計算機程式語言

教課教授 : 謝東儒

助教 :蔡詠聿、吳品頤

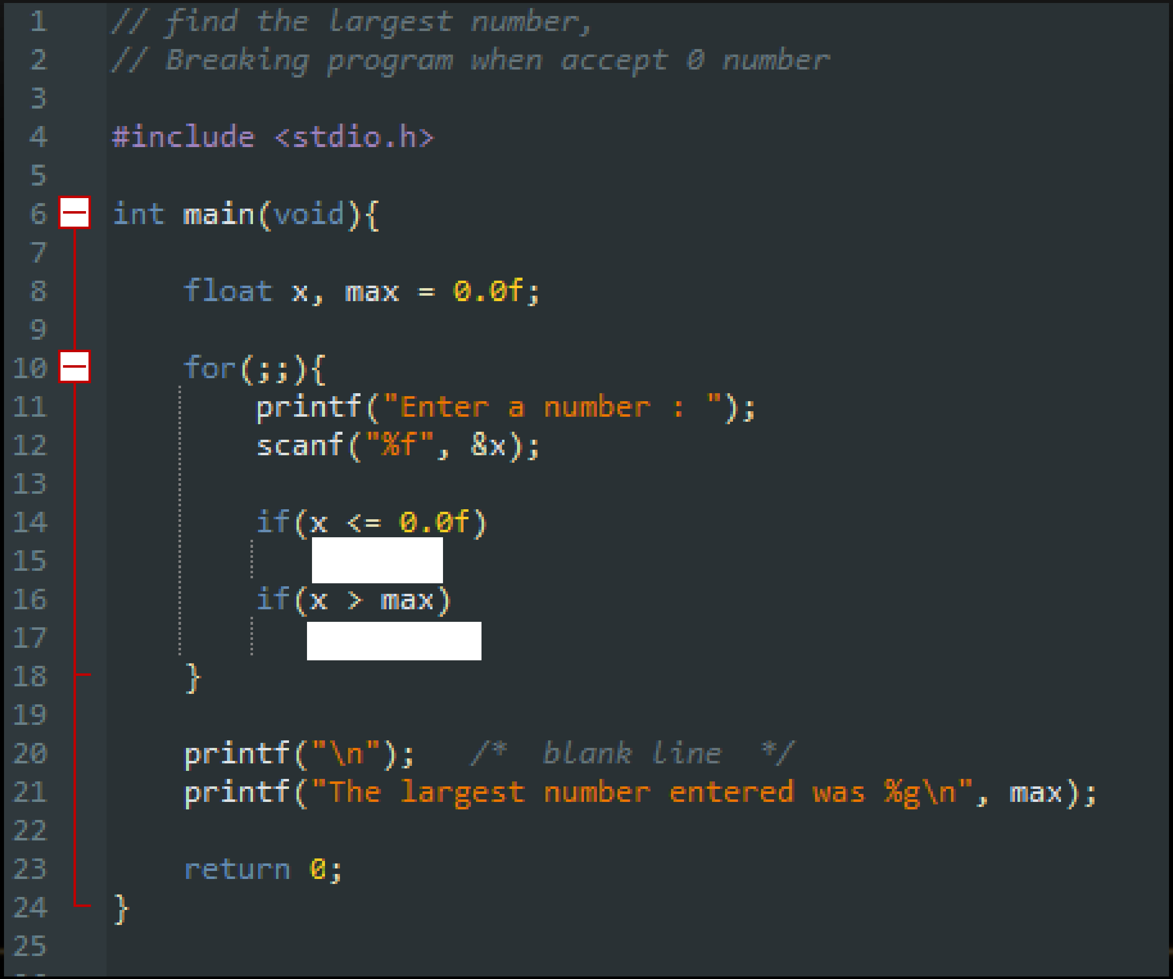
Chapter 6_project 1

Write a program that finds the largest in a series of numbers entered by the user. The program must prompt the user to enter numbers one by one. When the user enters 0 or a negative number, the program must display the largest nonnegative number entered:

```
    Enter a number: 60
    Enter a number: 38.3
    Enter a number: 4.89
    Enter a number: 100.62
    Enter a number: 75.2295
    Enter a number: 0

The largest number entered was 100.62
```

