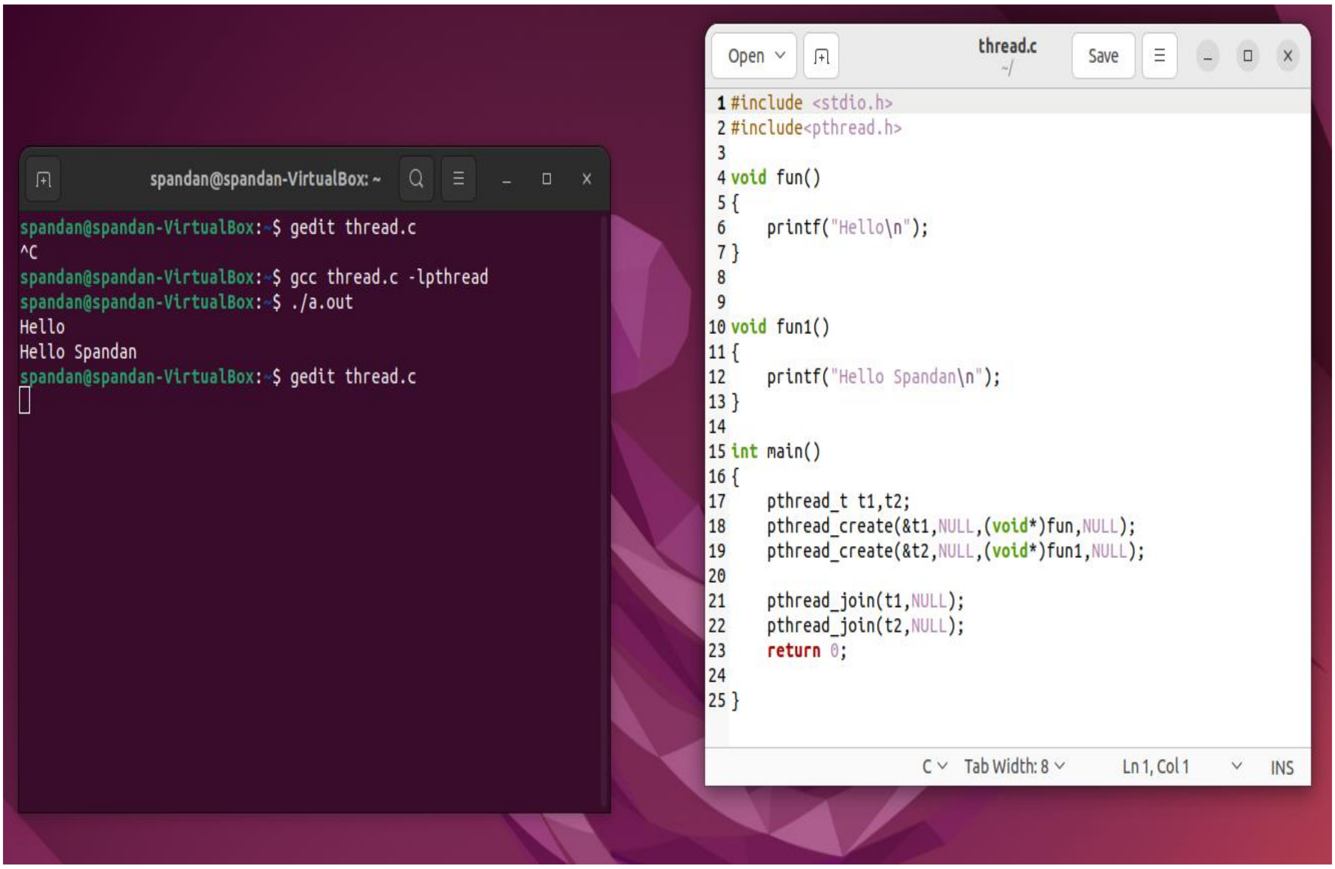
Name: Spandan Mukherjee

Registration Number: 21BCE1132

Slot: F2

Subject: Operating System (OS)

