Name : Mukul Kumar

Enrollment number : 201B162

Batch : B5

Subject : OOPL

Lab number : 1

Semester : 2

1. Write a program to round off an integer “i” to the next largest multiple of another

integer “j”. For example, 256 days when rounded off to the next largest multiple divisible

by a week result into 259.

Code :

#include <stdio.h>

int main()

{

int i, j, lower;

scanf(“%d %d”, &i, &j);

lower = i % j;

printf(“%d”, i + j – lower);

}



2. A number is entered through the keyboard. The number may contain 1,2,3,4, or 5

digits. Write a program to find the number of digits in the number.

Code :

#include <stdio.h>

int main()

{

int i,count=0,i\_backup,tester;

scanf("%d",&i);

i\_backup= i;

while(i\_backup!=0)

{

tester=i\_backup%10;

i\_backup/=10;

if(tester==9 || tester == 8|| tester == 7|| tester == 6)

{

printf("Invalid number");

return 0;

}

count++;

}

printf("%d",count);

return 0;

}



3. Write a program which finds a four-digit number AABB which is a perfect square. A

and B represent different digits. For example, 7744 is a four-digit perfect square number

which is also satisfying the condition AABB i.e. first two digits (AA=77) are same and

last two digits (BB=44) are same.

Code :

#include <stdio.h>

int main()

{

for (int i = 1111; i <= 9999; i++)

{

if ((i / 1000) % 10 == (i / 100) % 10 && (i / 10) % 10 == i % 10)

{

for (int j = 1; j <= i; j++)

{

if (j \* j == i)

{

printf("%d\n", i);

break;

