

Subject: PRF192- PFC

Workshop 03

Objectives:

- (1) Practicing skills at analyzing and implementing programs using user-defined functions.
- (2) Making familiar with some basic algorithms

Grading 10 programs, 1 mark/program

Program 1:

Objectives	Practice implementing simple functions
Related knowledge	Definition: A prime is positive integer that is greater than 1 and it is the multiple of 1 and itself only. Theorem: The integer n is a prime if and only if $n > 1$ and it can not be divided by all integers from 2 to $\lfloor \sqrt{n} \rfloor$. Use the library math.h to get the function sqrt(double) for getting the square root of a positive number.
Problem	Write a C that will accept a positive integer n, $n \geq 2$ then print out primes between 2 and n.
Analysis <i>Nouns:</i> <i>positive integer</i> $\rightarrow \text{int } n$	Suggested algorithm (logical order of verbs) Begin Do { Accept n; } While (n<2); For (i=2 to n) If (i is a prime) Print out i; → Function int prime (int i) End
Algorithm for checking whether an integer is a prime or not	<pre>int prime(int n) { int m = sqrt(n); /* m: square root of n */ int i; /* variable having value from 2 to m */ if (n<2) return 0; /* Condition 1 is not satisfied */ for (i=2; i<=m; i++) /* checking the second condition */ if (n%i==0) return 0; /* n is divided by i → n is not a prime */ return 1; /* n is a prime */ }</pre>

The screenshot shows the Dev-C++ IDE interface. The main window displays a C program named 'program1.c' which finds prime numbers up to a given limit. The output window shows the program's execution and the prime numbers 2, 3, 5, 7, 11, 13, and 19.

```

C:\Users\USER\Desktop\Cworkshop 03\program1.c - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
[Icons] (globals) | (program1.c) program2.c program3.c program4.c program5.c program6.c program7.c program8.c program9.c program10.c
1 #include<stdio.h>
2 #include<math.h>
3
4 int prime(int n) {
5     int m = sqrt(n);
6     int i;
7     if (n<2) return 0;
8     for(i=2;i<=m;i++)
9         if (n%i==0) return 0;
10    return 1;
11 }
12 int main(){
13     int i,n;
14     printf("Nhap n: ");
15     do
16         scanf("%d",&n);
17         while (n<=2);
18     printf("cac so nguyen to tu mot den n la: ");
19     for(i=2;i<=n;i++){
20         if (prime(i)) printf("%d ",i);
21     }
22 }

```

Nhap n: 20
cac so nguyen to tu mot den n la: 2 3 5 7 11 13 17 19
Process exited after 4.339 seconds with return value 20
Press any key to continue . . .

Compiler Resources Compile Log Debug Find Results Close

Abort Compilation < Line: 20 Col: 39 Sel: 0 Lines: 22 Length: 366 Insert Done parsing in 0.016 seconds

Windows Start Taskbar Icons Ps Pr

100% 352 PM ENG 5/25/2021

#include<stdio.h>
#include<math.h>

```

int prime(int n) {
    int m = sqrt(n);
    int i;
    if (n<2) return 0;
    for(i=2;i<=m;i++)
        if (n%i==0) return 0;
    return 1;
}
int main(){
    int i,n;
    printf("Nhap n: ");
    do
        scanf("%d",&n);
        while (n<=2);
        printf("cac so nguyen to tu mot den n la: ");
        for(i=2;i<=n;i++){
            if (prime(i)) printf("%d ",i);
        }
}

```

Program 2:

Objectives	Practice implementing simple functions
Related knowledge	Leap year (y): $(y \% 400 == 0) \text{ } (y \% 4 == 0 \text{ && } y \% 100 != 0)$
Problem	Write a C program that will accept data of a day then print out whether they are valid or not.
Analysis Data of a day → int d, m, y	Suggested algorithm (logical order of verbs) Begin Accept d, m, y If (valid(d,m,y)) print out "valid date" Else print out "invalid date" End
Algorithm for checking whether a date is valid or not	<pre>int validDate (int d, int m, int y) { int maxd = 31; /*max day of months 1, 3, 5, 7, 8, 10, 12 */ /* basic checking */ if (d<1 d>31 m<1 m>12) return 0; /* update maxd of a month */ if (m==4 m==6 m==9 m==11) maxd=30; else if (m==2) { /* leap year? */ if (y%400==0 (y%4==0 && y%100!=0)) maxd=29; else maxd=28; } return d<=maxd; }</pre>