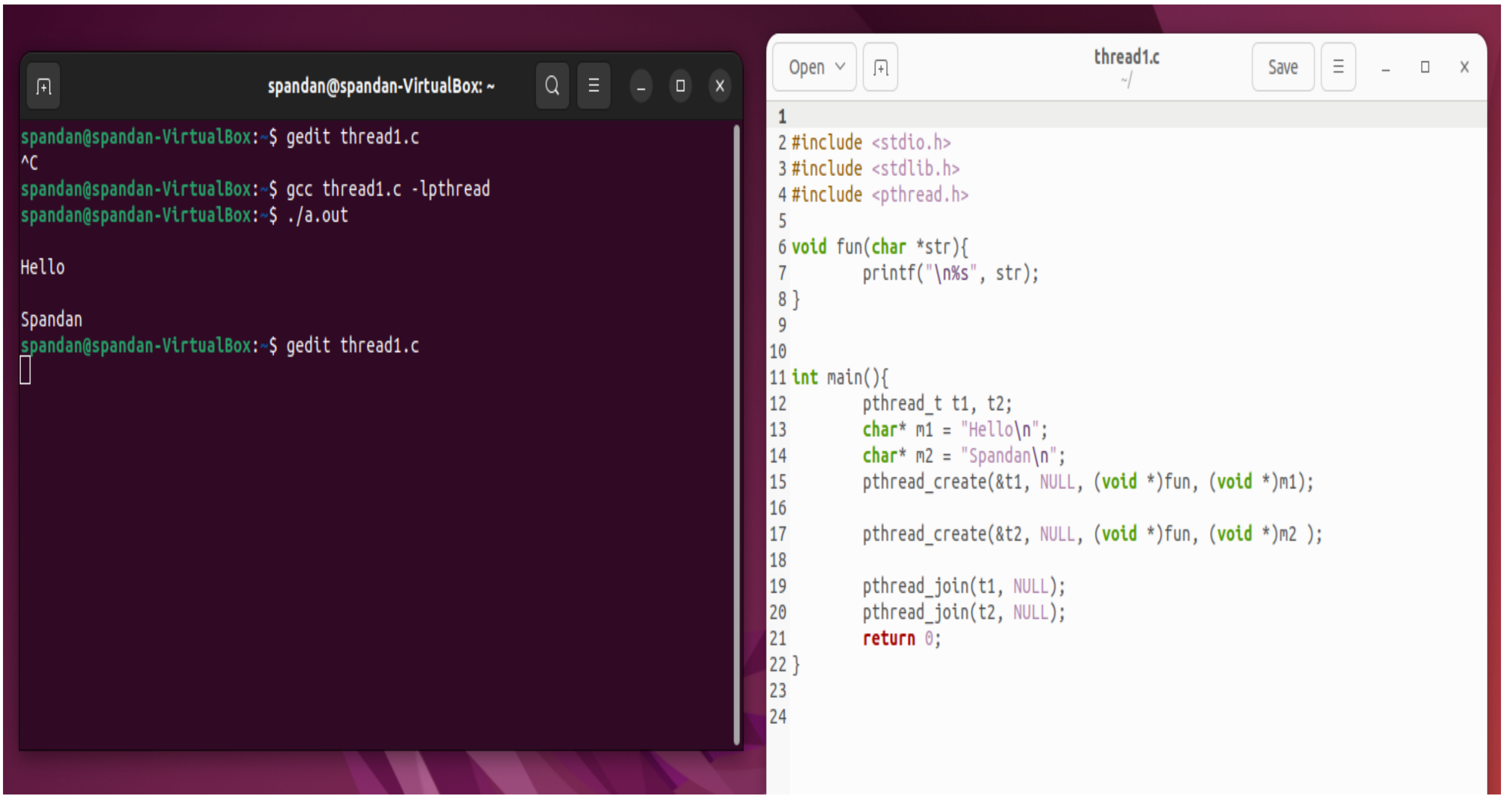
1) Design two threads to display two different messages using two different functions.



```
spandan@spandan-VirtualBox:~$ gedit thread.c
^C
spandan@spandan-VirtualBox:~$ gcc thread.c -lpthread
spandan@spandan-VirtualBox:~$ ./a.out
Hello
Hello Spandan
spandan@spandan-VirtualBox:~$ gedit thread.c
```

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

void fun()
{
    printf("Hello\n");
}


void fun1()
{
    printf("Hello Spandan\n");
}

int main()
{
    pthread_t t1,t2;
    pthread_create(&t1,NULL,(void*)fun,NULL);
    pthread_create(&t2,NULL,(void*)fun1,NULL);

    pthread_join(t1,NULL);
    pthread_join(t2,NULL);
    return 0;

}
```

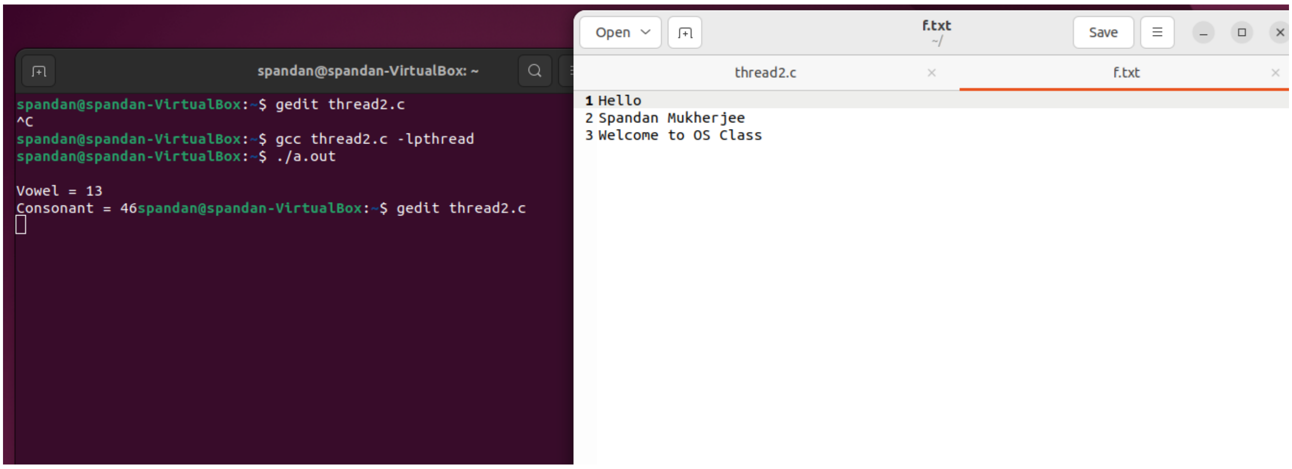
2) Design two threads to display two different messages using single function.