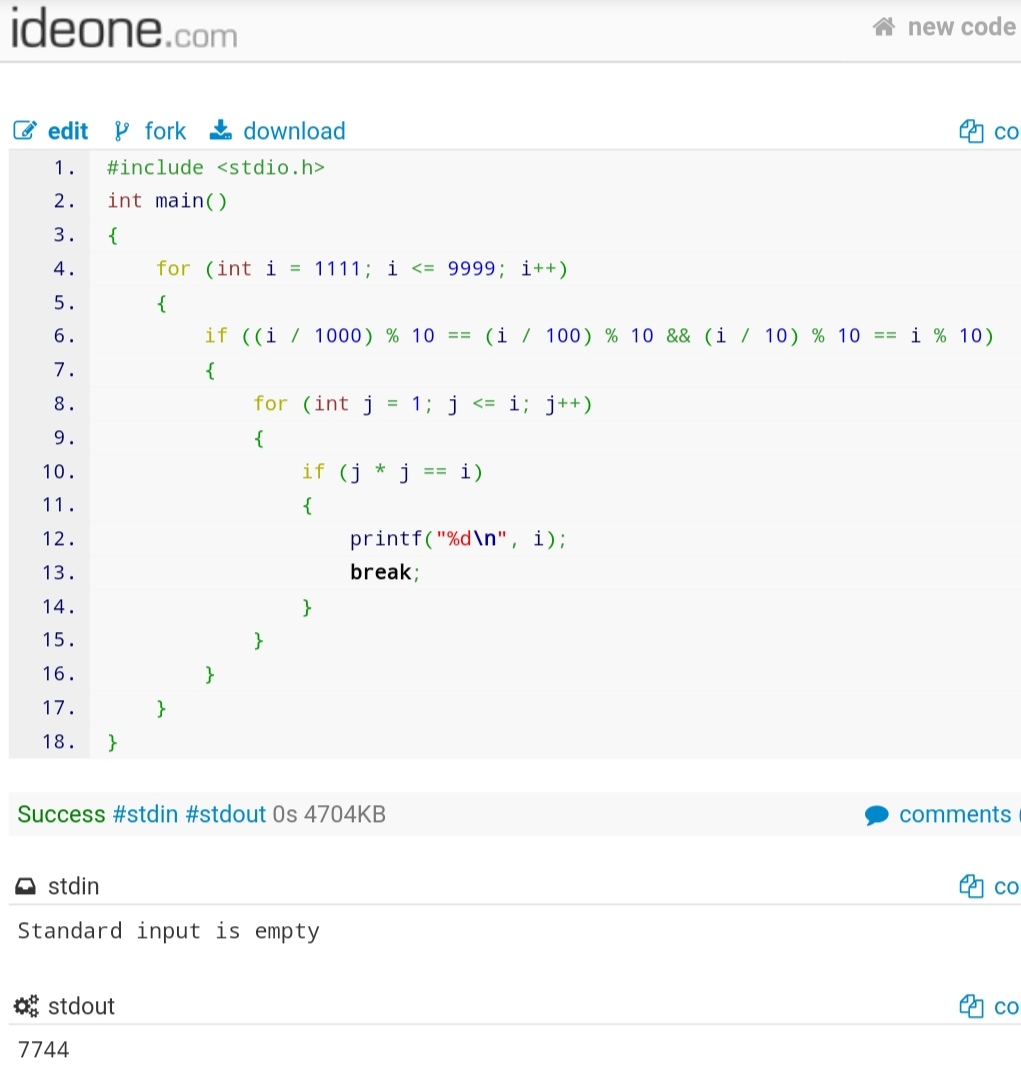
}

}

}

}

}



4. Write a program to calculate factorial of a number N through recursion.

Code :

#include <stdio.h>

int factorial(int);

int main()

