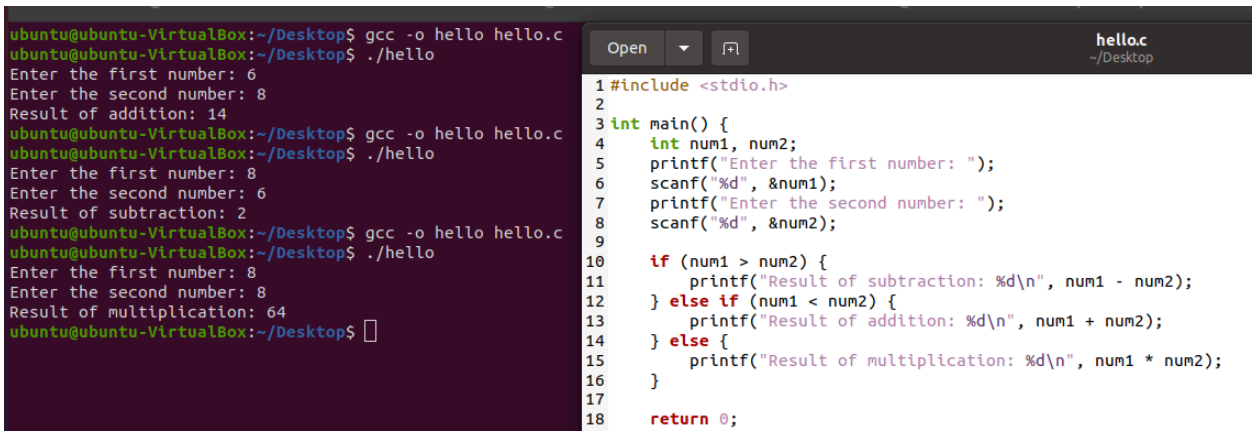
c. sed -n '5,17p' courseinfo.txt



2.

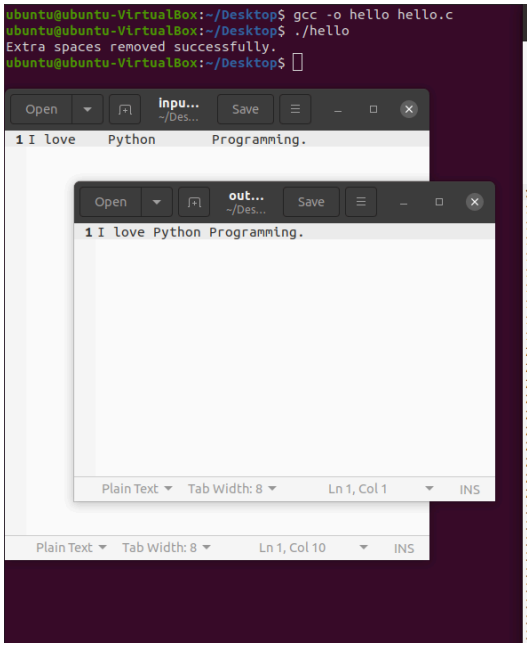
a.

```
#include <stdio.h>

int main() {
    int num1, num2;
    printf("Enter the first number: ");
    scanf("%d", &num1);
    printf("Enter the second number: ");
    scanf("%d", &num2);

    if (num1 > num2) {
        printf("Result of subtraction: %d\n", num1 - num2);
    } else if (num1 < num2) {
        printf("Result of addition: %d\n", num1 + num2);
    } else {
        printf("Result of multiplication: %d\n", num1 * num2);
    }

    return 0;
}
```



b.

```
#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>

void remove_extra_spaces(const char *input_file_name, const char *output_file_name)
{
    FILE *input_file = fopen(input_file_name, "r");
    FILE *output_file = fopen(output_file_name, "w");

    if (input_file == NULL || output_file == NULL) {
        printf("Error.\n");
        return;
    }

    char prev_char = ' ';
    char current_char;

    while ((current_char = fgetc(input_file)) != EOF) {
        if (!isspace(prev_char) || !isspace(current_char)) {
            fputc(current_char, output_file);
        }
        prev_char = current_char;
    }

    fclose(input_file);
    fclose(output_file);

    printf("Extra spaces removed successfully.\n");
}

int main() {
    const char *input_file_name = "input.txt";
    const char *output_file_name = "output.txt";

    remove_extra_spaces(input_file_name, output_file_name);

    return 0;
}
```