The screenshot shows the Dev-C++ IDE interface. On the left, the code editor displays the C program. On the right, the terminal window shows the execution of the program, where it prompts for a date and then prints 'valid date'.

```

C:\Users\USER\Desktop\Cworkshop 03\program2.c [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
program1.c program2.c program3.c program4.c program5.c program6.c program7.c program8.c program9.c program10.c
1 #include<stdio.h>
2 #include<math.h>
3
4 int validDate ( int d, int m, int y ) {
5     int maxd = 31;
6     if ( d<1 || d>31 || m<1 || m>12) return 0;
7     if ( m==4 || m==6 || m==9 || m==11) maxd=30;
8     else if (m==2) {
9         if ( y%400==0 || ( y%4==0 && y%100!=0) ) maxd=29;
10        else maxd=28;
11    }
12    return d<=maxd;
13 }
14 int main(){
15     int d,m,y;
16     printf("Nhập vào ngày tháng năm: ");
17     scanf("%d %d %d",&d,&m,&y);
18     if (validDate(d,m,y)) printf("valid date");
19     else printf("invalid date");
20 }

C:\Users\USER\Desktop\Cworkshop 03\program2.exe
Nhập vào ngày tháng năm: 03 12 2002
valid date
-----
Process exited after 6.696 seconds with return value 10
Press any key to continue . . .

```

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

int validDate ( int d, int m, int y ) {
    int maxd = 31;
```

```

if ( d<1 || d>31 || m<1 || m>12) return 0;
if ( m==4 || m==6 || m==9 || m==11) maxd=30;
else if (m==2) {
if ( y%400==0 || ( y%4==0 && y%100!=0)) maxd=29;
else maxd=28;
}
return d<=maxd;
}
int main(){
    int d,m,y;
    printf("Nhập vào ngày tháng năm: ");
    scanf("%d %d %d",&d,&m,&y);
    if (validDate(d,m,y)) printf("valid date");
    else printf("invalid date");
}

```

Program 3:

Objectives	Practice implementing simple functions
Related knowledge	A point p is in a circle if the distance from the center to p is less than the radius.
Problem	Write a C program that will accept a point and a circle having the center is (0,0) then print out the relative position of this point with the circle.
Analysis Nouns: A point → double x,y A circle → double r Relative position → int result → -1: (x,y) is out of the circle → 0: (x,y) is on the circle → 1: (x,y) is in the circle	Suggested algorithm (logical order of verbs) Begin Accept x, y; Do { Accept r; } While(r<0); result = getRelPos(x,y,r); if (result ==1) Print out “The point is in the circle”; else if (result==0) Print out “The point is on the circle”; else Print out “The point is out of the circle”; End
Algorithm for getting relative position of a point with a circle	int getRelPos (double x, double y, double r) { double d2=x*x + y*y; /* $d^2 = x^2 + y^2$ */ double r2= r*r; /* r^2 */ if (d2<r2) return 1 ; /* $d^2 < r^2 \rightarrow$ the point is in the circle */ else if (d2==r2) return 0 ; /* $d^2 = r^2 \rightarrow$ the point is on the circle */ return -1 ; /* $d^2 > r^2 \rightarrow$ the point is out of the circle */ }

The screenshot shows the Dev-C++ IDE interface. The top menu bar includes File, Edit, Search, View, Project, Execute, Tools, AStyle, Window, Help. The toolbar contains various icons for file operations. The title bar says "C:\Users\USER\Desktop\C\workshop 03\program3.c - [Executing] - Dev-C++ 5.11". The status bar at the bottom right shows "3:33 PM 5/25/2021". The code editor window displays the following C code:

```
1 #include<stdio.h>
2
3 int getRelPos ( double x, double y, double r) {
4     double d2=x*x + y*y;
5     double r2= r*r;
6     if (d2<r2) return 1;
7     else if (d2==r2) return 0;
8     return -1;
9 }
10
11 int main() {
12     double x,y,r;
13     printf("Nhap vao toa do diem: ");
14     scanf("%lf %lf",&x,&y);
15     do
16     {
17         printf("Nhap vao ban kinh: ");
18         scanf("%lf",&r);
19     }
20     while (r<0);
21     int result = getRelPos(x,y,r);
22     if (result ==1) printf("The point is in the circle");
23     else if (result==0) printf("The point is on the circle")
24     else printf("The point is out of the circle");
25 }
```

The output window shows the program's execution results:

```
C:\Users\USER\Desktop\C\workshop 03\program3.exe
Nhap vao toa do diem: 3 3
Nhap vao ban kinh: 5
The point is in the circle
-----
Process exited after 5.547 seconds with return value 26
Press any key to continue . . .
```

#include<stdio.h>

int getRelPos (double x, double y, double r) {
 double d2=x*x + y*y;
 double r2= r*r;
 if (d2<r2) return 1;
 else if (d2==r2) return 0;
 return -1;
}

int main() {
 double x,y,r;
 printf("Nhap vao toa do diem: ");
 scanf("%lf %lf",&x,&y);
 do
 {
 printf("Nhap vao ban kinh: ");
 scanf("%lf",&r);
 }
 while (r<0);
 int result = getRelPos(x,y,r);
 if (result ==1) printf("The point is in the circle");
 else if (result==0) printf("The point is on the circle");

```

    else printf("The point is out of the circle");
}

```

Program 4:

Objectives	Practice implementing simple functions
Related knowledge	$n! = 1*2*3*...*n$
Problem	Write a C program that will accept a positive integer then print out its factorial.
Analysis A positive integer → int n	Suggested algorithm (logical order of verbs) Begin Do { Accept n; } While (n<0); Print out factorial(n); End.
Algorithm for Computing factorial of an integer	<pre> double factorial (int n) { double p=1; int i; for (i=2; i<=n; i++) p *= i; return p; } </pre>

The screenshot shows the Dev-C++ IDE interface. On the left, the code for `program4.c` is displayed:

```

1 #include<stdio.h>
2
3 double factorial ( int n ) {
4     double p=1;
5     int i;
6     for (i=2; i<=n; i++) p *= i;
7     return p;
8 }
9
10 int main(){
11     int n;
12     do
13     {
14         printf("Nhập n: ");
15         scanf("%d",&n);
16     }
17     while (n<0);
18     printf("Kết quả là %lf",factorial(n));
19 }

```

On the right, the terminal window shows the execution results:

```

Select C:\Users\USER\Desktop\C\workshop_03\program4.exe
Nhập n: 5
Kết quả là 120.000000
Process exited after 0.9115 seconds with return value 21
Press any key to continue . . .

```

#include<stdio.h>

double factorial (int n) {
 double p=1;

```

int i;
for (i=2; i<=n; i++) p *= i;
return p;
}

int main(){
    int n;
    do
    {
        printf("Nhập n: ");
        scanf("%d",&n);
    }
    while (n<0);
    printf("Kết quả là %lf",factorial(n));
}

```

Program 5:

Objectives	Practice implementing simple functions
Related knowledge	Fibonacci sequence: 1 1 2 3 5 8 13 21 34 ... Two first numbers: 1 Others: Its value is the sum of 2 previous numbers
Problem	Write a C program that will print out the value at the n^{th} position in Fibonacci sequence.
Analysis A position → int n	Suggested algorithm (logical order of verbs) Begin Do { Accept n; } While (n<1); Print out fibo(n); End.
Algorithm for Computing the n^{th} value of the Fibonacci sequence	double fibo (int n) { int t1=1, t2=1, f=1, i ; for (i= 3, i<=n; i++) { f= t1 + t2; t1= t2; t2=f; } return f; }

How to compute the nth value of the Fibonacci sequence

Position 1	2	3	4	5	6	7	8	9	10
1	1	2	3	5	8	13	21	34	55
T1	T2	F							

	T1	T2	F						
	T1	T2	F						
		T1	T2	F					
			T1	T2	F				
				T1	T2	F			
					T1	T2	F		
						T1	T2	F	...

