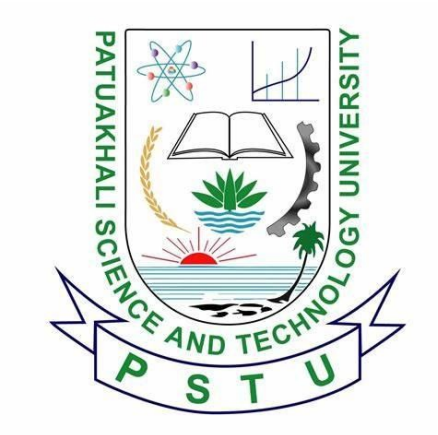


PATUAKHALI SCIENCE AND TECHNOLOGY UNIVERSITY

COURSE CODE CIT-112

SUBMITTED TO:

Md. Mahbubur Rahman

Department of Computer Science and Information Technology

Faculty of Computer Science and Engineering

SUBMITTED BY:

Md. Sharafat Karim

ID: **2102024**,

Registration No: **10151**

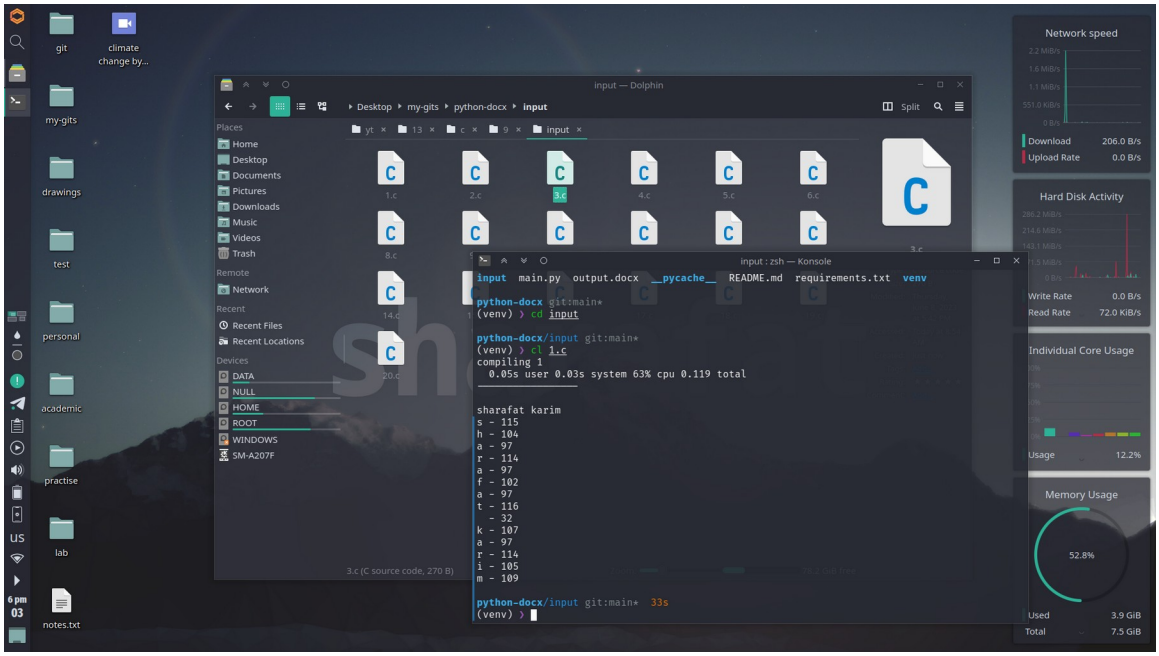
Faculty of Computer Science and Engineering

Assignment: **07**

Table of Contents

1 Write a program, which reads your name from the keyboard and outputs a list of ASCII codes, which represent your name.

```
#include<stdio.h>

int main()
{
   char name[32];
   scanf("%[^\n]", name);

   for (int i=0; name[i] != '\0'; i++)
   {
      printf("%c - %d\n", name[i], name[i]);
   }
}
```



2 Write a program to do string processing

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

int main()
{
   char answer[64];
   int attemp = 3;
   while (attemp--)
   {
      printf("Who is the inventor of C? \n");
      scanf(" %[^\n]", answer);
      if (strcmp(answer, "Dennis Ritchie") == 0)
      {
         printf("Good\n");
         break;
      }
      else
      {
         printf("try again\nCorrect answer: Dennis Ritchie\n");
      }
   }
}
```


input : zsh — Konsole
sharafat karim
s - 115
h - 104
a - 97
r - 114
a - 97
f - 102
a - 97
t - 116
- 32
k - 107
a - 97
r - 114
i - 105
m - 109
python-docx/input git:main* 33s
(venv) ❯ cl 2.c
compiling 2
0.06s user 0.03s system 73% cpu 0.114 total
Who is the inventor of C?
Dennis Ritchie
Good
python-docx/input git:main* 12s
(venv) ❯

3 Write a program to extract a portion of a character string and print the extracted string. Assume that m characters are extracted, starting with the nth character.

```
#include<stdio.h>

#define init 2
#define end 6

int main()
{
   char string[32], modified_string[32];
   scanf("%[^\n]", string);

   for (int i= init-1; i < end; i++)
   {
      modified_string[i-init+1] = string[i];
   }

   printf("%s\n", modified_string);
}
```


input : 3 — Konsole
t - 116
- 32
k - 107
a - 97
r - 114
i - 105
m - 109
python-docx/input git:main* 33s
(venv) ❯ cl 2.c
compiling 2
0.06s user 0.03s system 73% cpu 0.114 total
Who is the inventor of C?
Dennis Ritchie
Good
python-docx/input git:main* 12s
(venv) ❯ cl 3.c
compiling 3
0.05s user 0.03s system 66% cpu 0.111 total
Hello world, how is it going?
ello
python-docx/input git:main* 15s
(venv) ❯

4 Write a program which will read a text and count all occurrences of a particular word.

```
#include<stdio.h>

#define occurance 'a'

int main()
{
   char string[32];
   scanf("%[^\n]", string);
   int i=0, count = 0;
   while (string[i] != '\0')
   {
      if (string[i++] == occurance)
      {
         count++;
      }
   }
   printf("Total occurance of %c = %d\n", occurance, count);
}
```


input : 4 — Konsole
(venv) ❯ cl 2.c
compiling 2
0.06s user 0.03s system 73% cpu 0.114 total
Who is the inventor of C?
Dennis Ritchie
Good
python-docx/input git:main* 12s
(venv) ❯ cl 3.c
compiling 3
0.05s user 0.03s system 66% cpu 0.111 total
Hello world, how is it going?
ello
python-docx/input git:main* 15s
(venv) ❯ cl 4.c
compiling 4
0.04s user 0.02s system 93% cpu 0.068 total
hi there, I'm gonna find all occurances
Total occurance of a = 3
python-docx/input git:main* 10s
(venv) ❯

5 Write a program which will read a string and rewrite it in the alphabetical order. For example, the word STRING should be written as GINRST.

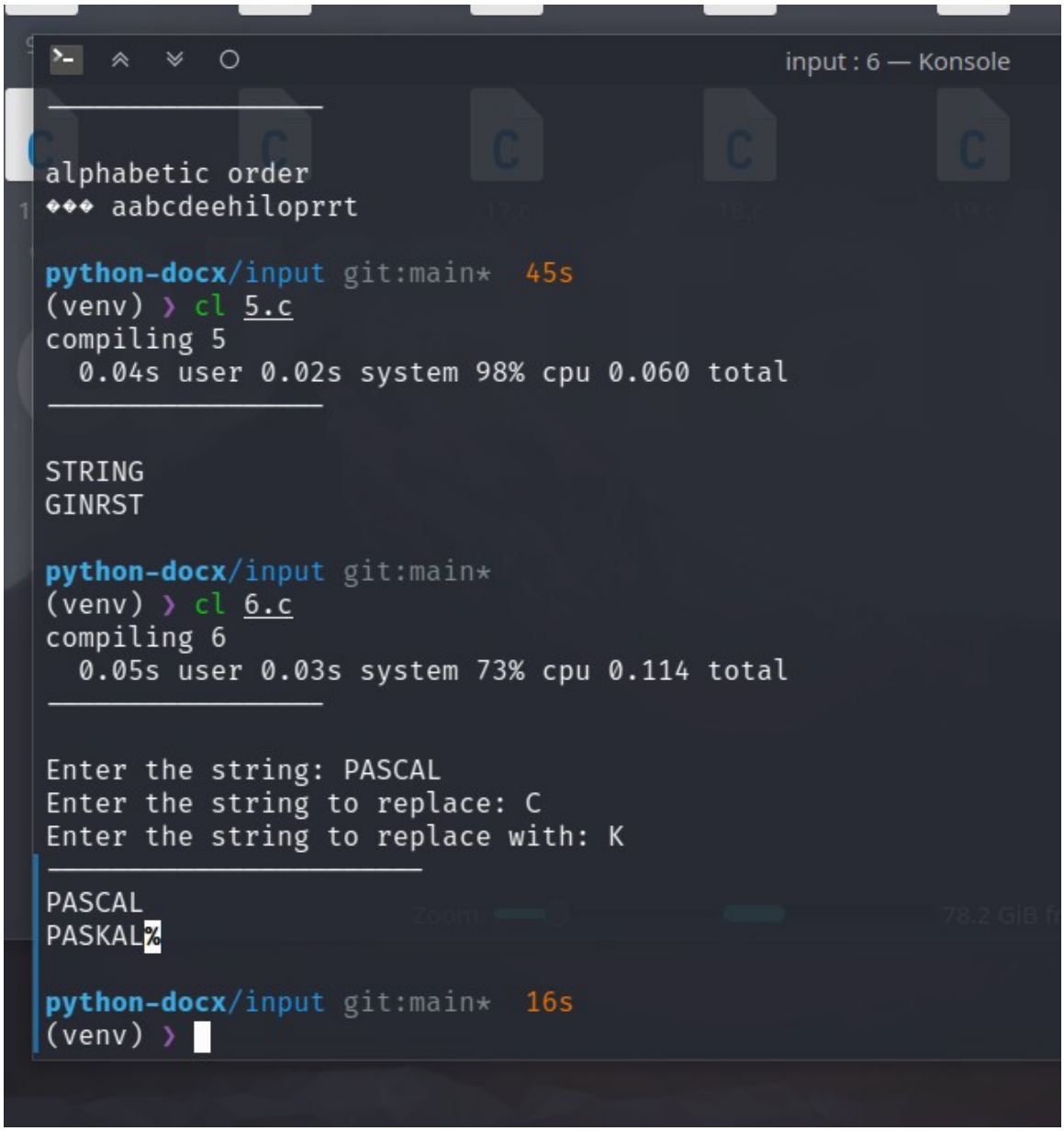
```
#include<stdio.h>
#include<string.h>