```
spandan@spandan-VirtualBox:~$ gedit thread1.c
^C
spandan@spandan-VirtualBox:~$ gcc thread1.c -lpthread
spandan@spandan-VirtualBox:~$ ./a.out

Hello

Spandan
spandan@spandan-VirtualBox:~$ gedit thread1.c
```

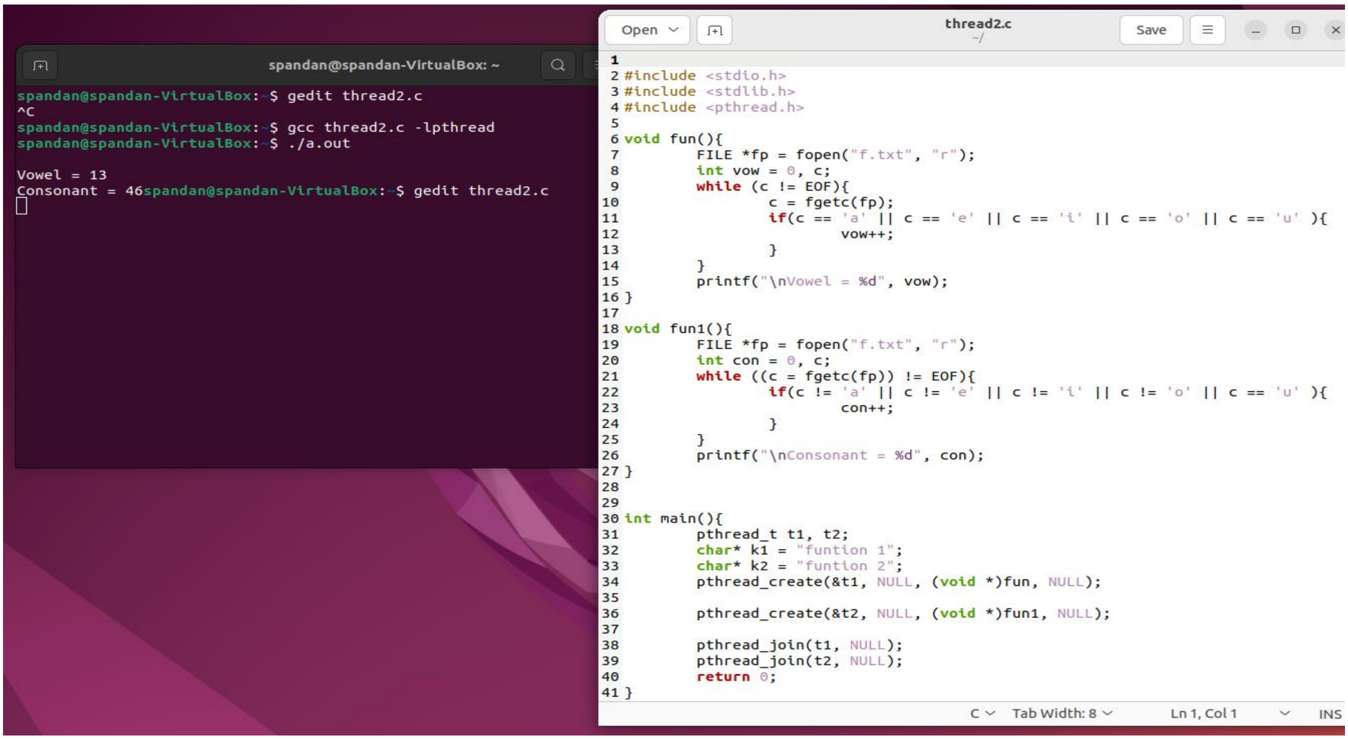
```c

#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>

void fun(char *str){
        printf("\n%s", str);
}


int main(){
        pthread_t t1, t2;
        char* m1 = "Hello\n";
        char* m2 = "Spandan\n";
        pthread_create(&t1, NULL, (void *)fun, (void *)m1);

        pthread_create(&t2, NULL, (void *)fun, (void *)m2 );

        pthread_join(t1, NULL);
        pthread_join(t2, NULL);
        return 0;
}


```

3) Design two threads to count the vowels and consonants either from "a.txt" file or a given string.



```
spandan@spandan-VirtualBox:~$ gedit thread2.c
^C
spandan@spandan-VirtualBox:~$ gcc thread2.c -lpthread
spandan@spandan-VirtualBox:~$ ./a.out

Vowel = 13
Consonant = 46spandan@spandan-VirtualBox:~$ gedit thread2.c
```

```
Hello
Spandan Mukherjee
Welcome to OS Class
```

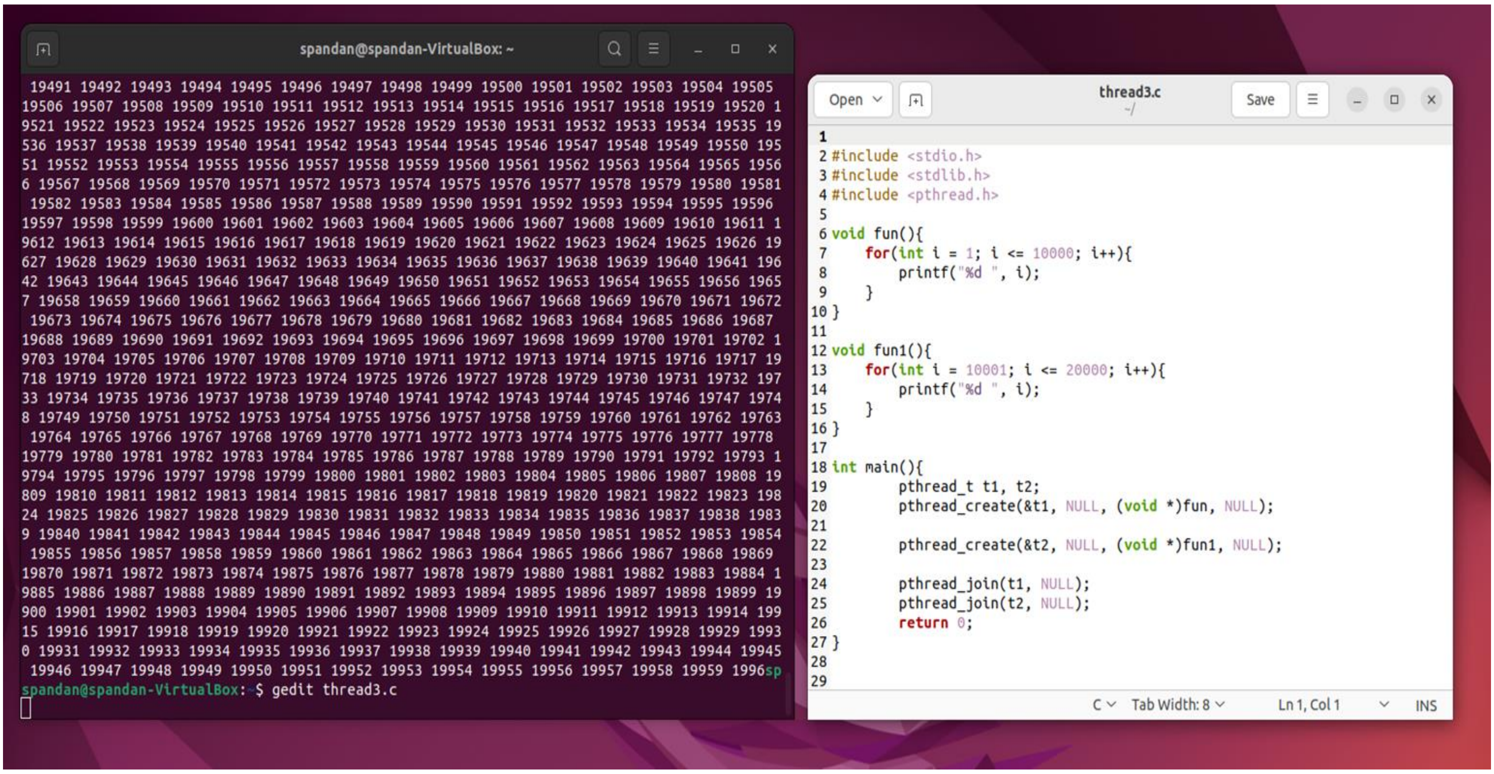