```

1 #include<stdio.h>
2
3 double fibo ( int n) {
4     int t1=1, t2=1, f=1, i;
5     for ( i=3;i<=n;i++) {
6         f= t1 + t2;
7         t1= t2;
8         t2=f;
9     }
10    return f;
11 }
12
13 int main(){
14     int n;
15     do
16     {
17         printf("Nhập n: ");
18         scanf("%d",&n);
19     }
20     while (n<1);
21     printf("Kết quả là %lf",fibo(n));
22 }

```

#include<stdio.h>

```

double fibo ( int n) {
    int t1=1, t2=1, f=1, i;
    for ( i=3;i<=n;i++) {
        f= t1 + t2;
        t1= t2;
        t2=f;
    }
    return f;
}

```

```

int main(){
    int n;
    do
    {
        printf("Nhập n: ");
        scanf("%d",&n);

```

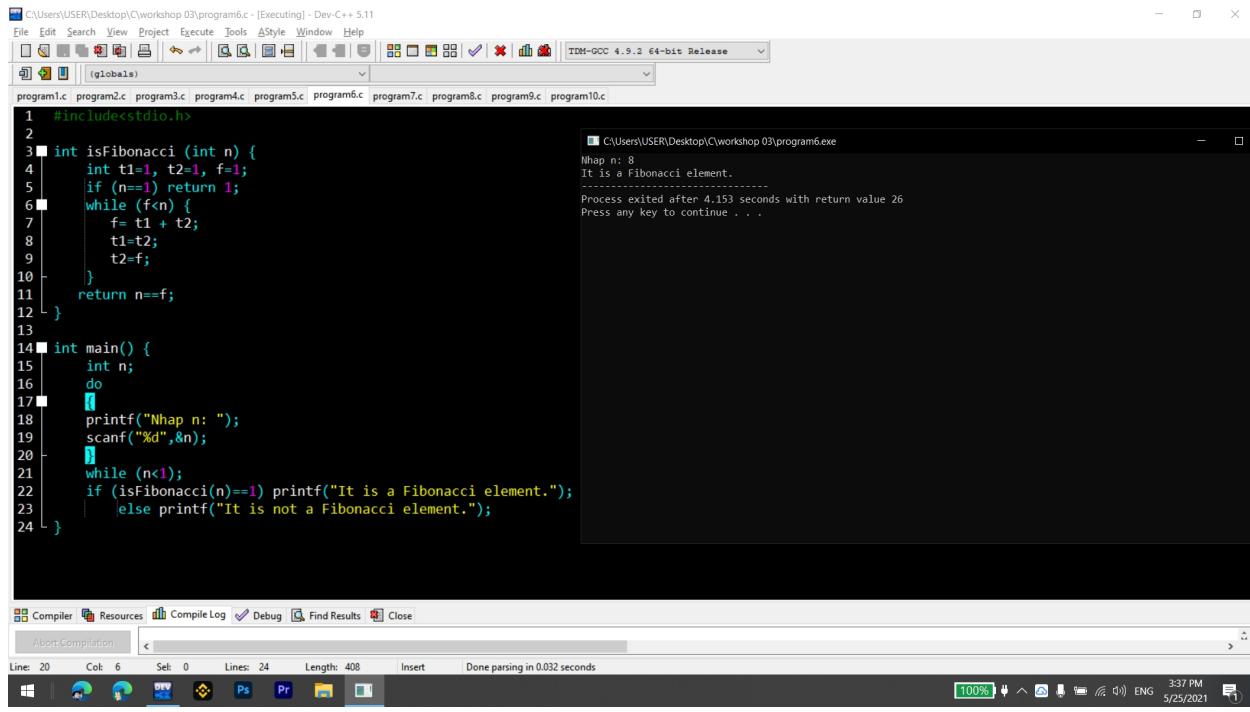
```