{

int user\_input\_number;

scanf("%d", &user\_input\_number);

printf("%d", factorial(user\_input\_number));

}

int factorial(int n)

{

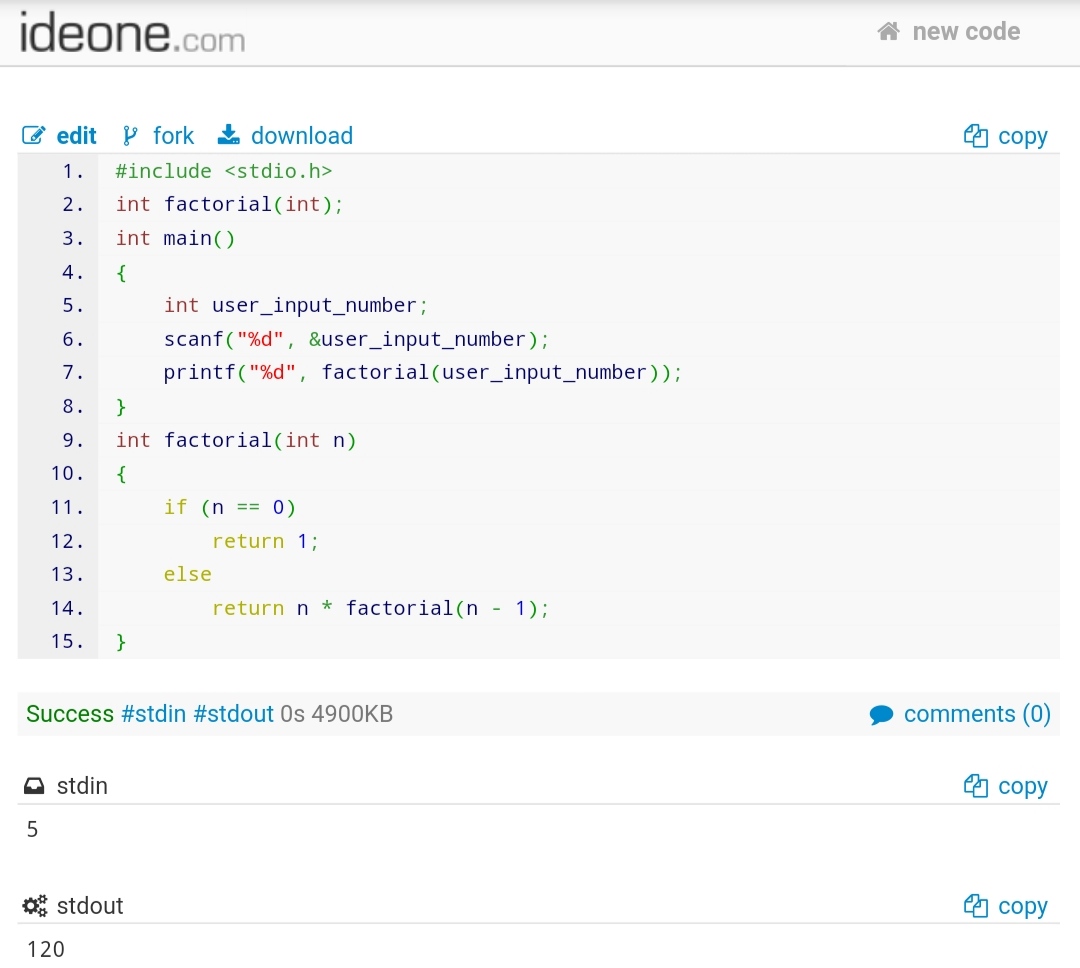
if (n == 0)