```
spandan@spandan-VirtualBox:~$ gedit thread2.c
^C
spandan@spandan-VirtualBox:~$ gcc thread2.c -lpthread
spandan@spandan-VirtualBox:~$ ./a.out

Vowel = 13
Consonant = 46spandan@spandan-VirtualBox:~$ gedit thread2.c
```

```

#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>

void fun(){
        FILE *fp = fopen("f.txt", "r");
        int vow = 0, c;
        while (c != EOF){
                c = fgetc(fp);
                if(c == 'a' || c == 'e' || c == 'i' || c == 'o' || c == 'u' ){
                        vow++;
                }
        }
        printf("\nVowel = %d", vow);
}

void fun1(){
        FILE *fp = fopen("f.txt", "r");
        int con = 0, c;
        while ((c = fgetc(fp)) != EOF){
                if(c != 'a' || c != 'e' || c != 'i' || c != 'o' || c == 'u' ){
                        con++;
                }
        }
        printf("\nConsonant = %d", con);
}


int main(){
        pthread_t t1, t2;
        char* k1 = "funtion 1";
        char* k2 = "funtion 2";
        pthread_create(&t1, NULL, (void *)fun, NULL);

        pthread_create(&t2, NULL, (void *)fun1, NULL);

        pthread_join(t1, NULL);
        pthread_join(t2, NULL);
        return 0;
}
```

4) Design two threads to display the numbers (i) from 1 to 10000 (ii) from 10001 to 20000 using (a) two different functions or (ii) single function. Discuss the nature of the output generated by the two threads in command prompt.

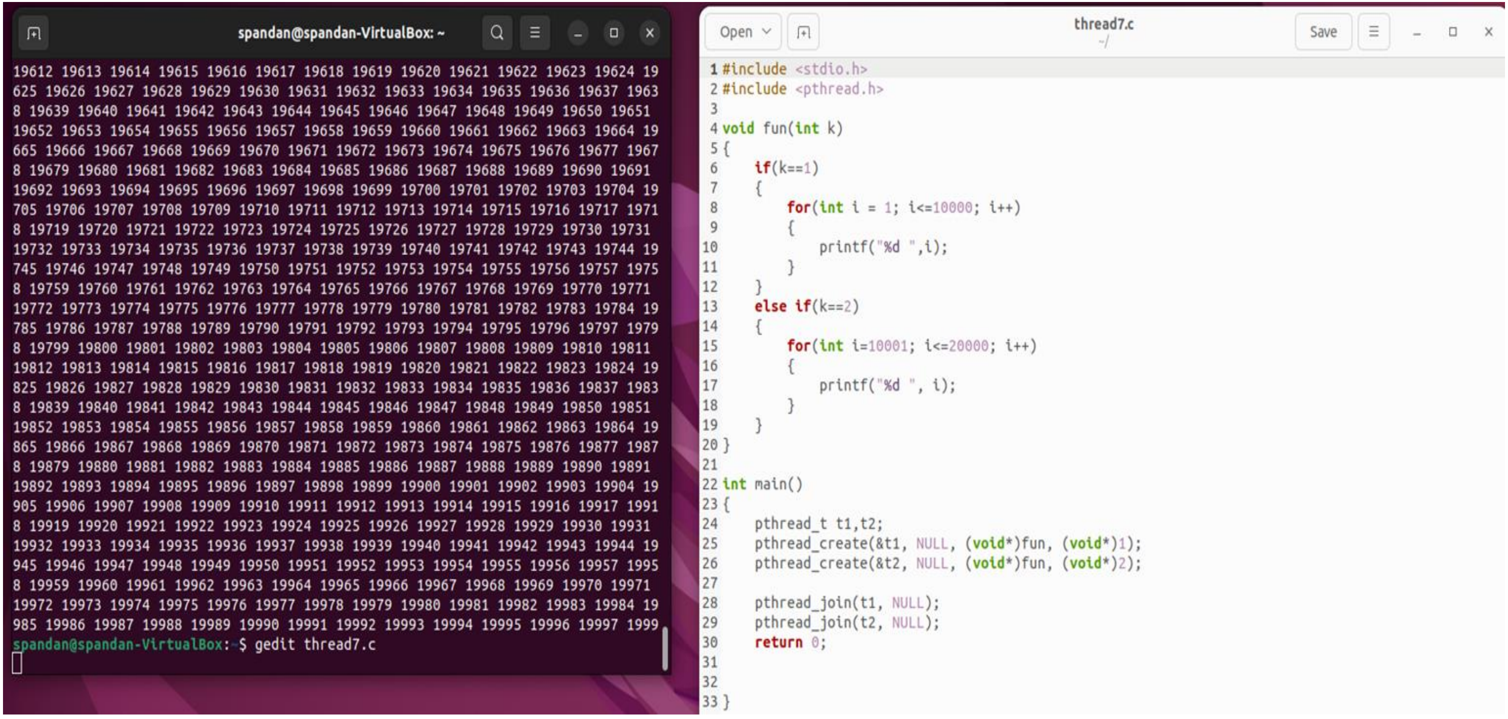
(i)