    }
    while (n<1);
    printf("Ket qua la %lf",fib(n));
}

```

Program 6:

Objectives	Practice implementing simple functions
Related knowledge	
Problem	Write a C program that will accept a positive integer then print out whether it is an element of the Fibonacci sequence or not.
Analysis An integer → int n	<p>Suggested algorithm (logical order of verbs)</p> <p>Begin</p> <p> Do {</p> <p> Accept n;</p> <p> }</p> <p> While (n<1);</p> <p> If (isFibonacci(n)==1) Print out "It is a Fibonacci element.";</p> <p> Else print out "It is not a Fibonacci element."</p> <p>End</p>
Algorithm for Checking whether an integer is a element of the Fibonacci sequence or not	<pre> int isFibonacci (int n) { int t1=1, t2=1, f=1; if (n==1) return 1; /* n belongs to the Fibonacci sequence*/ while (f<n) /* Find out the Fibo number f to n */ { f= t1 + t2; t1=t2; t2=f; } return n==f; /* if n==f → n is Fibo element → return 1 */ } </pre>



```
C:\Users\USER\Desktop\C\workshop 03\program6.c - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
program1.c program2.c program3.c program4.c program5.c program6.c program7.c program8.c program9.c program10.c
1 #include<stdio.h>
2
3 int isFibonacci (int n) {
4     int t1=1, t2=1, f=1;
5     if (n==1) return 1;
6     while (f<n) {
7         f= t1 + t2;
8         t1=t2;
9         t2=f;
10    }
11    return n==f;
12 }
13
14 int main() {
15     int n;
16     do
17     {
18         printf("Nhập n: ");
19         scanf("%d",&n);
20     }
21     while (n<1);
22     if (isFibonacci(n)==1) printf("It is a Fibonacci element.");
23     else printf("It is not a Fibonacci element.");
24 }
```

#include<stdio.h>

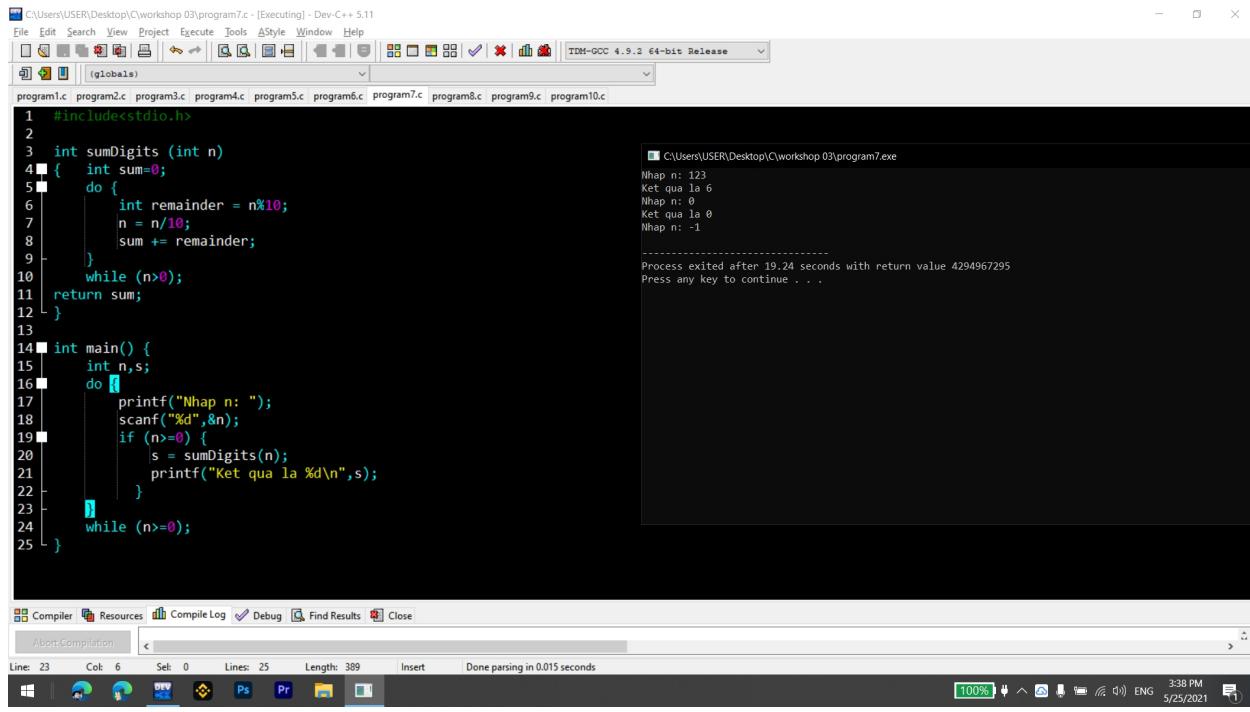
```
int isFibonacci (int n) {
    int t1=1, t2=1, f=1;
    if (n==1) return 1;
    while (f<n) {
        f= t1 + t2;
        t1=t2;
        t2=f;
    }
    return n==f;
}
```