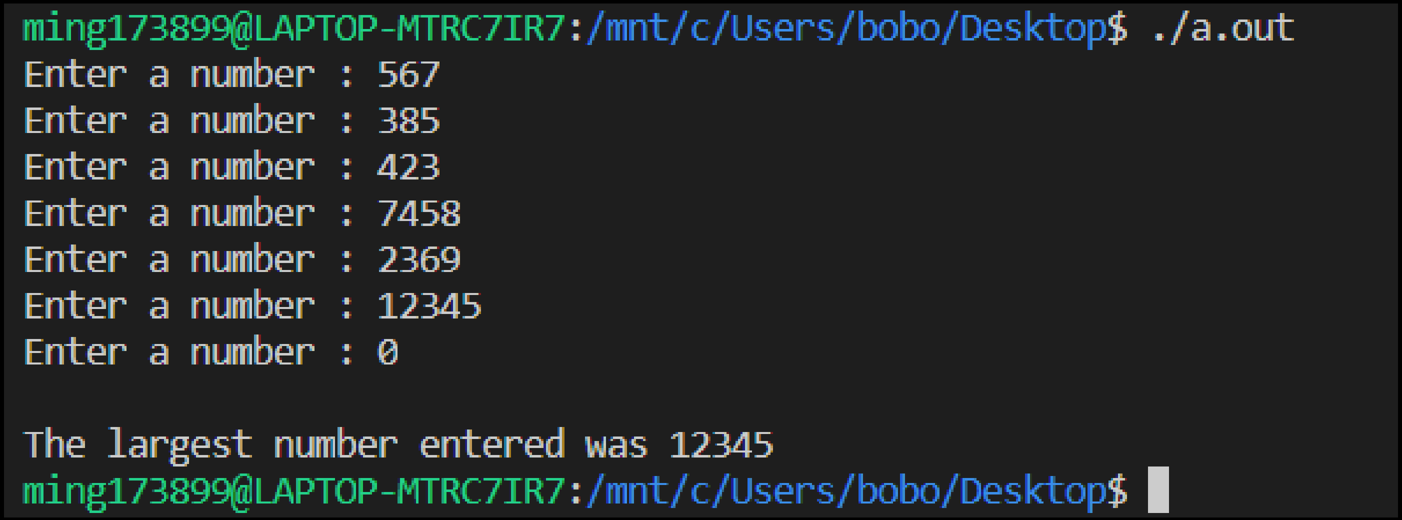
Notice that the numbers aren't necessarily integers.

Solution



```
// find the largest number,
// Breaking program when accept 0 number

#include <stdio.h>

int main(void){

    float x, max = 0.0f;

    for(;;){
        printf("Enter a number : ");
        scanf("%f", &x);

        if(x <= 0.0f)

        if(x > max)

    }

    printf("\n");   /*  blank line  */
    printf("The largest number entered was %g\n", max);

    return 0;
}

```

Example



```
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out
Enter a number : 567
Enter a number : 385
Enter a number : 423
Enter a number : 7458
Enter a number : 2369
Enter a number : 12345
Enter a number : 0

The largest number entered was 12345
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$
```