```
 19491 19492 19493 19494 19495 19496 19497 19498 19499 19500 19501 19502 19503 19504 19505
19506 19507 19508 19509 19510 19511 19512 19513 19514 19515 19516 19517 19518 19519 19520 1
9521 19522 19523 19524 19525 19526 19527 19528 19529 19530 19531 19532 19533 19534 19535 19
536 19537 19538 19539 19540 19541 19542 19543 19544 19545 19546 19547 19548 19549 19550 195
51 19552 19553 19554 19555 19556 19557 19558 19559 19560 19561 19562 19563 19564 19565 1956
6 19567 19568 19569 19570 19571 19572 19573 19574 19575 19576 19577 19578 19579 19580 19581
 19582 19583 19584 19585 19586 19587 19588 19589 19590 19591 19592 19593 19594 19595 19596
19597 19598 19599 19600 19601 19602 19603 19604 19605 19606 19607 19608 19609 19610 19611 1
9612 19613 19614 19615 19616 19617 19618 19619 19620 19621 19622 19623 19624 19625 19626 19
627 19628 19629 19630 19631 19632 19633 19634 19635 19636 19637 19638 19639 19640 19641 196
42 19643 19644 19645 19646 19647 19648 19649 19650 19651 19652 19653 19654 19655 19656 1965
7 19658 19659 19660 19661 19662 19663 19664 19665 19666 19667 19668 19669 19670 19671 19672
 19673 19674 19675 19676 19677 19678 19679 19680 19681 19682 19683 19684 19685 19686 19687
19688 19689 19690 19691 19692 19693 19694 19695 19696 19697 19698 19699 19700 19701 19702 1
9703 19704 19705 19706 19707 19708 19709 19710 19711 19712 19713 19714 19715 19716 19717 19
718 19719 19720 19721 19722 19723 19724 19725 19726 19727 19728 19729 19730 19731 19732 197
33 19734 19735 19736 19737 19738 19739 19740 19741 19742 19743 19744 19745 19746 19747 1974
8 19749 19750 19751 19752 19753 19754 19755 19756 19757 19758 19759 19760 19761 19762 19763
 19764 19765 19766 19767 19768 19769 19770 19771 19772 19773 19774 19775 19776 19777 19778
19779 19780 19781 19782 19783 19784 19785 19786 19787 19788 19789 19790 19791 19792 19793 1
9794 19795 19796 19797 19798 19799 19800 19801 19802 19803 19804 19805 19806 19807 19808 19
809 19810 19811 19812 19813 19814 19815 19816 19817 19818 19819 19820 19821 19822 19823 198
24 19825 19826 19827 19828 19829 19830 19831 19832 19833 19834 19835 19836 19837 19838 1983
9 19840 19841 19842 19843 19844 19845 19846 19847 19848 19849 19850 19851 19852 19853 19854
 19855 19856 19857 19858 19859 19860 19861 19862 19863 19864 19865 19866 19867 19868 19869
19870 19871 19872 19873 19874 19875 19876 19877 19878 19879 19880 19881 19882 19883 19884 1
9885 19886 19887 19888 19889 19890 19891 19892 19893 19894 19895 19896 19897 19898 19899 19
900 19901 19902 19903 19904 19905 19906 19907 19908 19909 19910 19911 19912 19913 19914 199
15 19916 19917 19918 19919 19920 19921 19922 19923 19924 19925 19926 19927 19928 19929 1993
0 19931 19932 19933 19934 19935 19936 19937 19938 19939 19940 19941 19942 19943 19944 19945
 19946 19947 19948 19949 19950 19951 19952 19953 19954 19955 19956 19957 19958 19959 1996sp
spandan@spandan-VirtualBox:~$ gedit thread3.c
```

```c

#include <stdio.h>
#include <stdlib.h>
#include <pthread.h>

void fun(){
    for(int i = 1; i <= 10000; i++){
        printf("%d ", i);
    }
}

void fun1(){
    for(int i = 10001; i <= 20000; i++){
        printf("%d ", i);
    }
}

int main(){
        pthread_t t1, t2;
        pthread_create(&t1, NULL, (void *)fun, NULL);

        pthread_create(&t2, NULL, (void *)fun1, NULL);