```
int main() {
    int n;
    do
    {
        printf("Nhập n: ");
        scanf("%d",&n);
    }
    while (n<1);
    if (isFibonacci(n)==1) printf("It is a Fibonacci element.");
    else printf("It is not a Fibonacci element.");
```

{

Program 7:

Objectives	Practice implementing simple functions
Related knowledge	Getting the rightmost digit of the integer n: n%10
Problem	Write a C program that will carry out some times. In each time, a nonnegative integer is accepted then print out the sum of its decimal digits. The program will terminate when its value of accepted number is negative.
Analysis Sum → int S=0 Accepted integer → int n	Suggested algorithm (logical order of verbs) Begin Do { Accept n; If (n>=0) { S = sumDigits(n); Print out S; } } While (n>=0); End
Algorithm for Computing sum of digits of a nonnegative integer	int sumDigits (int n) { int sum=0; /* initialize sum of digits */ Do { int remainder = n%10 ; /* Get a digit at unit position */ n = n/10; sum += remainder; } while (n>0); return sum; }



```
1 #include<stdio.h>
2
3 int sumDigits (int n)
4 { int sum=0;
5 do {
6     int remainder = n%10;
7     n = n/10;
8     sum += remainder;
9 }
10 while (n>0);
11 return sum;
12 }
13
14 int main() {
15     int n,s;
16     do {
17         printf("Nhập n: ");
18         scanf("%d",&n);
19         if (n>0) {
20             s = sumDigits(n);
21             printf("Kết quả là %d\n",s);
22         }
23     while (n>=0);
24 }
```

#include<stdio.h>

```
int sumDigits (int n)
{ int sum=0;
    do {
        int remainder = n%10;
        n = n/10;
        sum += remainder;
    }
    while (n>0);
return sum;
}
```

```
int main() {
    int n,s;
    do {
        printf("Nhập n: ");
        scanf("%d",&n);
        if (n>=0) {
            s = sumDigits(n);
            printf("Kết quả là %d\n",s);
        }
    }
```

```
while (n>=0);
}
```

Program 8:

Objectives	Practice implementing simple functions
Related knowledge	<p>Making a real number from its integral part and its fraction (its fraction must be positive).</p> <p>Example : 32 25 → 32.25 $25 \rightarrow 0.25 \rightarrow 32+0.25 = 32.25$</p> <p>Example -51 139 → -51.139 $139 \rightarrow 0.139 \rightarrow -51-0.139 = -51.139$</p> <pre>double makeDouble(int ipart, int fraction) { double d_f= fraction; while (d_f >=1) d_f = d_f/10; /* create the fraction <1 */ if (ipart<0) return ipart - d_f; /* case -51 - 0.139 */ return ipart + d_f; /* case 32 + 0.25 */ }</pre>
Problem	Write a C program that will accept the integral part and fraction of a real number then print out the this real number.
Analysis Integral part \rightarrow int ipart Fraction \rightarrow int fraction Real number \rightarrow double value	<p>Suggested algorithm (logical order of verbs)</p> <pre>Begin Accept ipart; Do { Accept fraction; } While fraction<0; value= makeDouble(ipart,fraction); Print out value; End</pre>