int main()
{
   char name[32];
   scanf("%[^\n]", name);

   int size = strlen(name);
   int i, j;
   for (i=0; i < size; i++)
   {
      for (j=0; j < size-i-1; j++)
      {
         if (name[j] > name[j+1])
         {
            char temp = name[j];
            name[j] = name[j+1];
            name[j+1] = temp;
         }
      }
   }
   printf("%s\n", name);
}
```


input : zsh — Konsole
python-docx/input git:main* 15s
(venv) ❯ cl 4.c
compiling 4
0.04s user 0.02s system 93% cpu 0.068 total
hi there, I'm gonna find all occurances
Total occurance of a = 3
python-docx/input git:main* 10s
(venv) ❯ cl 5.c
compiling 5
0.05s user 0.03s system 75% cpu 0.105 total
alphabetic order
◆◆◆ aabcdeehiloprrt
python-docx/input git:main* 45s
(venv) ❯ cl 5.c
compiling 5
0.04s user 0.02s system 98% cpu 0.060 total
STRING
GINRST
python-docx/input git:main*
(venv) ❯

6 Write a program to replace a particular word by another word in a given string. For example, the word “PASCAL” should be replaced by “C” in the text “It is good to program in PASCAL language.”

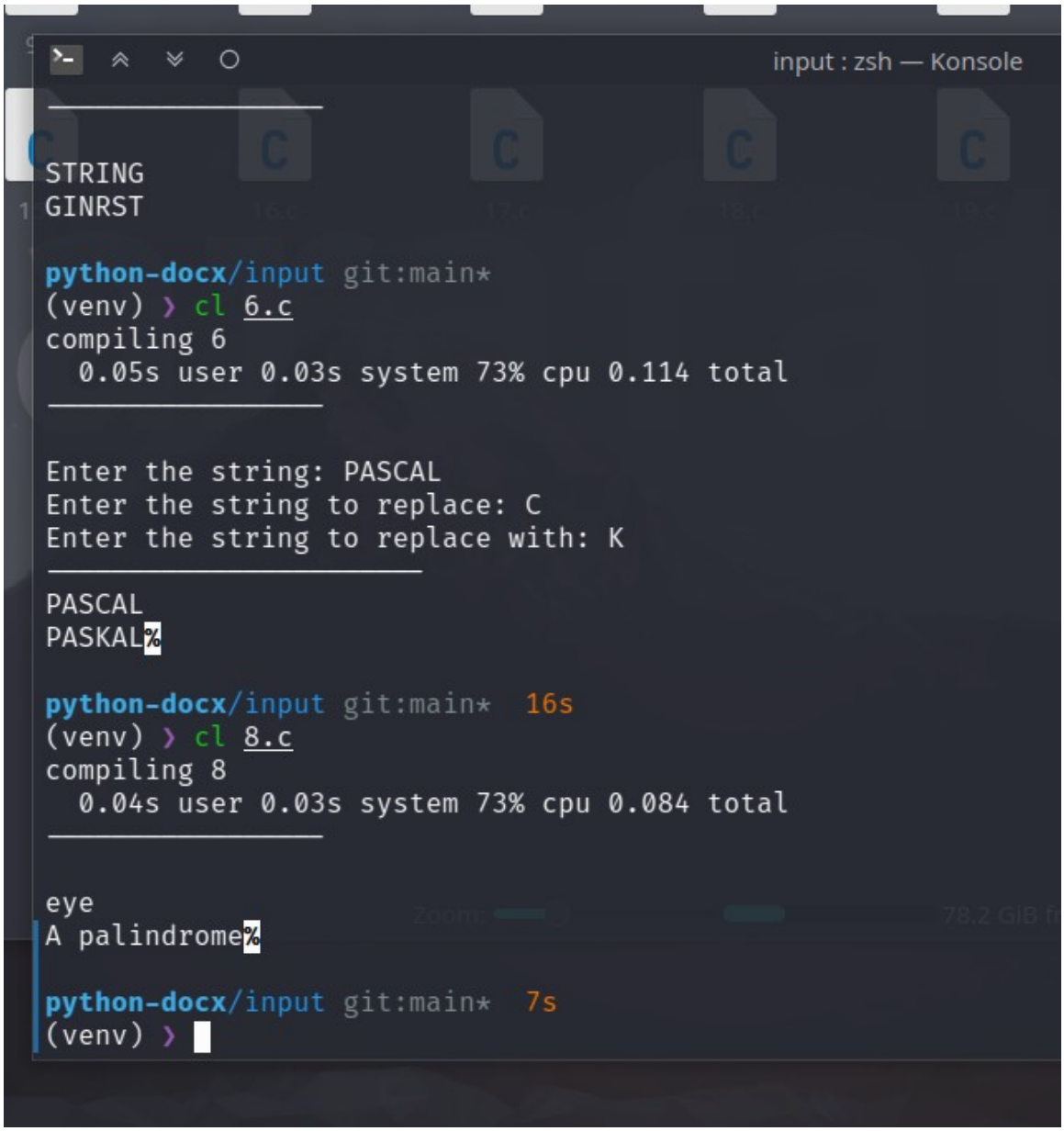
```
#include<stdio.h>
#include<string.h>

int main()
{
   char str[64], modified_str[64];
   printf("Enter the string: ");
   scanf("%[^\n]", str);
   char replace_from[16];
   printf("Enter the string to replace: ");
   scanf("%s", replace_from);
   char replace_with[16];
   printf("Enter the string to replace with: ");
   scanf("%s", replace_with);
   int size = strlen(str);

   printf("----------------------\n");

   int i, temp;
   for (i=0, temp=0; i<size; i++)
   {
      if (str[i] == replace_from[0])
      {
         for (int j=0; j<strlen(replace_from);j++)
         {
            if (str[i-j] != replace_from[j])
               break;
            else
            {
               for (int k=0; k<strlen(replace_with); k++)
               {
```

```
                modified_str[temp++] = replace_with[k];
                i += strlen(replace_from);
            }
        }
      }
    }
    modified_str[temp++] = str[i];
  }
  modified_str[temp] = '\0';

  printf("%s\n", str);
  printf("%s", modified_str);
}
```


input : 6 — Konsole
alphabetic order
◆◆◆ aabcdeehiloprrt
python-docx/input git:main* 45s
(venv) ❯ cl 5.c
compiling 5
0.04s user 0.02s system 98% cpu 0.060 total
STRING
GINRST
python-docx/input git:main*
(venv) ❯ cl 6.c
compiling 6
0.05s user 0.03s system 73% cpu 0.114 total
Enter the string: PASCAL
Enter the string to replace: C
Enter the string to replace with: K
PASCAL
PASKAL%
python-docx/input git:main* 16s
(venv) ❯

8 Write a program that reads a string from the keyboard and determines whether the string is a palindrome or not. (A string is a palindrome if it can be read from left and right with the same meaning. For example, Madam and Anna are palindrome strings. Ignore capitalization).

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

int main()
{
   char string[64];
   scanf("%[^\n]", string);
   int size = strlen(string);

   for (int i=0, j=size-1; i < size/2; i++, j--)
   {
      if (string[i] != string[j])
      {
         printf("Not a palindrome");
         return 0;
      }
   }
   printf("A palindrome");
   return 0;
}
```