Chapter 6_project 3

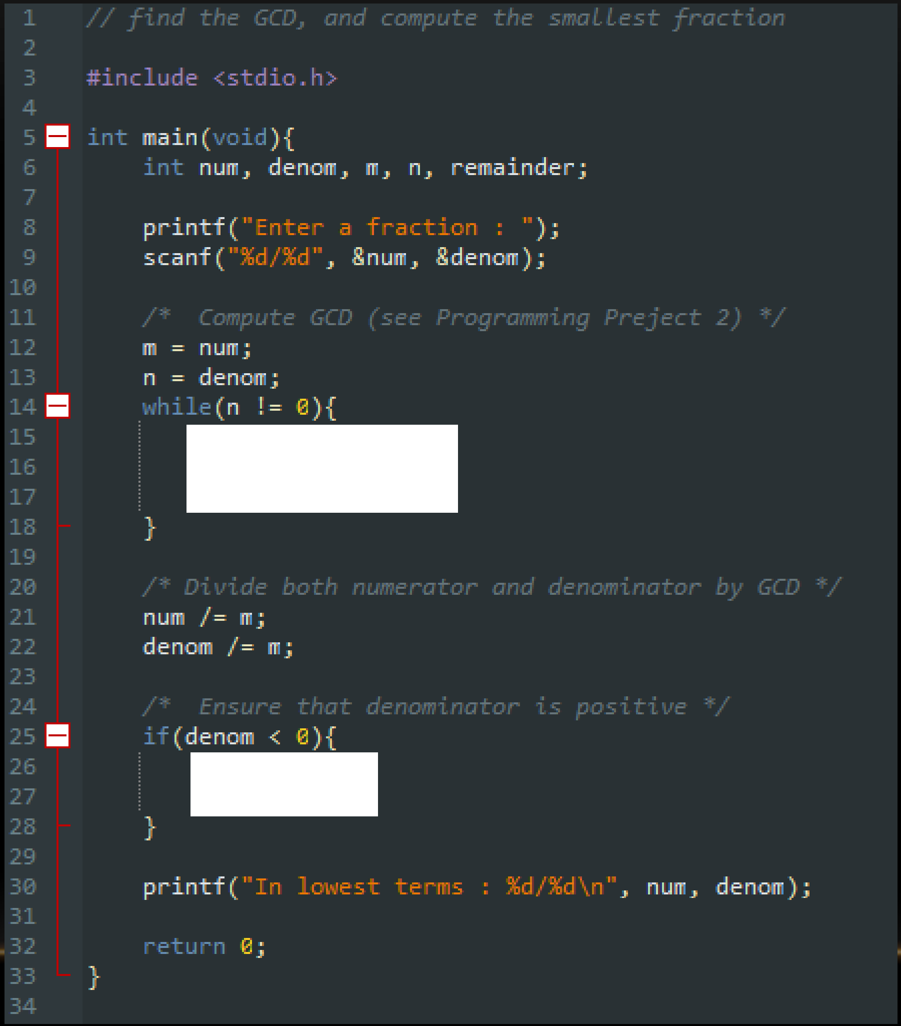
Write a program that asks the user to enter a fraction, then reduces the fraction to lowest terms:

```
Enter a fraction： 6/12

In lowest terms: 1/2
```

Hint: To reduce a fraction to lowest terms, first compute the GCD of the numerator and denominator.

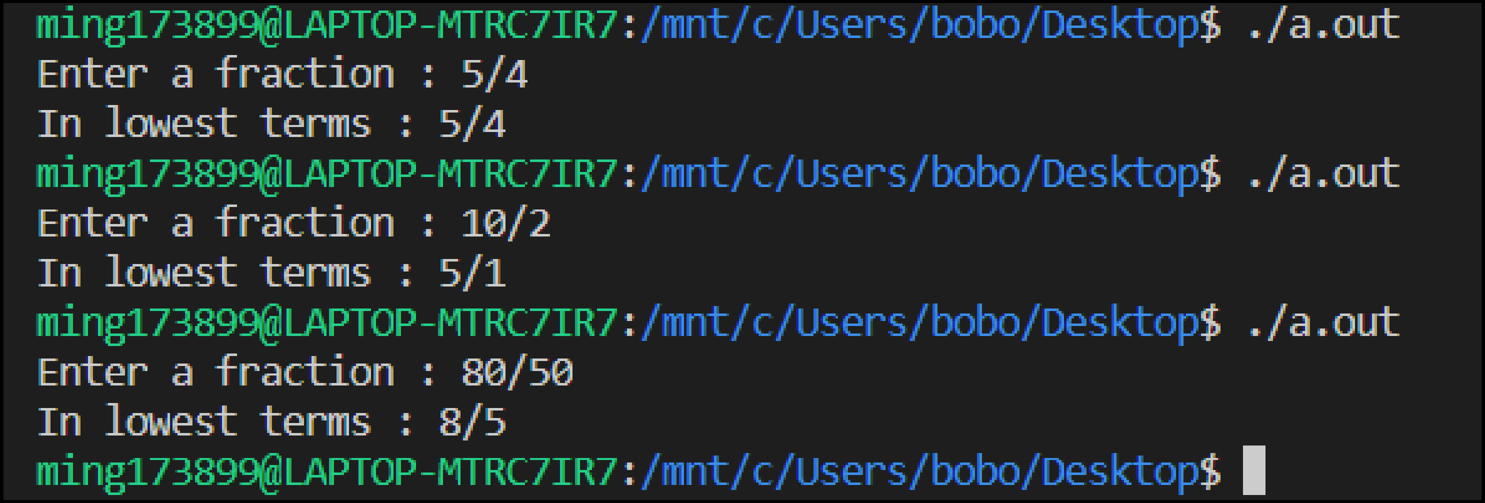
Then divide both the numerator and denominator by the GCD.