```

C:\Users\USER\Desktop\C\workshop 03\program8.c - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
program1.c program2.c program3.c program4.c program5.c program6.c program7.c program8.c program9.c program10.c
1 #include<stdio.h>
2
3 double makeDouble(int ipart, int fraction)
4 { double d_f= fraction;
5   while (d_f >=1) d_f = d_f/10;
6   if (ipart<0) return (ipart - d_f);
7   return ipart + d_f ;
8 }
9
10 int main() {
11   int ipart,fraction;
12   printf("Nhập vào phần nguyên: ");
13   scanf("%d",&ipart);
14   do
15   {
16     printf("Nhập vào phần thập phân: ");
17     scanf("%d",&fraction);
18   } while (fraction<0);
19   double value = makeDouble(ipart,fraction);
20   printf("%lf",value);
21 }
22

```

Nhập vào phần nguyên: 12
Nhập vào phần thập phân: 56
12.560000

Process exited after 3.437 seconds with return value 9
Press any key to continue . . .

[Compiler](#) [Resources](#) [Compile Log](#) [Debug](#) [Find Results](#) [Close](#)

Abort Compilation < Line: 16 Col: 45 Sel: 0 Lines: 22 Length: 478 Insert Done parsing in 0 seconds

100% ENG 3:43 PM 5/25/2021

#include<stdio.h>

double makeDouble(int ipart, int fraction)
{ double d f= fraction;
while (d f >=1) d f = d f/10;
if (ipart<0) return (ipart - d f);
return ipart + d f ;
}

int main() {
int ipart,fraction;
printf("Nhập vào phần nguyên: ");
scanf("%d",&ipart);
do
{
printf("Nhập vào phần thập phân: ");
scanf("%d",&fraction);
}
while (fraction<0);
double value = makeDouble(ipart,fraction);
printf("%lf",value);
}

Program 9:

Objectives	Practice implementing simple functions																												
Related knowledge	<p>Find out the greatest common divisor (gcd) and least common multiple (lcm) of two positive integers:</p> <p><i>Find out gcd of a and b</i></p> <table style="margin-left: auto; margin-right: auto;"> <tr> <th style="text-align: center;">a</th> <th style="text-align: center;">b</th> <th style="text-align: center;">a</th> <th style="text-align: center;">b</th> </tr> <tr> <td style="text-align: center;">14</td> <td style="text-align: center;">21</td> <td style="text-align: center;">13</td> <td style="text-align: center;">8</td> </tr> <tr> <td style="text-align: center;">14</td> <td style="text-align: center;">7</td> <td style="text-align: center;">5</td> <td style="text-align: center;">8</td> </tr> <tr> <td style="text-align: center;">7</td> <td style="text-align: center;">7</td> <td style="text-align: center;">5</td> <td style="text-align: center;">3</td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">2</td> <td style="text-align: center;">3</td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">2</td> <td style="text-align: center;">1</td> </tr> <tr> <td></td> <td></td> <td style="text-align: center;">1</td> <td style="text-align: center;">1</td> </tr> </table> <pre> int gcd(int a, int b) { while (a != b) if a>b then a -=b; else b -= a; return a; } int lcm (int a, int b) { return a*b/ gcd(a,b); } </pre>	a	b	a	b	14	21	13	8	14	7	5	8	7	7	5	3			2	3			2	1			1	1
a	b	a	b																										
14	21	13	8																										
14	7	5	8																										
7	7	5	3																										
		2	3																										
		2	1																										
		1	1																										
Problem	Write a C program that will accept two positive integers then print out their greatest common divisor and least common multiple.																												
Analysis Two integers → int a, b gcd → int d lcm → int m	Suggested algorithm (logical order of verbs) Begin Do { Accept a, b; } While (a<=0 OR b <=0); d = gcd(a,b); m = lcm (a.b); Print out d; Print out m; End																												