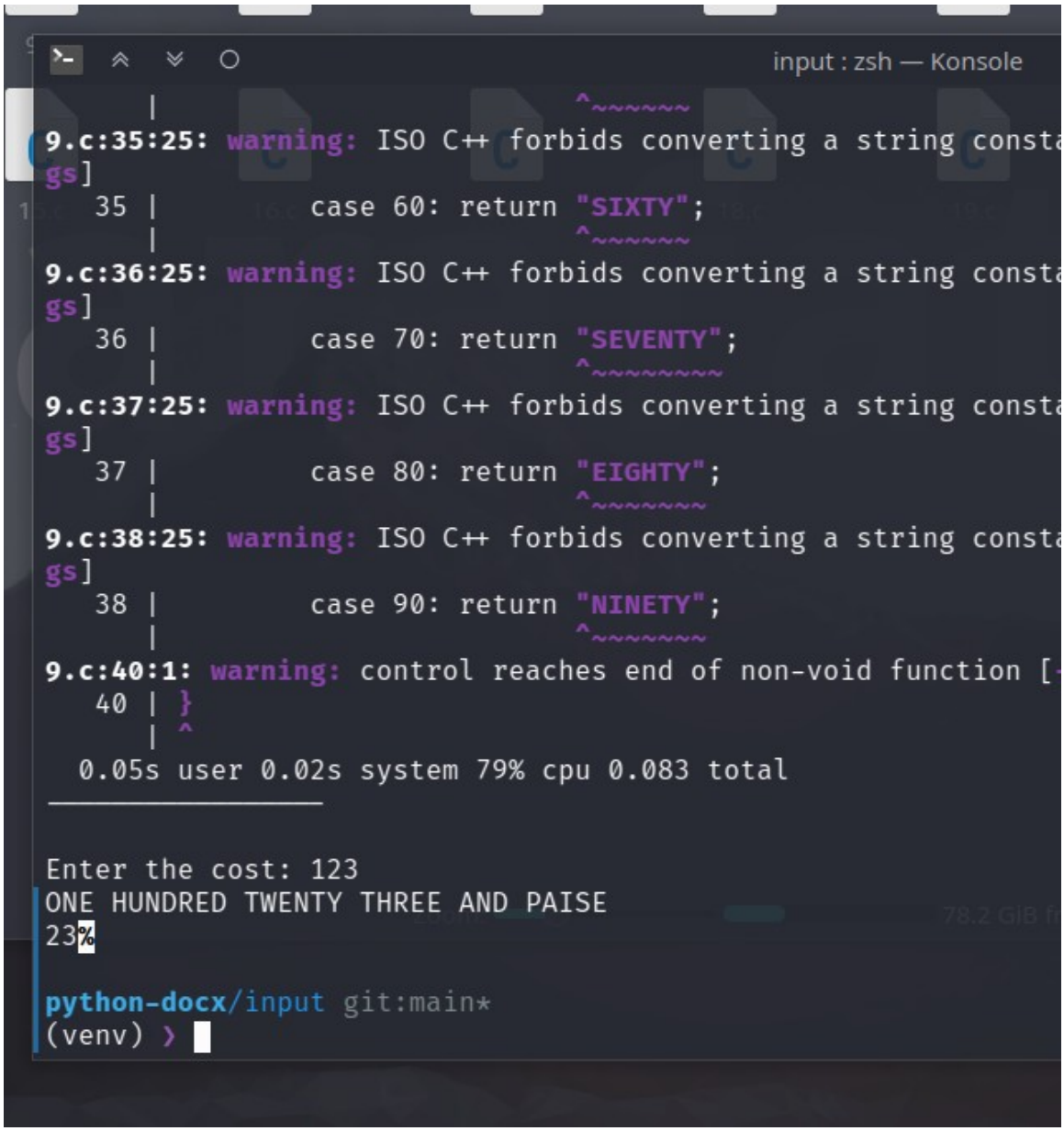

input : zsh — Konsole
STRING
GINRST
python-docx/input git:main*
(venv) ❯ cl 6.c
compiling 6
0.05s user 0.03s system 73% cpu 0.114 total
Enter the string: PASCAL
Enter the string to replace: C
Enter the string to replace with: K
PASCAL
PASKAL%
python-docx/input git:main* 16s
(venv) ❯ cl 8.c
compiling 8
0.04s user 0.03s system 73% cpu 0.084 total
eye
A palindrome%
python-docx/input git:main* 7s
(venv) ❯

9 Write program that reads the cost of an item in the form RRRR.PP (Where RRRR denotes Rupees and PP denotes Paise) and converts the value to a string of words that expresses the numeric value in words. For example, if we input 125.75, the output should be “ONE HUNDRED TWENTY FIVE AND PAISE SEVENTY FIVE”.

```
// Write program that reads the cost of an item in the form RRRR.PP (Where RRRR denotes Rupees and PP denotes Paise) and converts the value to a string of words that expresses the numeric value in words.
// For example, if we input 125.75, the output should be “ONE HUNDRED TWENTY FIVE AND PAISE SEVENTY FIVE”.

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

char* get_first_nine_number(int num)
{
   switch(num)
   {
      case 0: return "ZERO";
      case 1: return "ONE";
      case 2: return "TWO";
      case 3: return "THREE";
      case 4: return "FOUR";
      case 5: return "FIVE";
      case 6: return "SIX";
      case 7: return "SEVEN";
      case 8: return "EIGHT";
      case 9: return "NINE";
      case 10: return "TEN";
      case 11: return "ELEVEN";
      case 12: return "TWELVE";
      case 13: return "THIRTEEN";
      case 14: return "FOURTEEN";
      case 15: return "FIFTEEN";
```

```
        case 16: return "SIXTEEN";
        case 17: return "SEVENTEEN";
        case 18: return "EIGHTEEN";
        case 19: return "NINETEEN";
        case 20: return "TWENTY";
        case 30: return "THIRTY";
        case 40: return "FOURTY";
        case 50: return "FIFTY";
        case 60: return "SIXTY";
        case 70: return "SEVENTY";
        case 80: return "EIGHTY";
        case 90: return "NINETY";
    }
}


char* get_number(int num)
{
    char string[50];
    if (num <= 20 )
        return get_first_nine_number(num);
    else
    {
        strcpy(string, "");
        strcat(string, get_first_nine_number((num/10)*10));
        strcat(string, " ");
        strcat(string, get_first_nine_number(num%10));
        char *string2 = string;
        return(string2);
    }
}

int main()
{
    float cost;
    printf("Enter the cost: ");
    scanf("%f", &cost);
```

```
    int rupees = (int)cost;
    int paise = (int)((cost - rupees) * 100);


    char rupees_string[100];
    strcpy(rupees_string, "");


    if ( rupees >= 1000 )
    {
       int thousands = rupees / 1000;
       strcat(strcat(rupees_string, get_number(thousands)), " THOUSAND ");
       rupees = rupees % 1000;
    }
    if ( rupees >= 100 )
    {
       int hundreds = rupees / 100;
       strcat(strcat(rupees_string, get_number(hundreds)), " HUNDRED ");
       rupees = rupees % 100;
    }
    if ( rupees >= 1 )
    {
       strcat(strcat(rupees_string, get_number(rupees)), " AND PAISE ");
    }
    if ( paise >= 1 )
    {
       strcat(rupees_string, get_number(paise));
    }

    printf("%s", rupees_string);
    printf("\n%d", rupees);
}
```

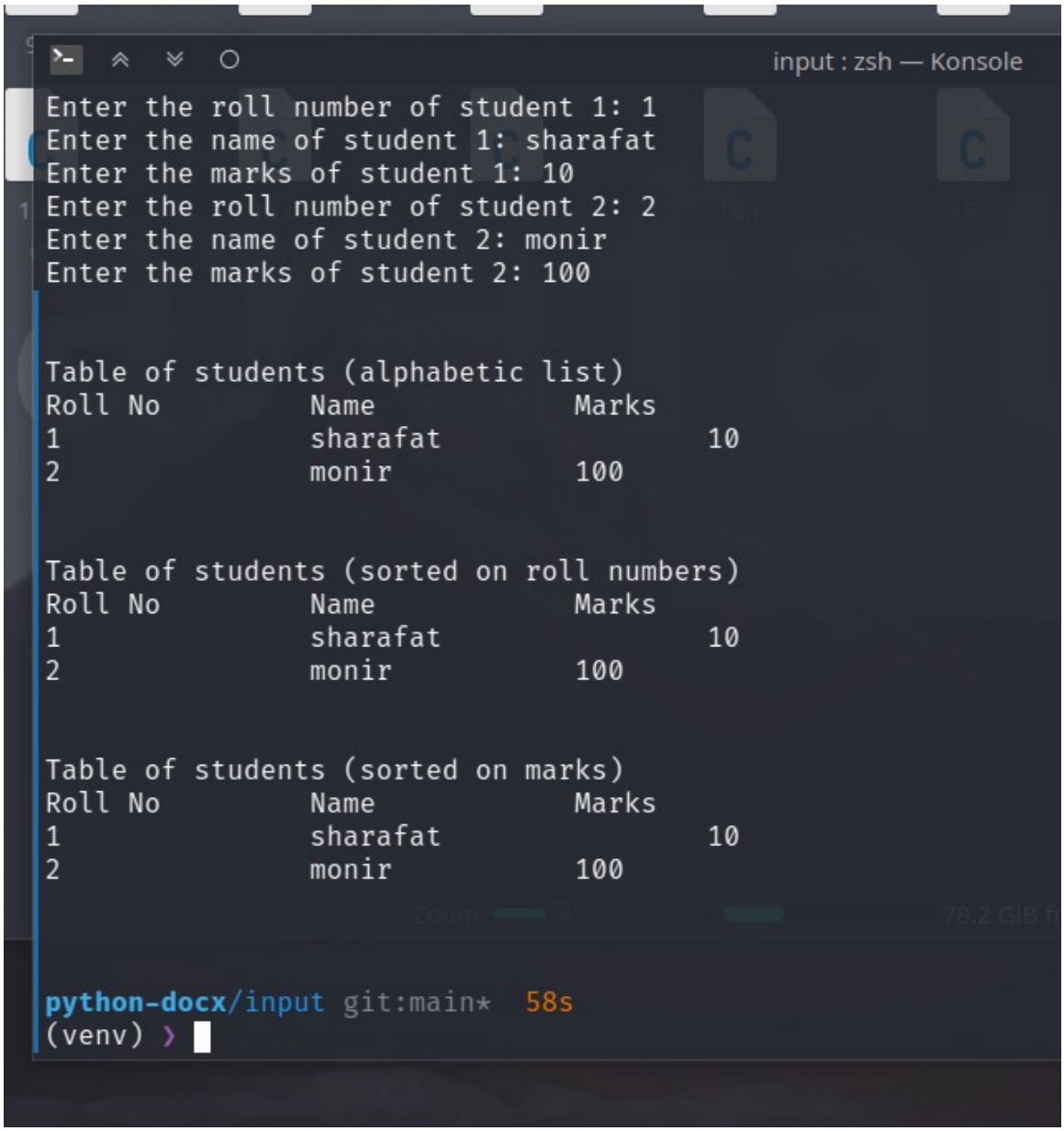

input : zsh — Konsole
9.c:35:25: warning: ISO C++ forbids converting a string consta
gs]
35 | case 60: return "SIXTY";
9.c:36:25: warning: ISO C++ forbids converting a string consta
gs]
36 | case 70: return "SEVENTY";
9.c:37:25: warning: ISO C++ forbids converting a string consta
gs]
37 | case 80: return "EIGHTY";
9.c:38:25: warning: ISO C++ forbids converting a string consta
gs]
38 | case 90: return "NINETY";
9.c:40:1: warning: control reaches end of non-void function [
40 | }
0.05s user 0.02s system 79% cpu 0.083 total
Enter the cost: 123
ONE HUNDRED TWENTY THREE AND PAISE
23%
python-docx/input git:main*
(venv) ❯

10 Develop a program that will read and store the details of a list of students in the format

```
#include<stdio.h>

struct student
{
   int roll_no;
   char name[50];
   int marks;
};

void sorting_array_with_index(int main[], int sorted_index[])
{
   int n = sizeof(main)/sizeof(main[0]);

   for (int i = 0; i < n; i++)
   {
      sorted_index[i] = i;
   }

   for (int i = 0; i < n; i++)
   {
      for (int j = i; j < n-1; j++)
      {
         if (main[sorted_index[j]] > main[sorted_index[j+1]])
         {
            int temp = sorted_index[j];
            sorted_index[j] = sorted_index[j+1];
            sorted_index[j+1] = temp;
         }
      }
   }
}
```