        pthread_join(t1, NULL);
        pthread_join(t2, NULL);
        return 0;
}


```

(ii)

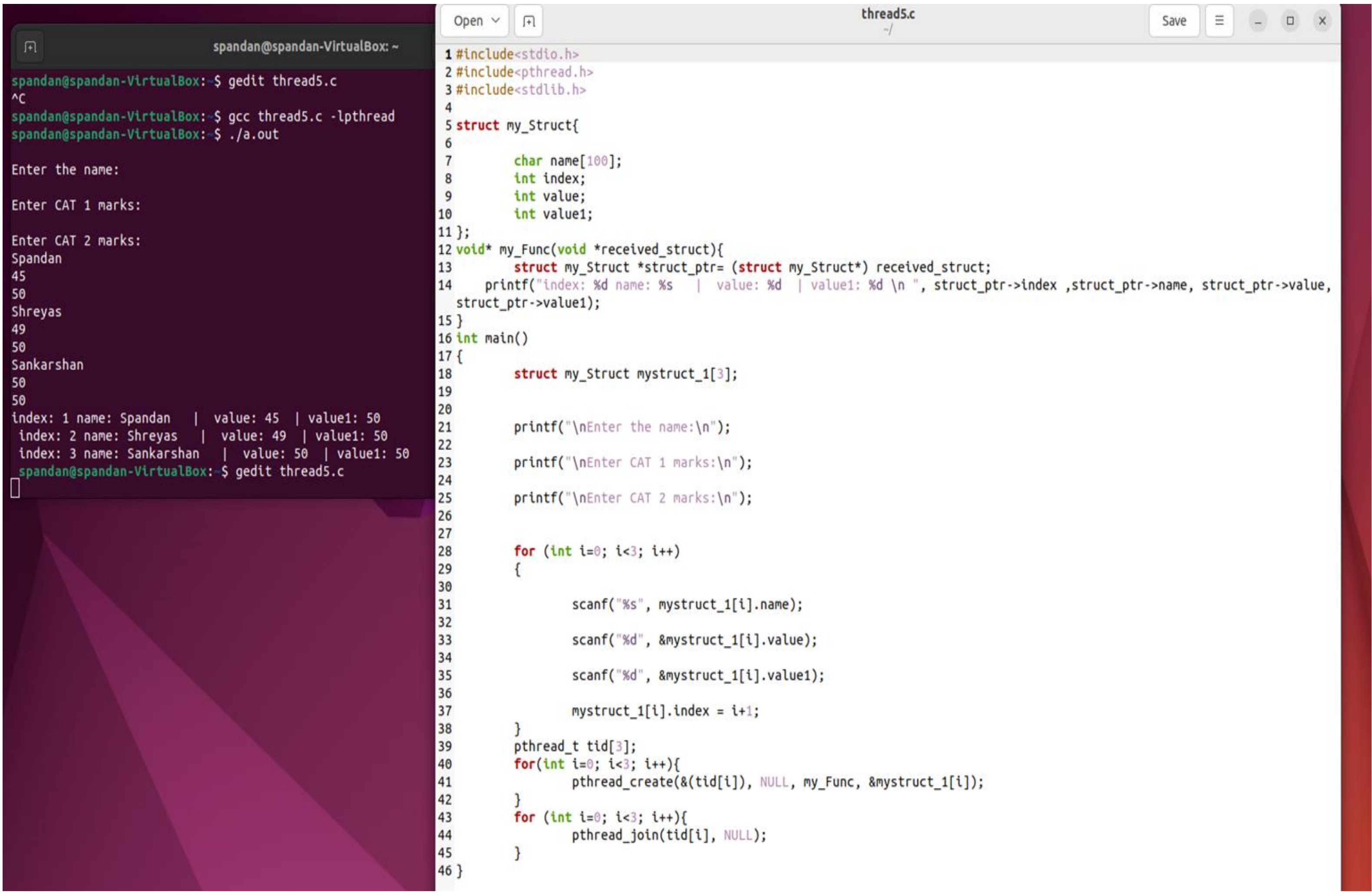


```
19612 19613 19614 19615 19616 19617 19618 19619 19620 19621 19622 19623 19624 19
625 19626 19627 19628 19629 19630 19631 19632 19633 19634 19635 19636 19637 1963
8 19639 19640 19641 19642 19643 19644 19645 19646 19647 19648 19649 19650 19651
19652 19653 19654 19655 19656 19657 19658 19659 19660 19661 19662 19663 19664 19
665 19666 19667 19668 19669 19670 19671 19672 19673 19674 19675 19676 19677 1967
8 19679 19680 19681 19682 19683 19684 19685 19686 19687 19688 19689 19690 19691
19692 19693 19694 19695 19696 19697 19698 19699 19700 19701 19702 19703 19704 19
705 19706 19707 19708 19709 19710 19711 19712 19713 19714 19715 19716 19717 1971
8 19719 19720 19721 19722 19723 19724 19725 19726 19727 19728 19729 19730 19731
19732 19733 19734 19735 19736 19737 19738 19739 19740 19741 19742 19743 19744 19
745 19746 19747 19748 19749 19750 19751 19752 19753 19754 19755 19756 19757 1975
8 19759 19760 19761 19762 19763 19764 19765 19766 19767 19768 19769 19770 19771
19772 19773 19774 19775 19776 19777 19778 19779 19780 19781 19782 19783 19784 19
785 19786 19787 19788 19789 19790 19791 19792 19793 19794 19795 19796 19797 1979
8 19799 19800 19801 19802 19803 19804 19805 19806 19807 19808 19809 19810 19811
19812 19813 19814 19815 19816 19817 19818 19819 19820 19821 19822 19823 19824 19
825 19826 19827 19828 19829 19830 19831 19832 19833 19834 19835 19836 19837 1983
8 19839 19840 19841 19842 19843 19844 19845 19846 19847 19848 19849 19850 19851
19852 19853 19854 19855 19856 19857 19858 19859 19860 19861 19862 19863 19864 19
865 19866 19867 19868 19869 19870 19871 19872 19873 19874 19875 19876 19877 1987
8 19879 19880 19881 19882 19883 19884 19885 19886 19887 19888 19889 19890 19891
19892 19893 19894 19895 19896 19897 19898 19899 19900 19901 19902 19903 19904 19
905 19906 19907 19908 19909 19910 19911 19912 19913 19914 19915 19916 19917 1991
8 19919 19920 19921 19922 19923 19924 19925 19926 19927 19928 19929 19930 19931
19932 19933 19934 19935 19936 19937 19938 19939 19940 19941 19942 19943 19944 19
945 19946 19947 19948 19949 19950 19951 19952 19953 19954 19955 19956 19957 1995
8 19959 19960 19961 19962 19963 19964 19965 19966 19967 19968 19969 19970 19971
19972 19973 19974 19975 19976 19977 19978 19979 19980 19981 19982 19983 19984 19
985 19986 19987 19988 19989 19990 19991 19992 19993 19994 19995 19996 19997 1999
spandan@spandan-VirtualBox:~$ gedit thread7.c
```

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

void fun(int k)
{
    if(k==1)
    {
        for(int i = 1; i<=10000; i++)
        {
            printf("%d ",i);
        }
    }
    else if(k==2)
    {
        for(int i=10001; i<=20000; i++)
        {
            printf("%d ", i);
        }
    }
}

int main()
{
    pthread_t t1,t2;
    pthread_create(&t1, NULL, (void*)fun, (void*)1);
    pthread_create(&t2, NULL, (void*)fun, (void*)2);

    pthread_join(t1, NULL);
    pthread_join(t2, NULL);
    return 0;


}
```