```

C:\Users\USER\Desktop\C\workshop 03\program9.c - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
program1.c program2.c program3.c program4.c program5.c program6.c program7.c program8.c program9.c program10.c
1 #include<stdio.h>
2
3 int gcd( int a, int b )
4 {
5     while ( a != b )
6     {
7         if (a>b) a -=b;
8         else b -= a;
9     }
10    return a;
11 }
12
13 int lcm ( int a, int b ) {
14     return a*b/ gcd(a,b);
15 }
16
17 int main() {
18     int a,b;
19     do
20     {
21         printf("Nhập hai số a và b: ");
22         scanf("%d%d",&a,&b);
23     }
24     while ( a<=0 || b <=0 );
25     printf("gcd = %d, lcm = %d",gcd(a,b),lcm(a,b));
26 }

```

Nhap hai so a va b: 12 30
gcd = 6, lcm = 60
Process exited after 2.325 seconds with return value 17
Press any key to continue . . .

#include<stdio.h>

```

int gcd( int a, int b)
{ while ( a != b )
    if (a>b) a -=b;
    else b -= a;
return a;
}
int lcm ( int a, int b) {
    return a*b/ gcd(a,b);
}

```

```

int main() {
    int a,b;
    do
    {
        printf("Nhập hai số a và b: ");
        scanf("%d%d",&a,&b);
    }
    while ( a<=0 || b <=0 );
    printf("gcd = %d, lcm = %d",gcd(a,b),lcm(a,b));
}

```

{

Program 10:

Objectives	Practice implementing simple functions
Related knowledge	<p>Print out the minimum and the maximum digits of a nonnegative integer integer</p> <p>Example: n= 10293 → Print out 9, 0</p> <pre>void printMinMaxDigits(int n) { int digit; /* Variable for extracting 1 digit */ int min, max ; /* Result variables */ digit = n% 10; /* get the first rightmost digit: 3 */ n=n/10; /* 1029, the remainder needs to proceed after*/ min=max=remainder; /* initialize results */ while (n>0) { digit = n%10; /* Get the next digit */ n=n/10; if (min > remainder) min=remainder; /* update results */ if (max < remainder) max=remainder; } Print out min, max; }</pre>
Problem	Write a C program that will accept a non-negative integer then print out its minimum and maximum digits.
Analysis Noun: A integer → int n	<p>Suggested algorithm (logical order of verbs)</p> <pre>Begin Do { Accept n; printMinMaxDigits(n); } While (n<0); End</pre>

The screenshot shows the Dev-C++ IDE interface. The main window displays a C program named 'program10.c'. The code defines a function 'printMinMaxDigits' that takes an integer 'n' and prints its minimum and maximum digits. It uses a while loop to extract digits from 'n' and update the min and max variables. The main function reads input from the user and calls the helper function. The status bar at the bottom shows the current line (Line: 23), column (Col: 20), selected text (Sel: 0), lines (Lines: 27), length (Length: 427), and the time (3:50 PM, 5/25/2021). The output window shows the program's execution results: 'Nhap n: 123459', 'min = 1, max = 9', and 'Process exited after 4.399 seconds with return value 123459'. A message 'Press any key to continue . . .' is displayed.

```
1 #include<stdio.h>
2
3 void printMinMaxDigits( int n)
4 {
5     int t,min, max;
6     t= n%10;
7     n=n/10;
8     min = t; max = t;
9     while (n>0)
10    {
11        t = n%10;
12        n=n/10;
13        if (min > t) min=t;
14        if (max < t) max=t;
15    }
16    printf("min = %d, max = %d\n",min, max);
17 }
18 int main() {
19     int n;
20     do
21     {
22         printf("Nhap n: ");
23         scanf("%d",&n);
24         printMinMaxDigits(n);
25     }
26     while (n<0);
27 }
```

```
#include<stdio.h>

void printMinMaxDigits( int n)
{
    int t,min, max;
    t= n%10;
    n=n/10;
    min = t; max = t;
    while (n>0)
    {
        t = n%10;
        n=n/10;
        if (min > t) min=t;
        if (max < t) max=t;
    }
    printf("min = %d, max = %d\n",min, max);
}

int main() {
    int n;
    do
    {
        printf("Nhap n: ");
        scanf("%d",&n);
        printMinMaxDigits(n);
    }
    while (n<0);
}
```