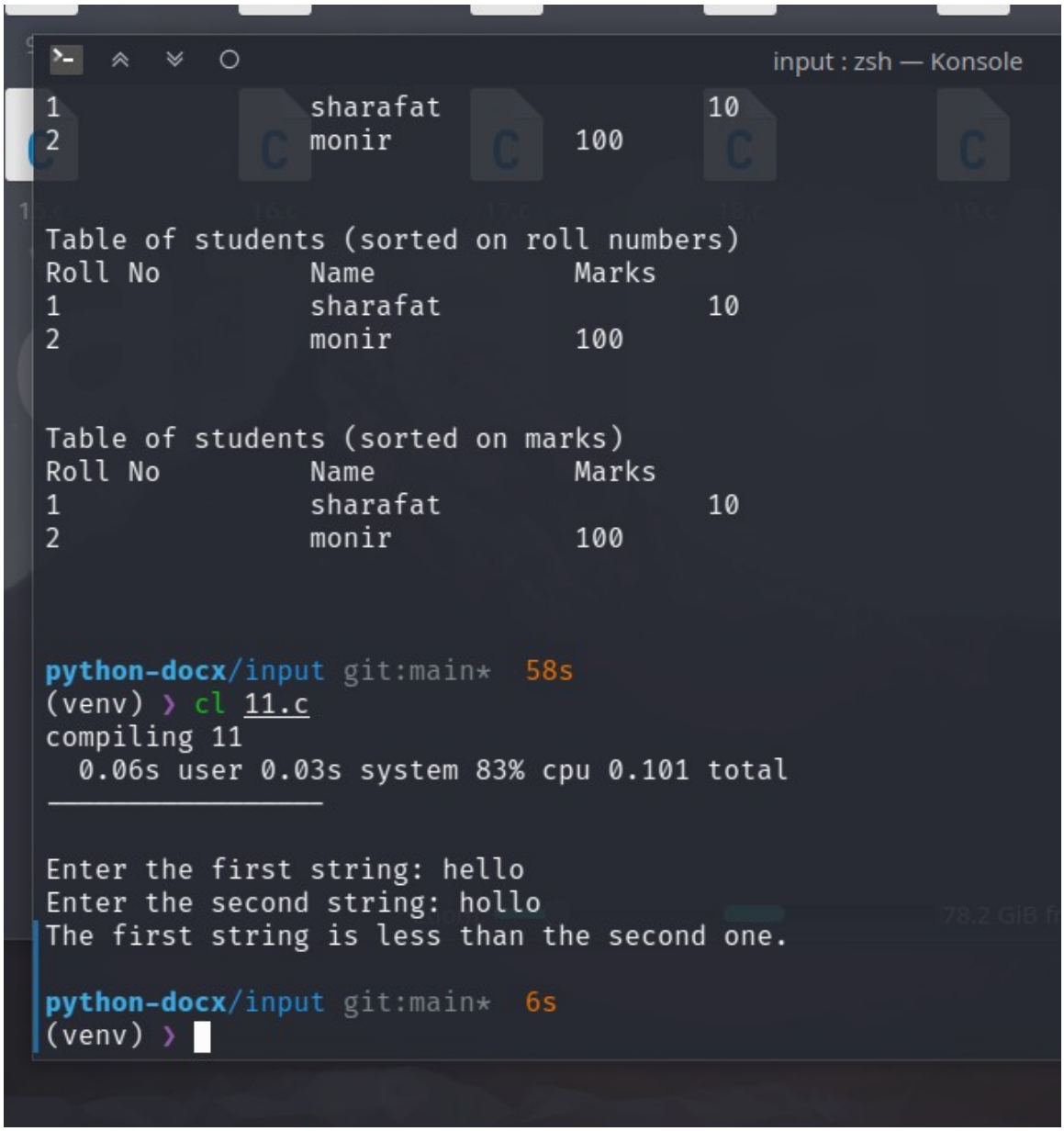
```
int main()
{
   int n;
   printf("Enter the number of students: ");
   scanf("%d", &n);
   struct student students[n];
   for (int i = 0; i < n; i++)
   {
      printf("Enter the roll number of student %d: ", i+1);
      scanf("%d", &students[i].roll_no);
      printf("Enter the name of student %d: ", i+1);
      scanf("%s", students[i].name);
      printf("Enter the marks of student %d: ", i+1);
      scanf("%d", &students[i].marks);
   }

   printf("\n\n");
   printf("Table of students (alphabetic list)\n");
   printf("Roll No\t\tName\t\tMarks\n");
   for (int i = 0; i < n; i++)
   {
      printf("%d\t\t%s\t\t%d\n", students[i].roll_no, students[i].name,
students[i].marks);
   }

   printf("\n\n");
   printf("Table of students (sorted on roll numbers)\n");
   int sorted_roll_no_index[n];
   int students_roll_no[n];
   for (int i = 0; i < n; i++)
   {
      students_roll_no[i] = students[i].roll_no;
   }
   sorting_array_with_index(students_roll_no, sorted_roll_no_index);
   printf("Roll No\t\tName\t\tMarks\n");
   for (int i = 0; i < n; i++)
   {
```

```
        printf("%d\t\t%s\t\t%d\n", students[sorted_roll_no_index[i]].roll_no,
students[sorted_roll_no_index[i]].name, students[sorted_roll_no_index[i]].marks);
    }
    printf("\n\n");
    printf("Table of students (sorted on marks)\n");
    int sorted_marks_index[n];
    int students_marks[n];
    for (int i = 0; i < n; i++)
    {
        students_marks[i] = students[i].marks;
    }
    sorting_array_with_index(students_marks, sorted_marks_index);
    printf("Roll No\t\tName\t\tMarks\n");
    for (int i = 0; i < n; i++)
    {
        printf("%d\t\t%s\t\t%d\n", students[sorted_marks_index[i]].roll_no,
students[sorted_marks_index[i]].name, students[sorted_marks_index[i]].marks);
    }
    printf("\n\n");
    return 0;
}
```


input : zsh — Konsole
Enter the roll number of student 1: 1
Enter the name of student 1: sharafat
Enter the marks of student 1: 10
Enter the roll number of student 2: 2
Enter the name of student 2: monir
Enter the marks of student 2: 100
Table of students (alphabetic list)
Roll No Name Marks
1 sharafat 10
2 monir 100
Table of students (sorted on roll numbers)
Roll No Name Marks
1 sharafat 10
2 monir 100
Table of students (sorted on marks)
Roll No Name Marks
1 sharafat 10
2 monir 100
python-docx/input git:main* 58s
(venv) ❯

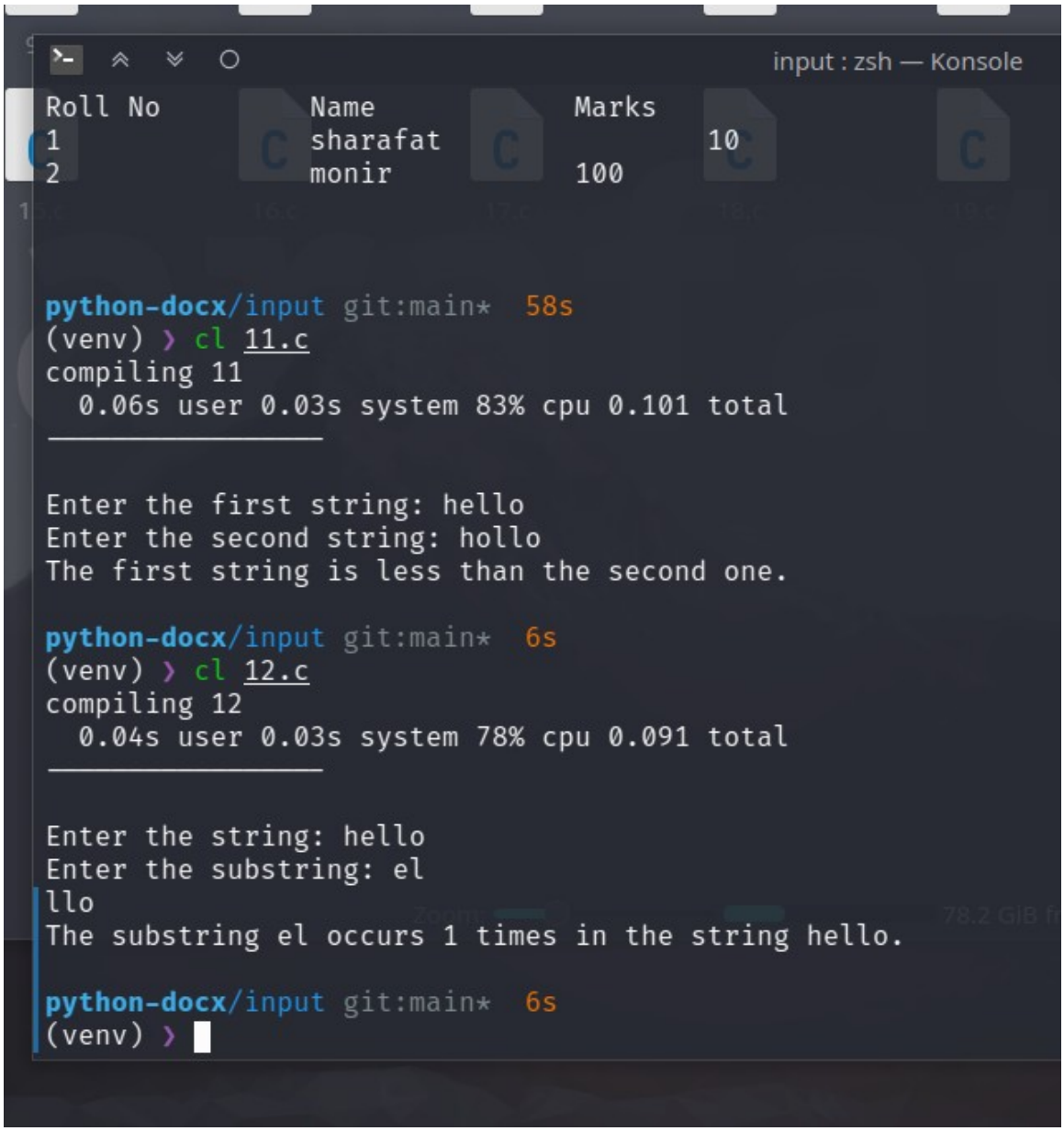
11 Write a program to read two strings and compare them using the function strncmp( ) and print a message that the first string is equal, less, or greater than the second one.