return 1;

else

return n \* factorial(n - 1);

}



5. Write a program which takes a string as input from user and returns the length of that

string without using any string library functions.

Code :

#include <stdio.h>

int main()

{

char string[1000];

int i;

scanf("%[^\n]%\*c",string);

for (i = 0; i < 1000; i++)

{

if (string[i] == '\0')

{

break;

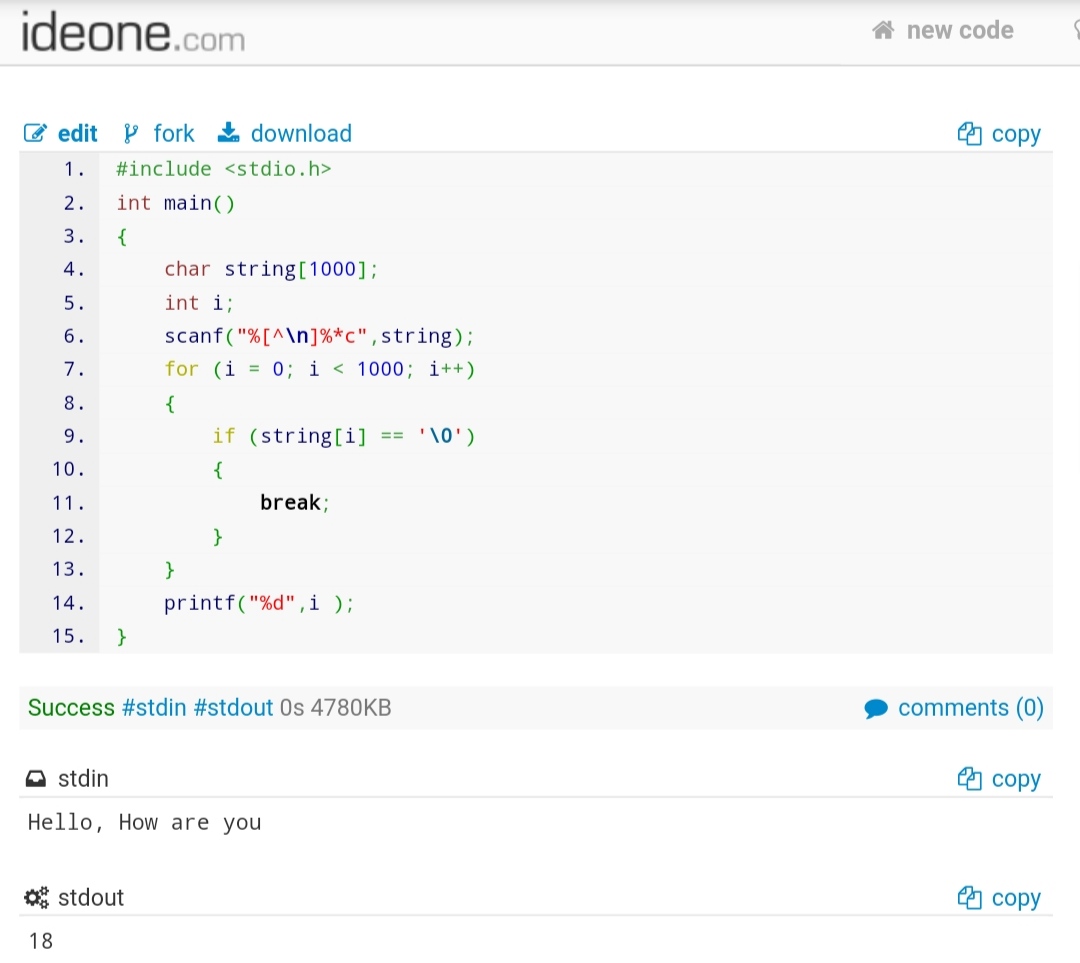
}

}

printf("%d",i );

return 0;

}



6. Write a pointer version of the function strcat(s,t) which concatenates the string t

to the end of string s.

Code :

#include <stdio.h>

int main()

{

char string1[100], string2[100];

int i, j;

scanf("%[^\n]%\*c", string1);

scanf("%[^\n]%\*c", string2);

for (i = 0; \*(string1 + i) != '\0'; ++i);

for (j = 0; \*(string2 + j) != '\0'; ++j, ++i)

{

\*(string1 + i) = \*(string2 + j);