Solution



```c
// find the GCD, and compute the smallest fraction

#include <stdio.h>

int main(void){
    int num, denom, m, n, remainder;

    printf("Enter a fraction : ");
    scanf("%d/%d", &num, &denom);

    /*  Compute GCD (see Programming Preject 2) */
    m = num;
    n = denom;
    while(n != 0){



    }

    /* Divide both numerator and denominator by GCD */
    num /= m;
    denom /= m;

    /*  Ensure that denominator is positive */
    if(denom < 0){


    }

    printf("In lowest terms : %d/%d\n", num, denom);

    return 0;
}

```

Example



```
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out
Enter a fraction : 5/4
In lowest terms : 5/4
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out
Enter a fraction : 10/2
In lowest terms : 5/1
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out
Enter a fraction : 80/50
In lowest terms : 8/5
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$
```

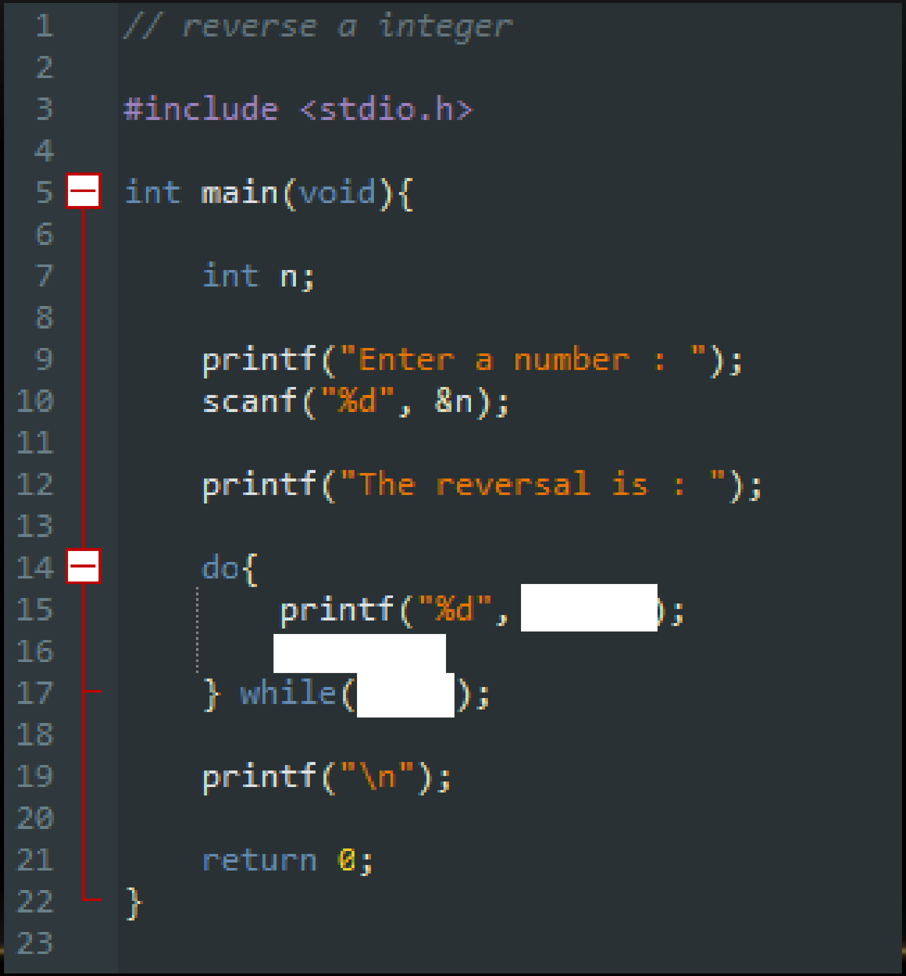
Chapter 6_project 5

Programming Project 1 in Chapter 4 asked you to write a program that displays a two-digit number with its digits reversed.