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

int main()
{
   char str1[100], str2[100];
   printf("Enter the first string: ");
   scanf("%s", str1);
   printf("Enter the second string: ");
   scanf("%s", str2);

   int result = strncmp(str1, str2, 100);
   if (result == 0)
   {
      printf("The first string is equal to the second one.\n");
   }
   else if (result < 0)
   {
      printf("The first string is less than the second one.\n");
   }
   else
   {
      printf("The first string is greater than the second one.\n");
   }
}
```


input : zsh — Konsole
1 sharafat 10
2 monir 100
Table of students (sorted on roll numbers)
Roll No Name Marks
1 sharafat 10
2 monir 100
Table of students (sorted on marks)
Roll No Name Marks
1 sharafat 10
2 monir 100
python-docx/input git:main* 58s
(venv) ❯ cl 11.c
compiling 11
0.06s user 0.03s system 83% cpu 0.101 total
Enter the first string: hello
Enter the second string: hollo
The first string is less than the second one.
python-docx/input git:main* 6s
(venv) ❯

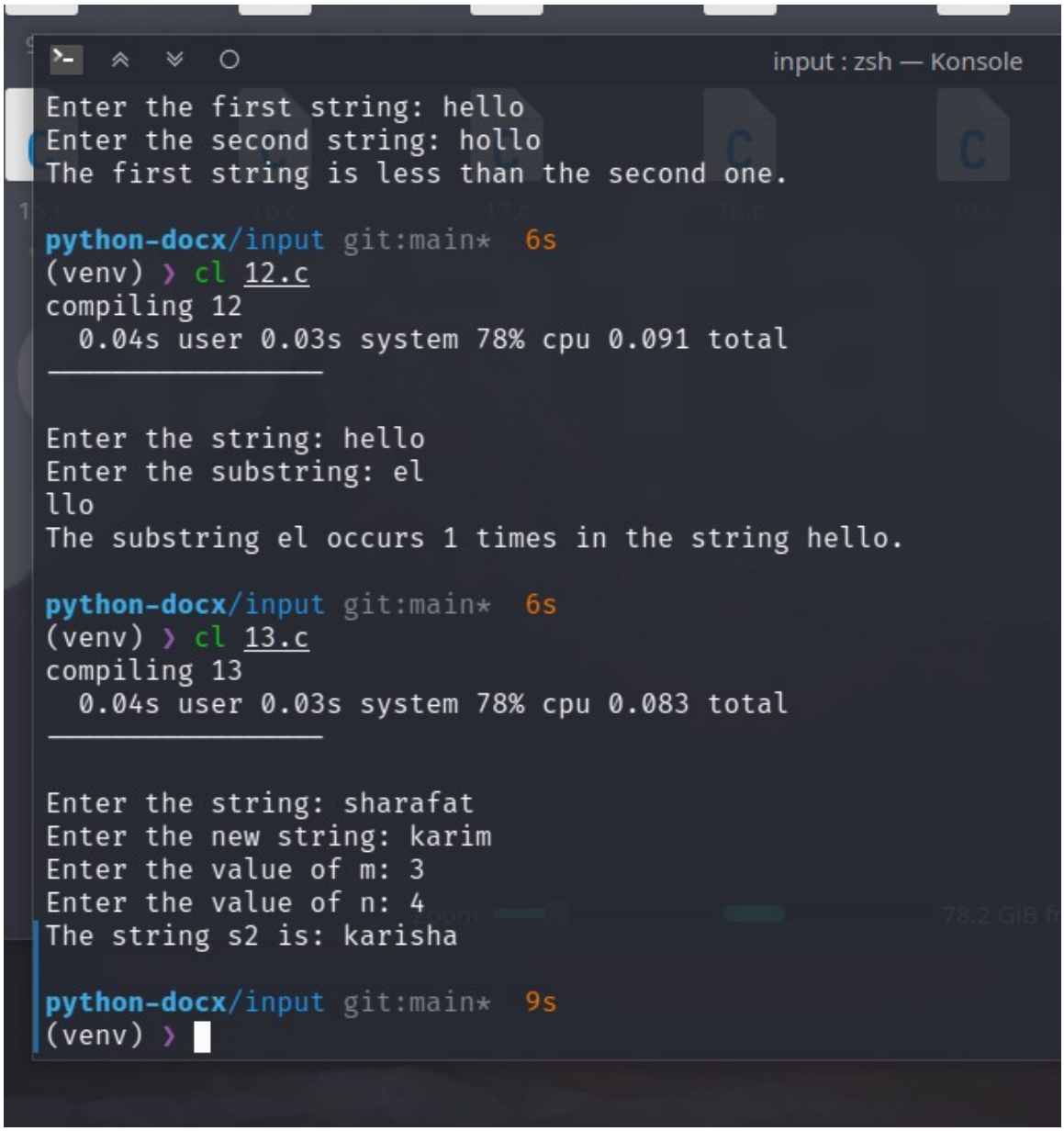
12 Write a program to read a line of text from the keyboard and print out the number of occurrences of a given substring using the function strstr ( ).

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

int main()
{
   char str[100], substr[100];
   printf("Enter the string: ");
   scanf("%[^\n]%*c", str);
   printf("Enter the substring: ");
   scanf("%[^\n]%*c", substr);

   int count = 0;
   char *ptr = strstr(str, substr);
   printf("%s\n", ptr+1);
   while (ptr != NULL)
   {
      count++;
      ptr = strstr(ptr+1, substr);
   }

   printf("The substring %s occurs %d times in the string %s.\n", substr, count, str);
}
```


input : zsh — Konsole
Roll No Name Marks
1 sharafat 10
2 monir 100
python-docx/input git:main* 58s
(venv) ❯ cl 11.c
compiling 11
0.06s user 0.03s system 83% cpu 0.101 total
Enter the first string: hello
Enter the second string: hollo
The first string is less than the second one.
python-docx/input git:main* 6s
(venv) ❯ cl 12.c
compiling 12
0.04s user 0.03s system 78% cpu 0.091 total
Enter the string: hello
Enter the substring: el
llo
The substring el occurs 1 times in the string hello.
python-docx/input git:main* 6s
(venv) ❯

13 Write a program that will copy m consecutive characters from a string s1 beginning at position n into another string s2.

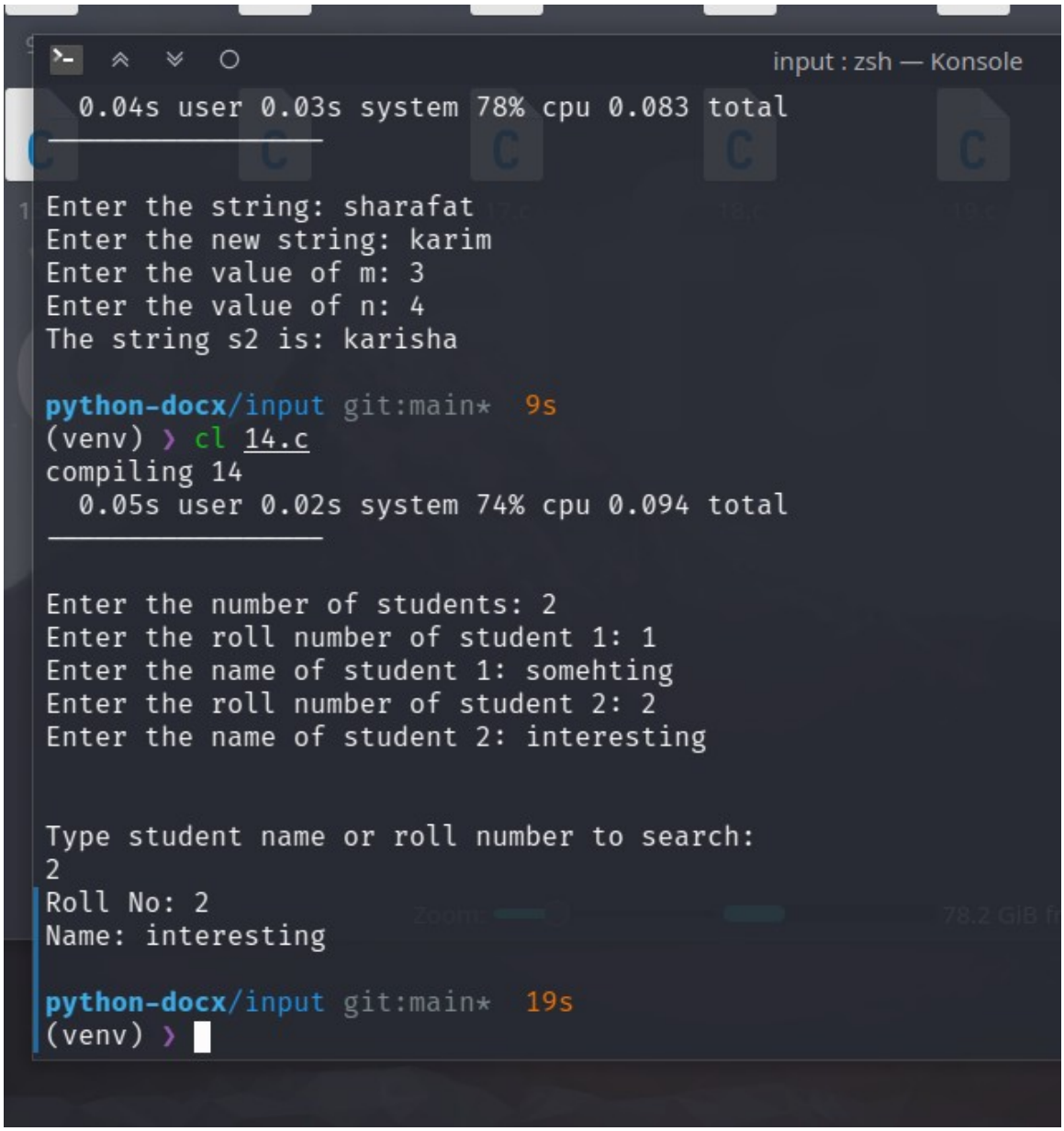
```
#include<stdio.h>
#include<string.h>

int main()
{
   char s1[100], s2[100];
   int m, n;
   printf("Enter the string: ");
   scanf("%[^\n]%*c", s1);
   printf("Enter the new string: ");
   scanf("%[^\n]%*c", s2);
   printf("Enter the value of m: ");
   scanf("%d", &m);
   printf("Enter the value of n: ");
   scanf("%d", &n);

   strncpy(s2+n, s1, m);
   printf("The string s2 is: %s\n", s2);
}
```


input : zsh — Konsole
Enter the first string: hello
Enter the second string: hollo
The first string is less than the second one.
python-docx/input git:main* 6s
(venv) ❯ cl 12.c
compiling 12
0.04s user 0.03s system 78% cpu 0.091 total
Enter the string: hello
Enter the substring: el
llo
The substring el occurs 1 times in the string hello.
python-docx/input git:main* 6s
(venv) ❯ cl 13.c
compiling 13
0.04s user 0.03s system 78% cpu 0.083 total
Enter the string: sharafat
Enter the new string: karim
Enter the value of m: 3
Enter the value of n: 4
The string s2 is: karisha
python-docx/input git:main* 9s
(venv) ❯