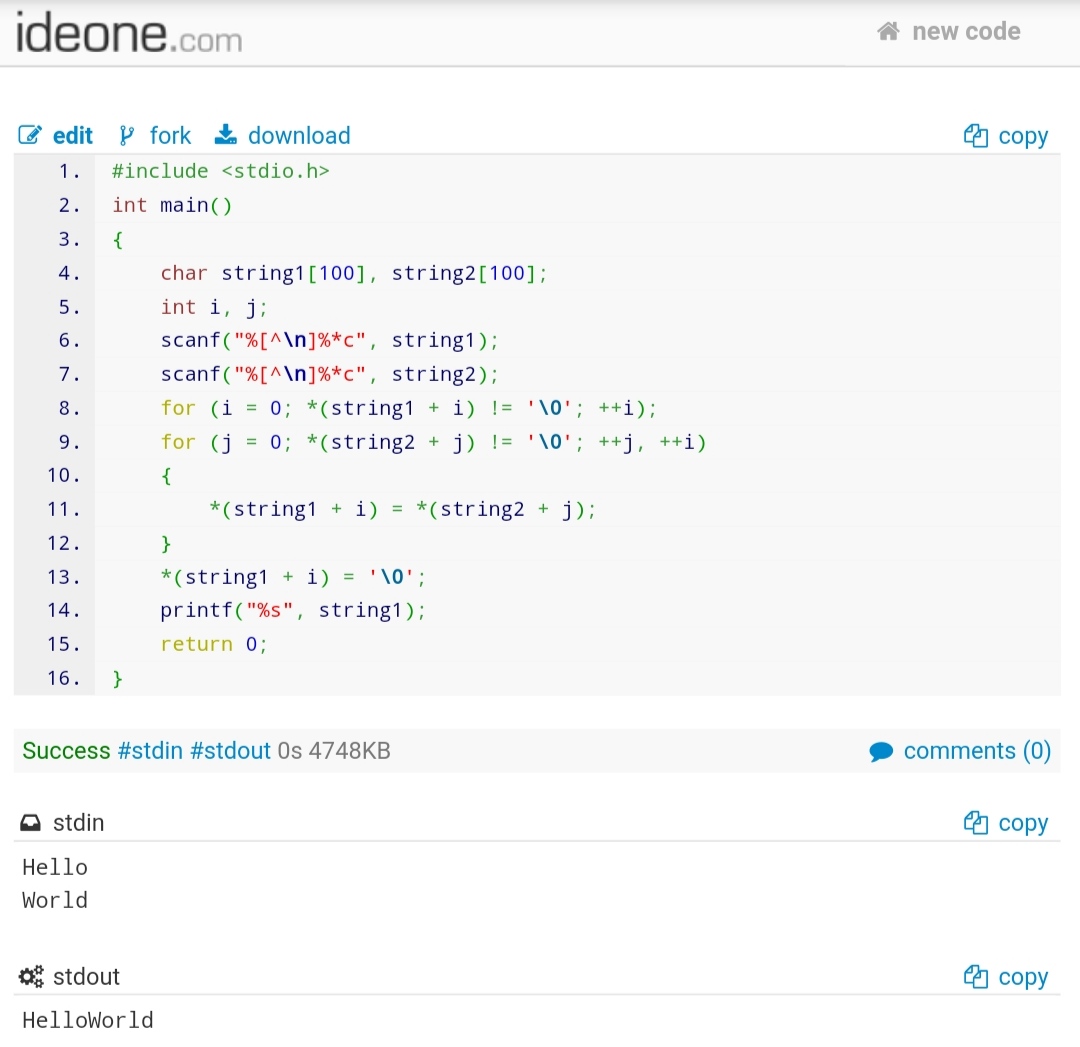
}

\*(string1 + i) = '\0';

printf("%s", string1);

return 0;

}



7. Write the function strend(s,t), which returns 1 if the string t occurs at the end of

the string s, and zero otherwise.

Sample Test case1:

Input:

s=”Object Oriented Programming using C++”

t=”using C++”

Output: 1

Sample Test case2:

Input:

s=”Object Oriented Programming using C++”

t=”Programming”

Output: 0

Code :

#include <stdio.h>

#include <string.h>

Int strend(char[], char[]);

Int main()

{

Char array\_1[100] ;

Char array\_2[100] ;

Scanf(“%[^\n]%\*c”,array\_1);

Scanf(“%[^\n]%\*c”,array\_2);

Printf(“%d”, strend(array\_1, array\_2));

Return 0;

}

Int strend(char a[], char b[])

{

Int length\_a = strlen(a);

Int length\_b = strlen(b);

Int count = 0;

For (int I = length\_b, j = length\_a; I > 0; i--, j--)

{

If (a[j – 1] == b[I – 1])

{

Count++;

}

Else

{

Return 0;