Generalize the program so that the number can have one, two, three, or more digits.

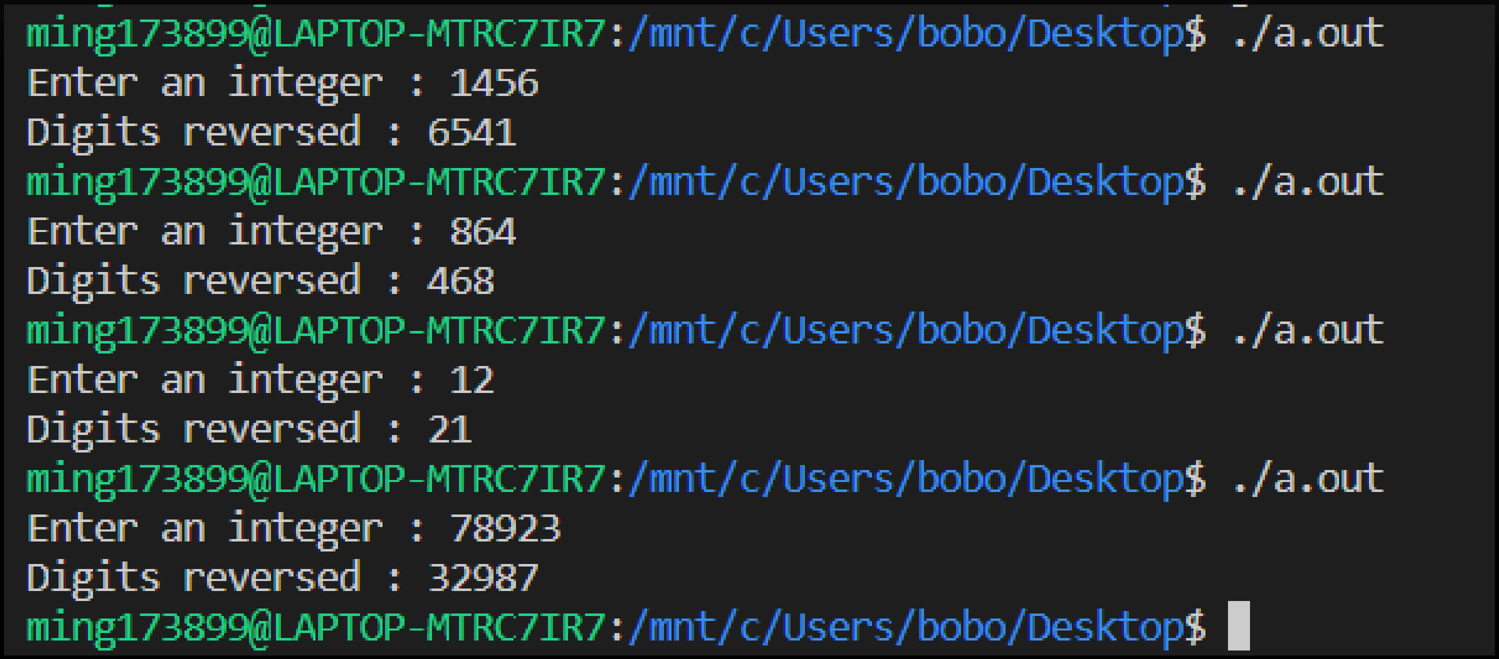
Hint: Use a do loop that repeatedly divides the number by 10, stopping when it reaches 0.

Solution



```c
// reverse a integer

#include <stdio.h>

int main(void){

    int n;

    printf("Enter a number : ");
    scanf("%d", &n);

    printf("The reversal is : ");

    do{
        printf("%d", [redacted]);
    [redacted]
    } while([redacted]);

    printf("\n");

    return 0;
}

```

Example



```
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out
Enter an integer : 1456
Digits reversed : 6541
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out
Enter an integer : 864
Digits reversed : 468
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out
Enter an integer : 12
Digits reversed : 21
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out
Enter an integer : 78923
Digits reversed : 32987
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$
```