14 Write a program to create a directory of students with roll numbers. The program should display the roll number for a specified name and vice-versa.

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

struct student
{
   int roll_no;
   char name[50];
};

int main()
{
   struct student students[100];
   int n;
   printf("Enter the number of students: ");
   scanf("%d", &n);

   for (int i = 0; i < n; i++)
   {
      printf("Enter the roll number of student %d: ", i+1);
      scanf("%d", &students[i].roll_no);
      printf("Enter the name of student %d: ", i+1);
      scanf("%s", students[i].name);
   }

   printf("\n\n");
   printf("Type student name or roll number to search: \n");
   char search[50];
   scanf("%s", search);

   int found = 0;
```

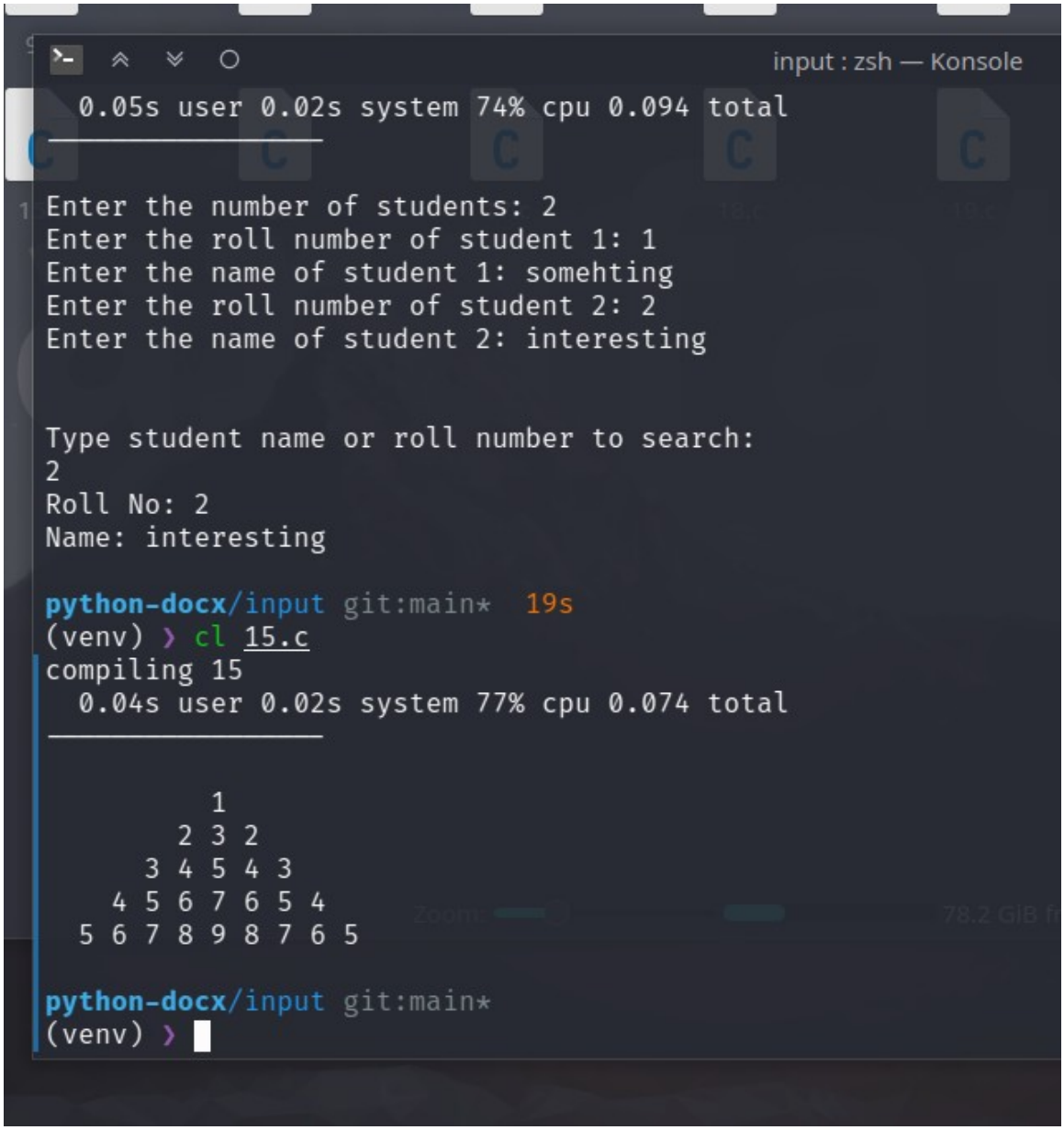
```
    for (int i = 0; i < n; i++)
    {
        if (students[i].roll_no == atoi(search) || strcmp(students[i].name, search) == 0)
        {
            printf("Roll No: %d\n", students[i].roll_no);
            printf("Name: %s\n", students[i].name);
            found = 1;
            break;
        }
    }
}
```


input : zsh — Konsole
0.04s user 0.03s system 78% cpu 0.083 total
Enter the string: sharafat
Enter the new string: karim
Enter the value of m: 3
Enter the value of n: 4
The string s2 is: karisha
python-docx/input git:main* 9s
(venv) ❯ cl 14.c
compiling 14
0.05s user 0.02s system 74% cpu 0.094 total
Enter the number of students: 2
Enter the roll number of student 1: 1
Enter the name of student 1: somehting
Enter the roll number of student 2: 2
Enter the name of student 2: interesting
Type student name or roll number to search:
2
Roll No: 2
Name: interesting
python-docx/input git:main* 19s
(venv) ❯

15 Given a string make a pyramid out of it

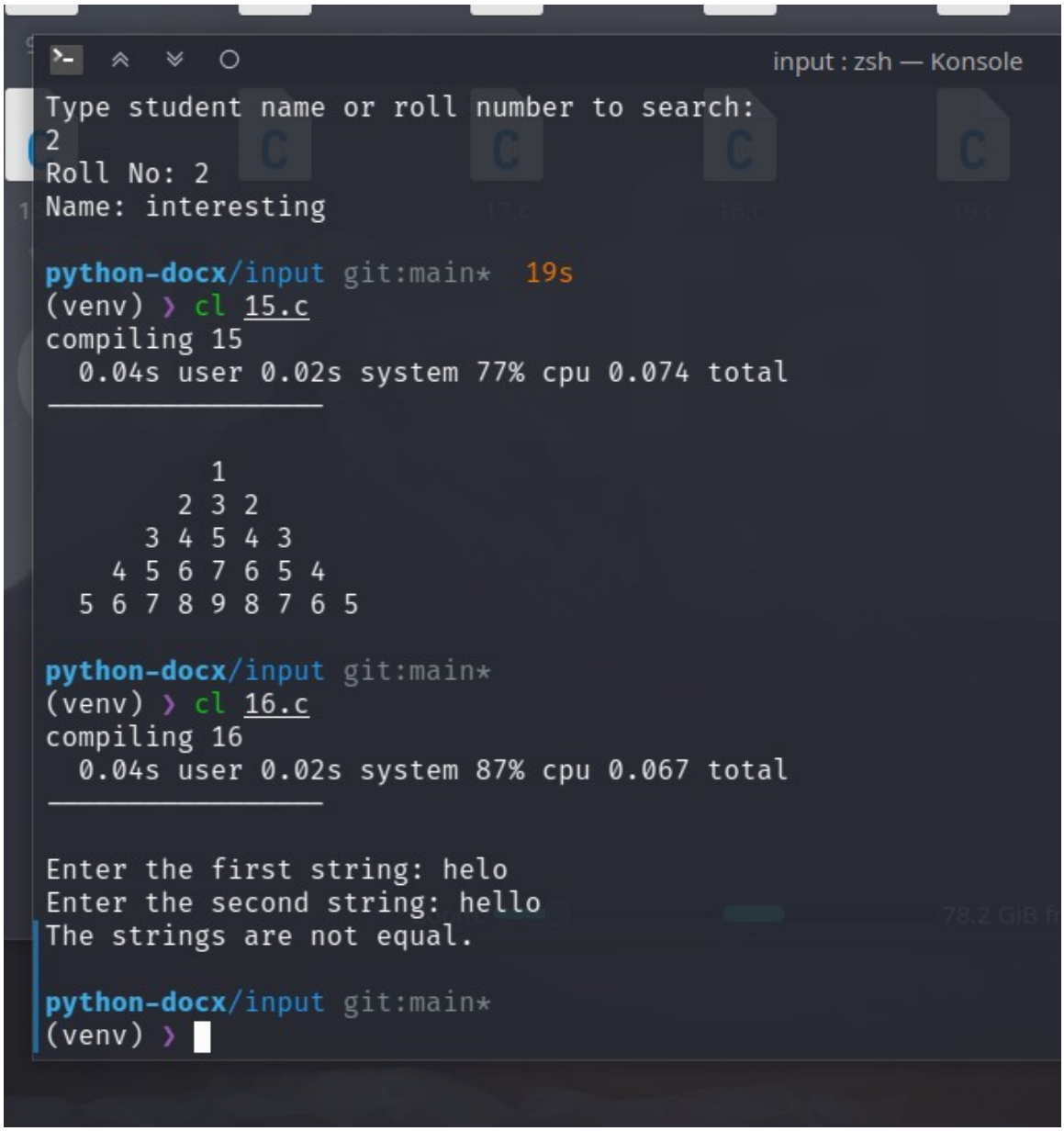
```
// Given a string
// ```
//         char str [ ] = “123456789” ;
// ```
// Write a program that displays the following:
// ```
//              1
//            2 3 2
//          3 4 5 4 3
//        4 5 6 7 6 5 4
//      5 6 7 8 9 8 7 6 5
// ```