Open hello.c ~/Desktop Save

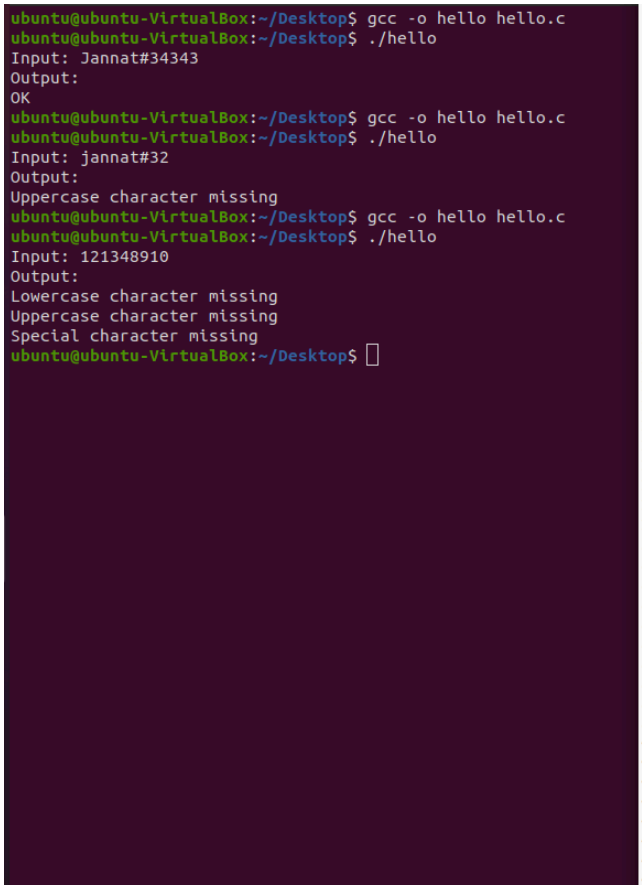
```c
#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>

void remove_extra_spaces(const char *input_file_name, const char *output_file_name) {
    FILE *input_file = fopen(input_file_name, "r");
    FILE *output_file = fopen(output_file_name, "w");

    if (input_file == NULL || output_file == NULL) {
        printf("Error.\n");
        return;
    }

    char prev_char = ' ';
    char current_char;

    while ((current_char = fgetc(input_file)) != EOF) {
        if (!isspace(prev_char) || !isspace(current_char)) {
            fputc(current_char, output_file);
        }
        prev_char = current_char;
    }

    fclose(input_file);
    fclose(output_file);

    printf("Extra spaces removed successfully.\n");
}

int main() {
    const char *input_file_name = "input.txt";
    const char *output_file_name = "output.txt";

    remove_extra_spaces(input_file_name, output_file_name);

    return 0;
}
```

c.

```
#include <stdio.h>
#include <string.h>
#include <ctype.h>

int is_lowercase(const char *password) {
    for (int i = 0; password[i] != '\0'; i++) {
        if (islower(password[i])) {
            return 1;
        }
    }
    return 0;
}

int is_uppercase(const char *password) {
    for (int i = 0; password[i] != '\0'; i++) {
        if (isupper(password[i])) {
            return 1;
        }
    }
    return 0;
}

int is_digit(const char *password) {
    for (int i = 0; password[i] != '\0'; i++) {
        if (isdigit(password[i])) {
            return 1;
        }
    }
    return 0;
}

int is_special_char(const char *password) {
    const char *special_char = "_,$#@";
    for (int i = 0; password[i] != '\0'; i++) {
        if (strchr(special_char, password[i]) != NULL) {
            return 1;
        }
```

```
    }
    return 0;
}

void check_password(const char *password) {
    int lowercase_present = is_lowercase(password);
    int uppercase_present = is_uppercase(password);
    int digit_present = is_digit(password);
    int special_char_present = is_special_char(password);

    if (!lowercase_present) {
        printf("Lowercase character missing\n");
    }
    if (!uppercase_present) {
        printf("Uppercase character missing\n");
    }
    if (!digit_present) {
        printf("Digit missing\n");
    }
    if (!special_char_present) {
        printf("Special character missing\n");
    }

    if (lowercase_present && uppercase_present && digit_present &&
special_char_present) {
        printf("OK\n");
    }
}

int main() {
    char password[100];

    printf("Input: ");
    fgets(password, sizeof(password), stdin);

    if (password[strlen(password) - 1] == '\n') {
        password[strlen(password) - 1] = '\0';
    }

    printf("Output:\n");
```

```
    check_password(password);

    return 0;
}
```




```c
#include <ctype.h>

int is_lowercase(const char *password) {
    for (int i = 0; password[i] != '\0'; i++) {
        if (islower(password[i])) {
            return 1;
        }
    }
    return 0;
}

int is_uppercase(const char *password) {
    for (int i = 0; password[i] != '\0'; i++) {
        if (isupper(password[i])) {
            return 1;
        }
    }
    return 0;
}

int is_digit(const char *password) {
    for (int i = 0; password[i] != '\0'; i++) {
        if (isdigit(password[i])) {
            return 1;
        }
    }
    return 0;
}

int is_special_char(const char *password) {
    const char *special_char = "_,$#@";
    for (int i = 0; password[i] != '\0'; i++) {
        if (strchr(special_char, password[i]) != NULL) {
            return 1;
        }
    }
    return 0;
}

void check_password(const char *password) {
    int lowercase_present = is_lowercase(password);
    int uppercase_present = is_uppercase(password);
    int digit_present = is_digit(password);
    int special_char_present = is_special_char(password);