Chapter 6_project 11

The value of the mathematical constant e can be expressed as an infinite series:

e = 1+1/1!+1/2!+1/3!+……

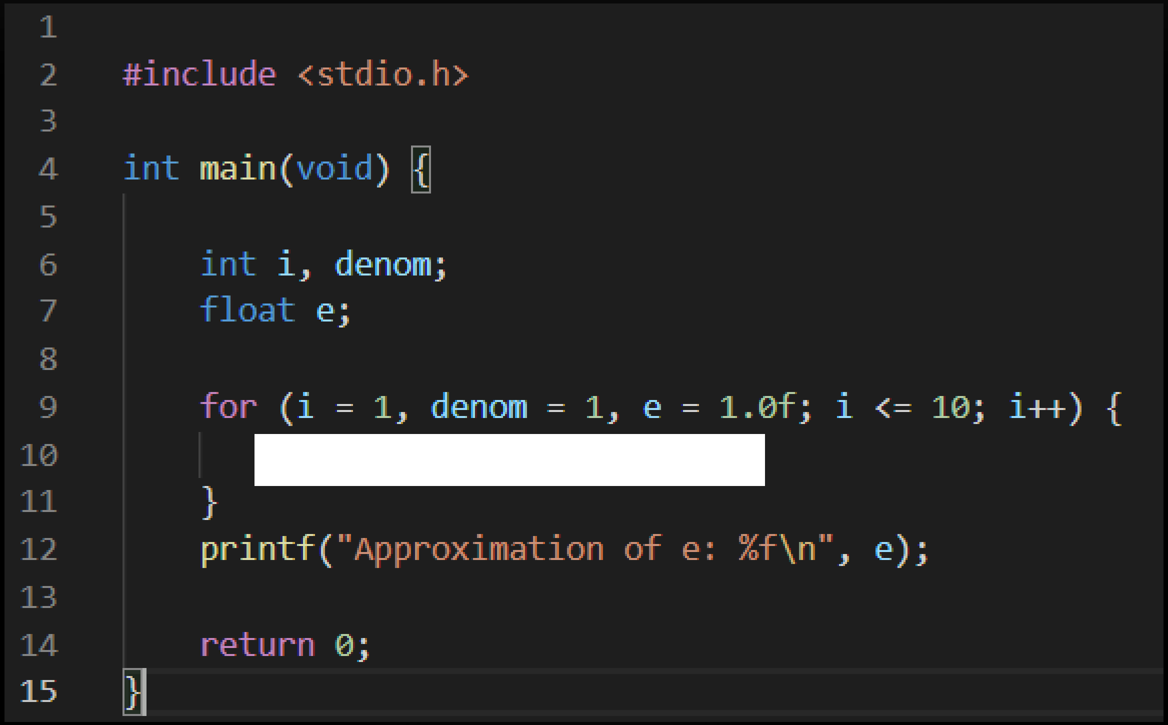
Write a program that approximates e by computing the value of

1+1/1!+1/2!+1/3!+….+1/n!

Where n is an integer entered by the user.

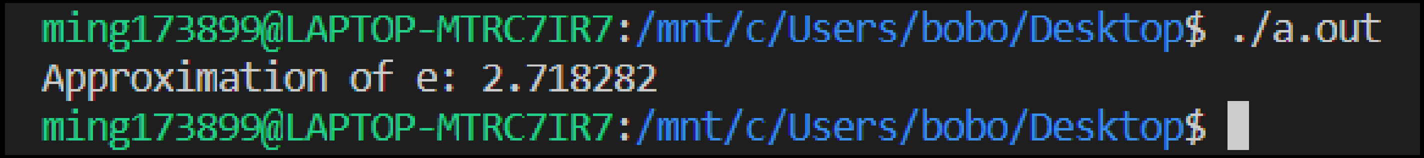
（此題無輸入）

Solution



```c

#include <stdio.h>

int main(void) {

    int i, denom;
    float e;

    for (i = 1, denom = 1, e = 1.0f; i <= 10; i++) {
        [redacted]
    }
    printf("Approximation of e: %f\n", e);

    return 0;
}
```

Example



```
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$ ./a.out
Approximation of e: 2.718282
ming173899@LAPTOP-MTRC7IR7:/mnt/c/Users/bobo/Desktop$
```