#include <stdio.h>

int main()
{
   char str[] = "123456789";
   int n = 5;
   for (int i = 0; i < n; i++)
   {
      for (int j = n-i; j > 0; j--)
      {
         printf("  ");
      }
      for (int j = i+1; j < 2*i+2; j++)
      {
         printf("%c ", str[j-1]);
      }
      for (int j = 2*i; j > i; j--)
      {
         printf("%c ", str[j-1]);
      }
      printf("\n");
```

```
  }
}
```


input : zsh — Konsole
0.05s user 0.02s system 74% cpu 0.094 total
Enter the number of students: 2
Enter the roll number of student 1: 1
Enter the name of student 1: somehting
Enter the roll number of student 2: 2
Enter the name of student 2: interesting
Type student name or roll number to search:
2
Roll No: 2
Name: interesting
python-docx/input git:main* 19s
(venv) ❯ cl 15.c
compiling 15
0.04s user 0.02s system 77% cpu 0.074 total
1
2 3 2
3 4 5 4 3
4 5 6 7 6 5 4
5 6 7 8 9 8 7 6 5
python-docx/input git:main*
(venv) ❯

16 Write a C program to compare two strings without using any string function.

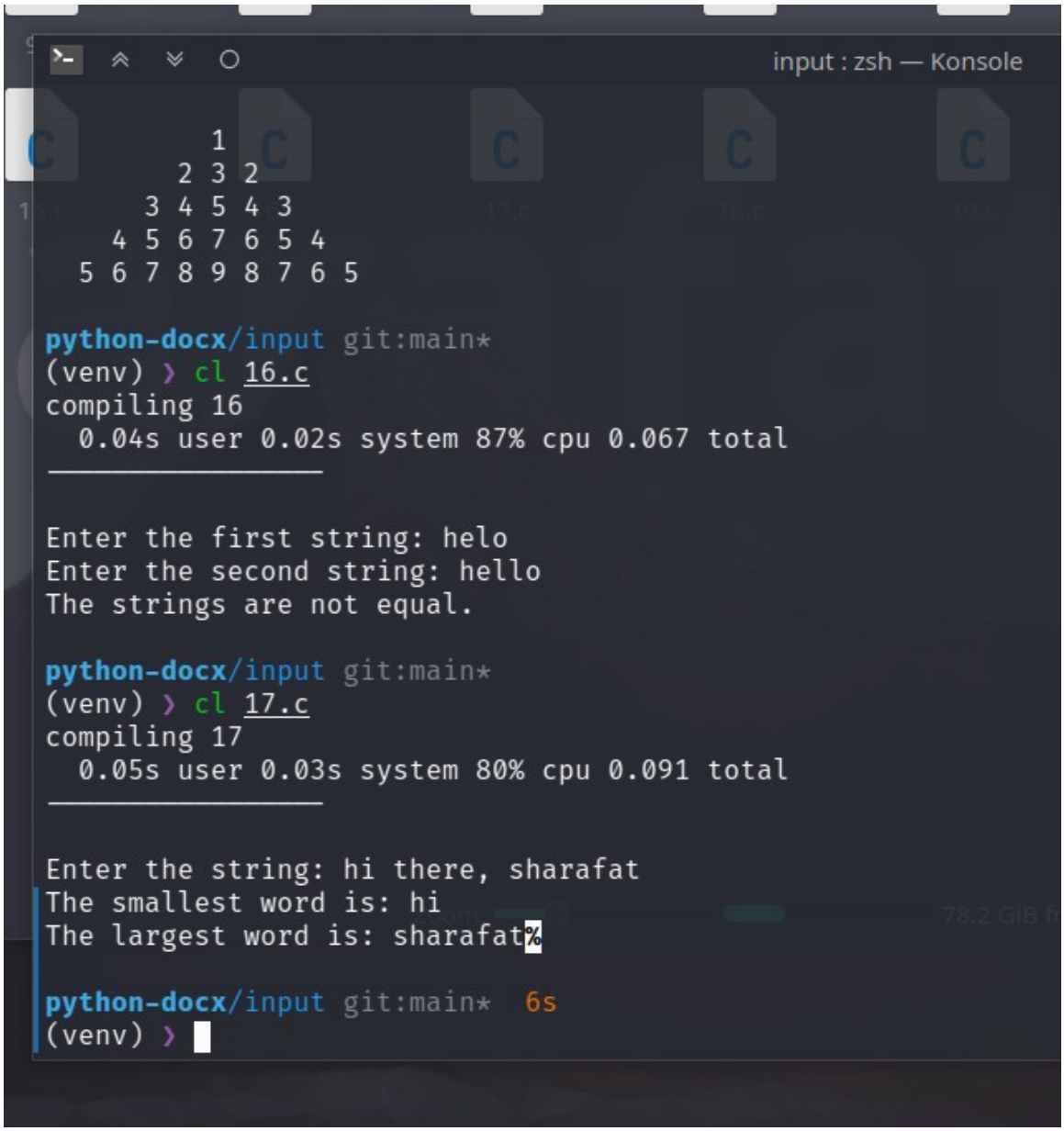
```
// Write a C program to compare two strings without using any string function.

#include <stdio.h>

int main()
{
   char str1[100], str2[100];
   printf("Enter the first string: ");
   scanf("%[^\n]%*c", str1);
   printf("Enter the second string: ");
   scanf("%[^\n]%*c", str2);

   int i = 0;
   while (str1[i] != '\0' && str2[i] != '\0')
   {
      if (str1[i] != str2[i])
      {
         printf("The strings are not equal.\n");
         return 0;
      }
      i++;
   }
   if (str1[i] == '\0' && str2[i] == '\0')
   {
      printf("The strings are equal.\n");
   }
   else
   {
      printf("The strings are not equal.\n");
   }
}
```


input : zsh — Konsole
Type student name or roll number to search:
2
Roll No: 2
Name: interesting
python-docx/input git:main* 19s
(venv) ❯ cl 15.c
compiling 15
0.04s user 0.02s system 77% cpu 0.074 total
1
2 3 2
3 4 5 4 3
4 5 6 7 6 5 4
5 6 7 8 9 8 7 6 5
python-docx/input git:main*
(venv) ❯ cl 16.c
compiling 16
0.04s user 0.02s system 87% cpu 0.067 total
Enter the first string: helo
Enter the second string: hello
The strings are not equal.
python-docx/input git:main*
(venv) ❯

17 Write a C program to find the largest and smallest sized word in a string.

```
// Write a C program to find the largest and smallest sized word in a string.

#include <stdio.h>

int main()
{
   char str[100];
   printf("Enter the string: ");
   scanf("%[^\n]%*c", str);

   int i = 0;
   int min = 100, max = 0;
   int min_index = 0, max_index = 0;
   while (str[i] != '\0')
   {
      int count = 0;
      while (str[i] != ' ' && str[i] != '\0')
      {
         count++;
         i++;
      }
      if (count < min)
      {
         min = count;
         min_index = i - count;
      }
      if (count > max)
      {
         max = count;
         max_index = i - count;
      }
      if (str[i] == '\0')
```

```
        {
            break;
        }
        i++;
    }

    printf("The smallest word is: ");
    for (int j = min_index; j < min_index + min; j++)
    {
        printf("%c", str[j]);
    }
    printf("\n");

    printf("The largest word is: ");
    for (int j = max_index; j < max_index + max; j++)
    {
        printf("%c", str[j]);
    }
}
```


input : zsh — Konsole
1
2 3 2
3 4 5 4 3
4 5 6 7 6 5 4
5 6 7 8 9 8 7 6 5
python-docx/input git:main*
(venv) ❯ cl 16.c
compiling 16
0.04s user 0.02s system 87% cpu 0.067 total
Enter the first string: helo
Enter the second string: hello
The strings are not equal.
python-docx/input git:main*
(venv) ❯ cl 17.c
compiling 17
0.05s user 0.03s system 80% cpu 0.091 total
Enter the string: hi there, sharafat
The smallest word is: hi
The largest word is: sharafat%
python-docx/input git:main* 6s
(venv) ❯

18 Write a C program to replace all the white spaces in a string with double white spaces.

```
// Write a C program to replace all the white spaces in a string with double white spaces.