```

d.

```
#include <stdio.h>
#include <string.h>

int is_updated_email(const char *email) {
    const char *old_domain = "@kaaj.com";
    const char *new_domain = "@sheba.xyz";


    const char *at_symbol = strchr(email, '@');
```

```
    if (at_symbol == NULL) {
        return 0;
    }


    if (strcmp(at_symbol, old_domain) == 0) {
        return 0;
    } else if (strcmp(at_symbol, new_domain) == 0) {
        return 1;
    } else {
        return 0;
    }
}

int main() {
    char email[100];

    printf("Enter the email address: ");
    scanf("%s", email);

    if (is_updated_email(email)) {
        printf("Email address is okay\n");
    } else {
        printf("Email address is outdated\n");
    }

    return 0;
}
```

```
ubuntu@ubuntu-VirtualBox:~/Desktop$ gcc -o hello hello.c
ubuntu@ubuntu-VirtualBox:~/Desktop$ ./hello
Enter the email address: fahmid@kaaj.com
Email address is outdated
ubuntu@ubuntu-VirtualBox:~/Desktop$ gcc -o hello hello.c
ubuntu@ubuntu-VirtualBox:~/Desktop$ ./hello
Enter the email address: zaki@sheba.xyz
Email address is okay
ubuntu@ubuntu-VirtualBox:~/Desktop$
```

```c
#include <stdio.h>
#include <string.h>

int is_updated_email(const char *email) {
    const char *old_domain = "@kaaj.com";
    const char *new_domain = "@sheba.xyz";


    const char *at_symbol = strchr(email, '@');
    if (at_symbol == NULL) {
        return 0;
    }


    if (strcmp(at_symbol, old_domain) == 0) {
        return 0;
    } else if (strcmp(at_symbol, new_domain) == 0) {
        return 1;
    } else {
        return 0;
    }
}

int main() {
    char email[100];

    printf("Enter the email address: ");
    scanf("%s", email);

    if (is_updated_email(email)) {
        printf("Email address is okay\n");
    } else {
        printf("Email address is outdated\n");
    }

    return 0;
}

```

e.

```c
#include <stdio.h>
#include <string.h>
#include <ctype.h>

int is_palindrome(const char *str) {
    const char *start = str;
    const char *end = str + strlen(str) - 1;

    while (start < end) {
        while (!isalnum(*start) && start < end) {
            start++;
        }
        while (!isalnum(*end) && start < end) {
            end--;
        }

        if (tolower(*start) != tolower(*end)) {
```

```
            return 0;
        }

        start++;
        end--;
    }

    return 1;
}

int main() {
    char input[100];

    printf("Enter a string: ");
    fgets(input, sizeof(input), stdin);
    if (input[strlen(input) - 1] == '\n') {
        input[strlen(input) - 1] = '\0';
    }

    if (is_palindrome(input)) {
        printf("Palindrome\n");
    } else {
        printf("Not Palindrome\n");
    }

    return 0;
}
```

```
ubuntu@ubuntu-VirtualBox:~/Desktop$ gcc -o hello hello.c
ubuntu@ubuntu-VirtualBox:~/Desktop$ ./hello
Enter a string: 6778uhu
Not Palindrome
ubuntu@ubuntu-VirtualBox:~/Desktop$ gcc -o hello hello.c
ubuntu@ubuntu-VirtualBox:~/Desktop$ ./hello
Enter a string: abccba
Palindrome
ubuntu@ubuntu-VirtualBox:~/Desktop$
```

```c
#include <stdio.h>
#include <string.h>
#include <ctype.h>

int is_palindrome(const char *str) {
    const char *start = str;
    const char *end = str + strlen(str) - 1;

    while (start < end) {
        while (!isalnum(*start) && start < end) {
            start++;
        }
        while (!isalnum(*end) && start < end) {
            end--;
        }

        if (tolower(*start) != tolower(*end)) {
            return 0;
        }

        start++;
        end--;
    }

    return 1;
}

int main() {
    char input[100];

    printf("Enter a string: ");
    fgets(input, sizeof(input), stdin);
    if (input[strlen(input) - 1] == '\n') {
        input[strlen(input) - 1] = '\0';
    }

    if (is_palindrome(input)) {
        printf("Palindrome\n");
    } else {
        printf("Not Palindrome\n");
    }

    return 0;
}

```