5) Design two threads to display the student name and CAT1 mark by first thread and student name and CAT2 mark by second thread using **struct data type for a single student (tutorial: https://www.**n this website the header files have not been included - you include it and refer this web site for passing structure and modify as per the question.

Implement the same program for array of students of class strength 70 and analyze the output generated by the program.



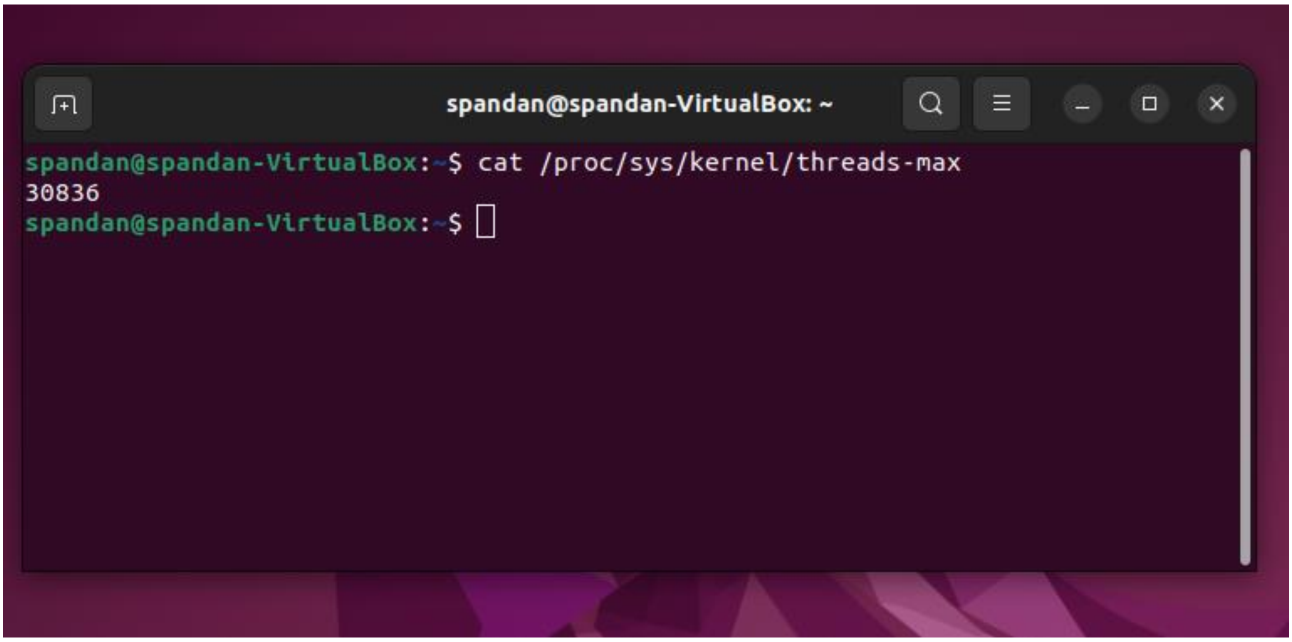
```
spandan@spandan-VirtualBox:~$ gedit thread5.c
^C
spandan@spandan-VirtualBox:~$ gcc thread5.c -lpthread
spandan@spandan-VirtualBox:~$ ./a.out

Enter the name:

Enter CAT 1 marks:

Enter CAT 2 marks:
Spandan
45
50
Shreyas
49
50
Sankarshan
50
50
index: 1 name: Spandan   |  value: 45  | value1: 50
 index: 2 name: Shreyas   |  value: 49  | value1: 50
 index: 3 name: Sankarshan   |  value: 50  | value1: 50
 spandan@spandan-VirtualBox:~$ gedit thread5.c
```