#include <stdio.h>

int main()
{
   char str[100];
   printf("Enter the string: ");
   scanf("%[^\n]%*c", str);
   char white_space_doubled_str[200];

   int i = 0, j = 0;
   while (str[i] != '\0')
   {
      if (str[i] == ' ')
      {
         white_space_doubled_str[j] = ' ';
         white_space_doubled_str[j+1] = ' ';
         j += 2;
      }
      else
      {
         white_space_doubled_str[j] = str[i];
         j++;
      }
      i++;
   }

   white_space_doubled_str[j] = '\0';
   printf("The string with double white spaces is: %s\n", white_space_doubled_str);
```

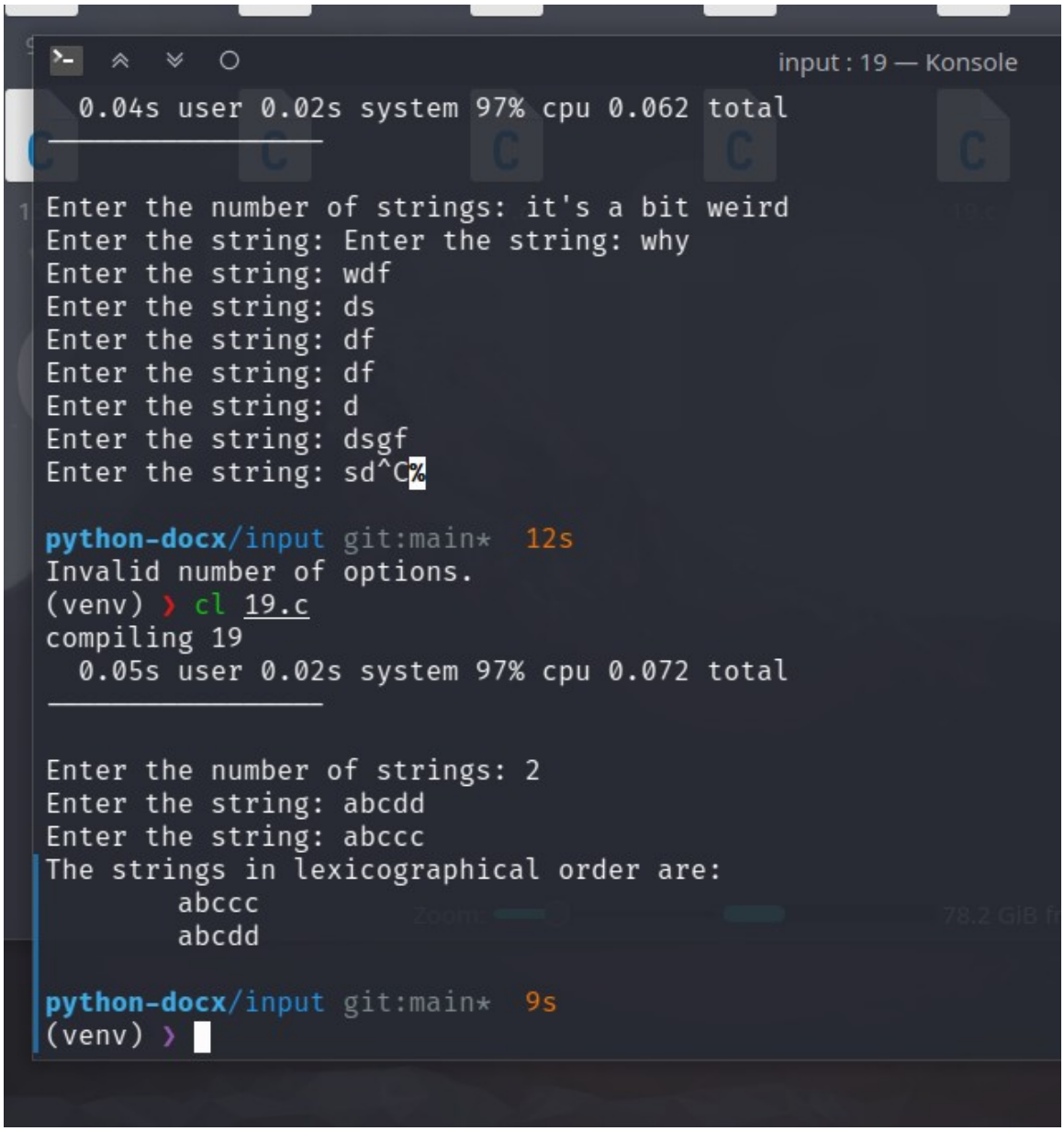
```
    return 0;
}
```


input : zsh — Konsole
compiling 16
0.04s user 0.02s system 87% cpu 0.067 total
Enter the first string: helo
Enter the second string: hello
The strings are not equal.
python-docx/input git:main*
(venv) ❯ cl 17.c
compiling 17
0.05s user 0.03s system 80% cpu 0.091 total
Enter the string: hi there, sharafat
The smallest word is: hi
The largest word is: sharafat%
python-docx/input git:main* 6s
(venv) ❯ cl 18.c
compiling 18
0.04s user 0.02s system 80% cpu 0.083 total
Enter the string: hello world
The string with double white spaces is: hello world
python-docx/input git:main*
(venv) ❯

19 Write a C program to enter multiple strings and display them in lexicographical order.

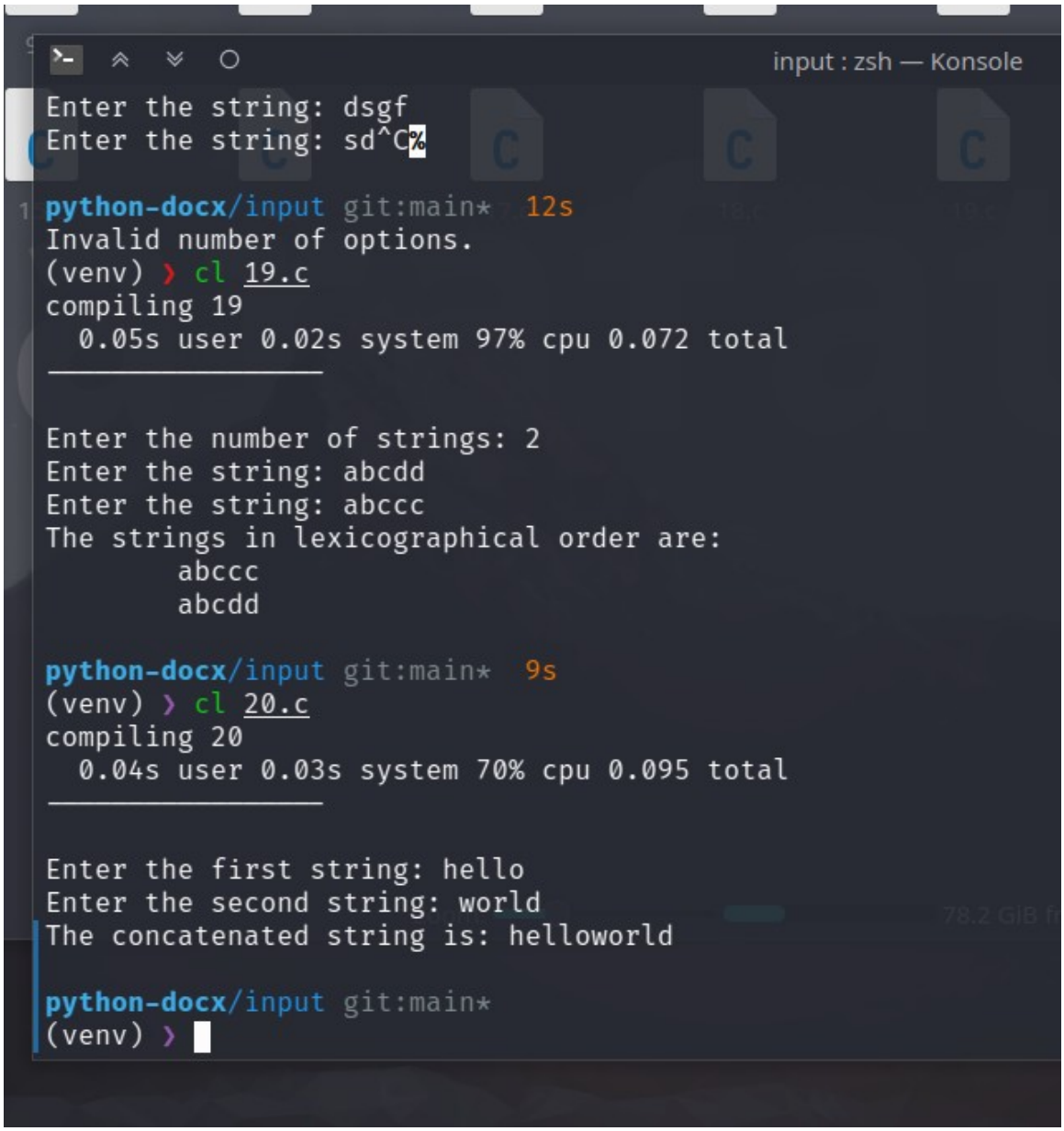
```
// Write a C program to enter multiple strings and display them in lexicographical order.

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

int main()
{
   int n;
   printf("Enter the number of strings: ");
   scanf("%d%*c", &n);
   char str[n][100];
   for (int i = 0; i < n; i++)
   {
      printf("Enter the string: ");
      scanf("%[^\n]%*c", &str[i]);
   }

   for (int i = 0; i < n; i++)
   {
      for (int j = i+1; j < n; j++)
      {
         if (strcmp(str[i], str[j]) > 0)
         {
            char temp[100];
            strcpy(temp, str[i]);
            strcpy(str[i], str[j]);
            strcpy(str[j], temp);
         }
      }
   }
```

```
    printf("The strings in lexicographical order are:\n");
    for (int i = 0; i < n; i++)
    {
        printf("\t%s\n", str[i]);
    }
}
```



20 Write a C program to concatenate two strings without using any string function.

```
// Write a C program to concatenate two strings without using any string function.

#include <stdio.h>

int main()
{
   char str1[100], str2[100];
   printf("Enter the first string: ");
   scanf("%[^\n]%*c", str1);
   printf("Enter the second string: ");
   scanf("%[^\n]%*c", str2);

   int i = 0;
   while (str1[i] != '\0')
   {
      i++;
   }
   int j = 0;
   while (str2[j] != '\0')
   {
      str1[i] = str2[j];
      i++;
      j++;
   }
   str1[i] = '\0';
   printf("The concatenated string is: %s\n", str1);
}
```


input : zsh — Konsole
Enter the string: dsgf
Enter the string: sd^C%
python-docx/input git:main* 12s
Invalid number of options.
(venv) ❯ cl 19.c
compiling 19
0.05s user 0.02s system 97% cpu 0.072 total
Enter the number of strings: 2
Enter the string: abcdd
Enter the string: abccc
The strings in lexicographical order are:
abccc
abcdd
python-docx/input git:main* 9s
(venv) ❯ cl 20.c
compiling 20
0.04s user 0.03s system 70% cpu 0.095 total
Enter the first string: hello
Enter the second string: world
The concatenated string is: helloworld
python-docx/input git:main*
(venv) ❯