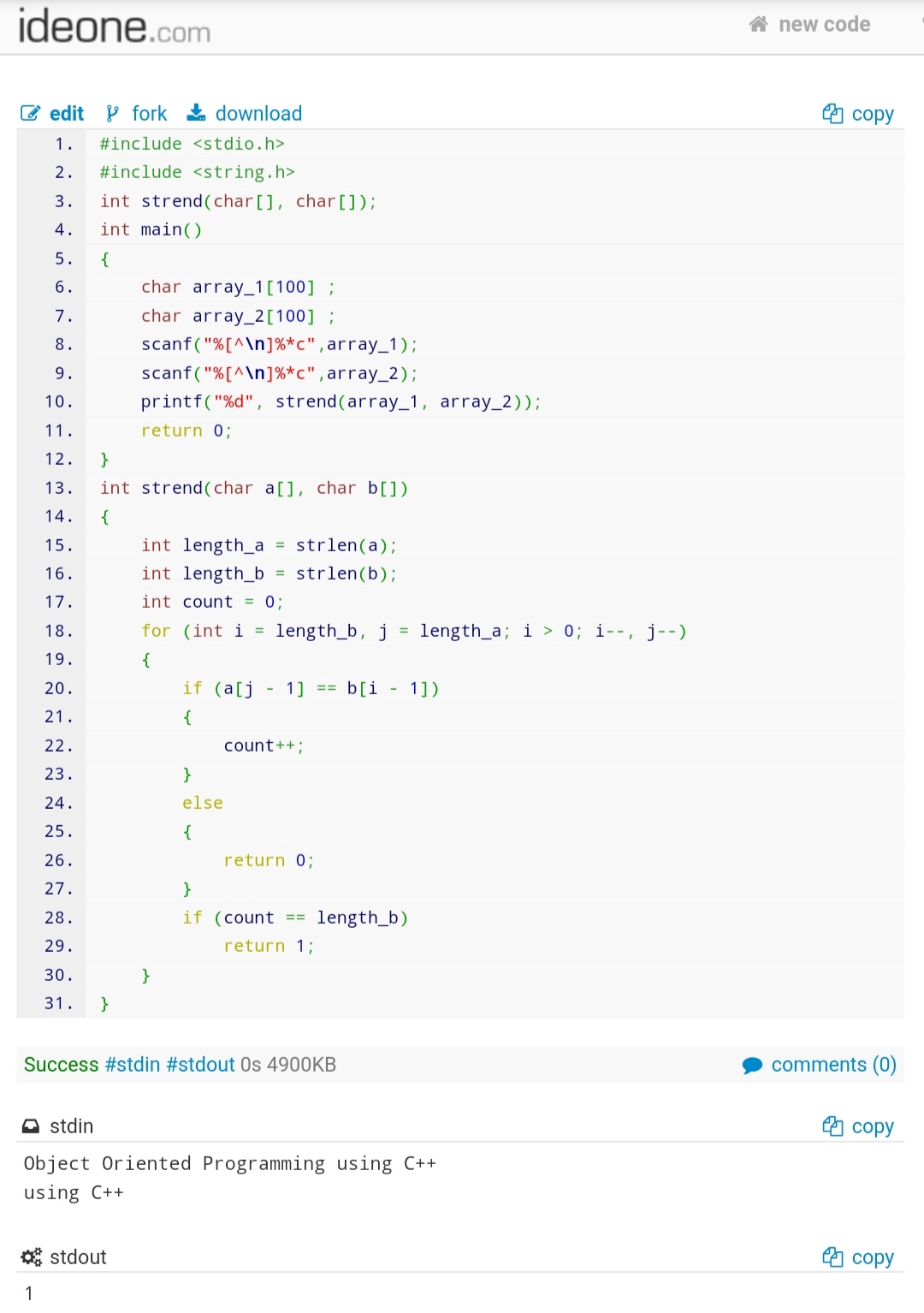
}

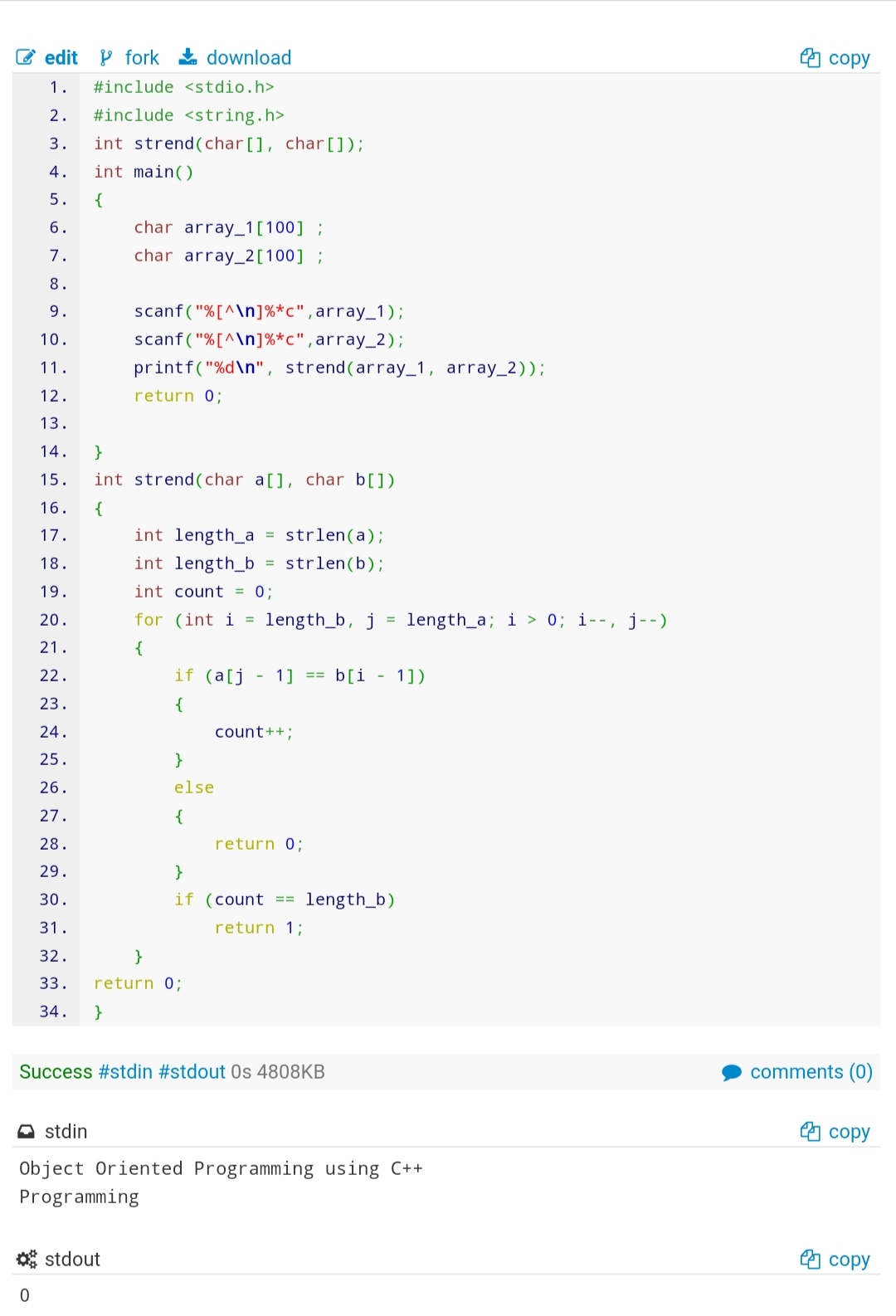
If (count == length\_b)

Return 1;

}

}





Advance practice problems

1. Write a program to find K’th smallest and K’th largest element in unsorted array.

Sample Test case1:

Input:

A[]=4, 5, 60, 70, 33, 44

K=2

Output: 2nd smallest number is 5 and 2nd largest number is 60

Sample Test case2:

Input:

A[]=2, 46, 56, 68, 3, 34, 489, 457, 4545, 100

K=5

Output: 5th smallest number is 56 and 5th largest number is 68

Code :

#include <stdio.h>

int main()

{

int n;

scanf("%d", &n);

int array[n], k;

for (int i = 0; i < n; i++)

scanf("%d", &array[i]);

scanf("%d", &k);

for (int i = 0; i < k; i++)

{

for (int j = i + 1; j < n; j++)

{

if (array[j] < array[i])

{

int c = array[j];

array[j] = array[i];

array[i] = c;

}

}

}

for (int i = n - k - 1; i < n; i++)

{

for (int j = k - 1 ; j < n ; j++ )

{

if (array[i] < array[j])

{

int c = array[i];

array[i] = array[j];

array[j] = c;

}

}

}

if (k==1)

printf("%dst smallest number is %d and %dst largest number is %d",k,array[k-1],k,array[n-k]);

else if (k==2)

printf("%dnd smallest number is %d and %dnd largest number is %d",k,array[k-1],k,array[n-k]);

else if (k==3)

printf("%drd smallest number is %d and %drd largest number is %d",k,array[k-1],k,array[n-k]);

else

printf("%dth smallest number is %d and %dth largest number is %d",k,array[k-1],k,array[n-k]);

return 0;

}