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

struct my_Struct{

        char name[100];
        int index;
        int value;
        int value1;
};
void* my_Func(void *received_struct){
        struct my_Struct *struct_ptr= (struct my_Struct*) received_struct;
    printf("index: %d name: %s   |  value: %d   | value1: %d \n ", struct_ptr->index ,struct_ptr->name, struct_ptr->value,
struct_ptr->value1);
}
int main()
{
        struct my_Struct mystruct_1[3];


        printf("\nEnter the name:\n");

        printf("\nEnter CAT 1 marks:\n");

        printf("\nEnter CAT 2 marks:\n");


        for (int i=0; i<3; i++)
        {

                scanf("%s", mystruct_1[i].name);

                scanf("%d", &mystruct_1[i].value);

                scanf("%d", &mystruct_1[i].value1);

                mystruct_1[i].index = i+1;
        }
        pthread_t tid[3];
        for(int i=0; i<3; i++){
                pthread_create(&(tid[i]), NULL, my_Func, &mystruct_1[i]);
        }
        for (int i=0; i<3; i++){
                pthread_join(tid[i], NULL);
        }
}
```

The above problem can easily handled with the help of 1 thread which will recall the values of the datatype.

The above code limits of the loop can be changed to 70 as we can’t enter that much data.

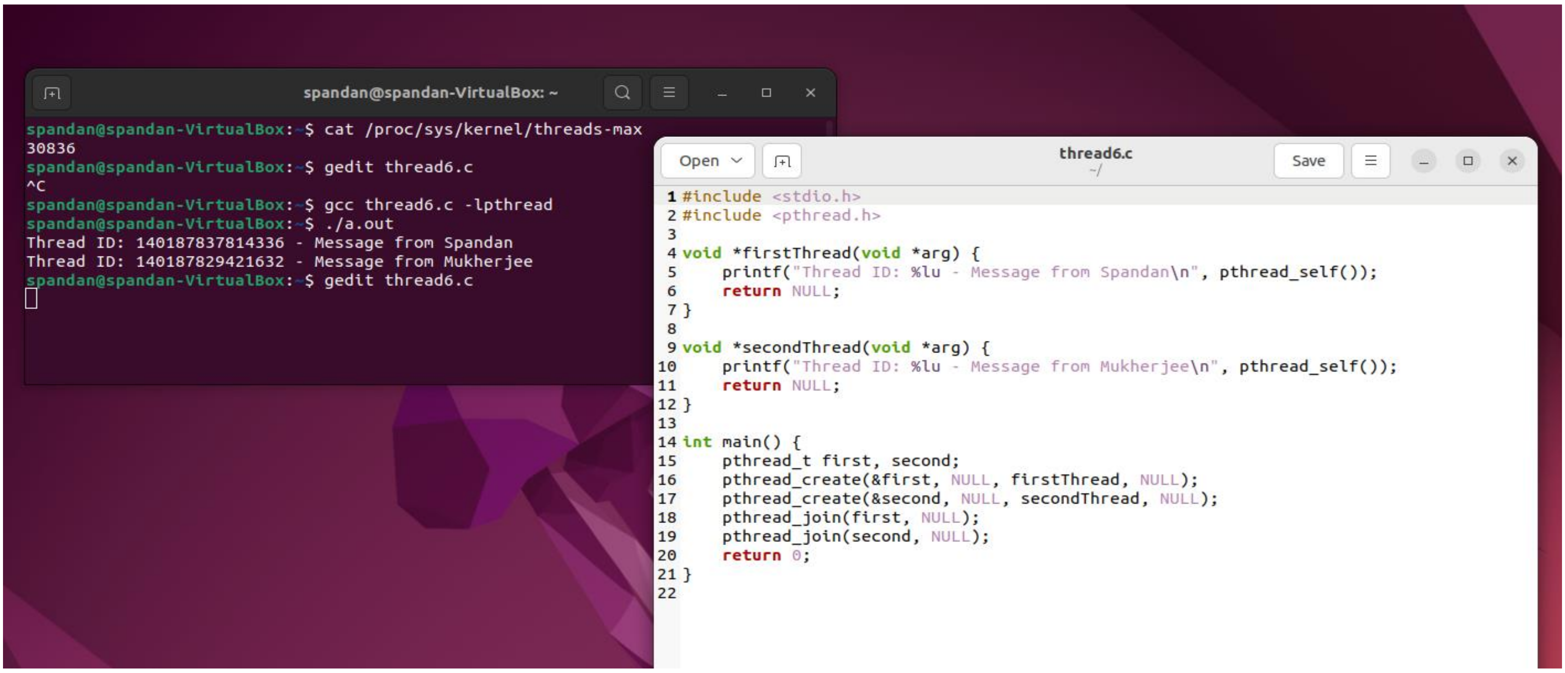
6)Find out the maximum number of threads your system can generate?



```
spandan@spandan-VirtualBox:~$ cat /proc/sys/kernel/threads-max
30836
spandan@spandan-VirtualBox:~$ 
```

7) Create two threads and display the two messages along with the **corresponding thread_id. (pthread_self() function returns thread id) (reference: https://www.)**

https://riptutorial.com/ - returning value from the threads may be explained later.
https://www.cs.cmu.edu/afs/cs/



```
spandan@spandan-VirtualBox:~$ cat /proc/sys/kernel/threads-max
30836
spandan@spandan-VirtualBox:~$ gedit thread6.c
^C
spandan@spandan-VirtualBox:~$ gcc thread6.c -lpthread
spandan@spandan-VirtualBox:~$ ./a.out
Thread ID: 140187837814336 - Message from Spandan
Thread ID: 140187829421632 - Message from Mukherjee
spandan@spandan-VirtualBox:~$ gedit thread6.c
```

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

void *firstThread(void *arg) {
    printf("Thread ID: %lu - Message from Spandan\n", pthread_self());
    return NULL;
}

void *secondThread(void *arg) {
    printf("Thread ID: %lu - Message from Mukherjee\n", pthread_self());
    return NULL;
}

int main() {
    pthread_t first, second;
    pthread_create(&first, NULL, firstThread, NULL);
    pthread_create(&second, NULL, secondThread, NULL);
    pthread_join(first, NULL);
    pthread_join(second, NULL);
    return 0;
}

```