



7. Truyền tham số cho hàm

a. Truyền tham trị

- **Giá trị** của các tham số thực được truyền cho các tham số hình thức tương ứng của hàm.
- Mọi thay đổi trong hàm trên các tham số hình thức sẽ không ảnh hưởng tới giá trị ban đầu của của các tham số thực.



Chương 5: Hàm



Ví dụ:

//Hàm hoán vị hai số nguyên

```
void HoanVi(int a, int b) {  
    int tam; //khai báo biến cục bộ tam  
    tam = a; a = b; b = tam;  
}
```

```
void main() {  
    int x = 3, y = 7;  
    HoanVi(x, y); /*Lời gọi hàm HoanVi với hai tham số  
                  thực x và y*/  
    <In x, y>; //x và y không thay đổi  
}
```



b. Truyền tham trở

- **Địa chỉ** của các tham số thực (có dấu & đặt trước) được truyền cho các tham số hình thức tương ứng của hàm và các tham số hình thức phải được khai báo dưới dạng **con trở** (có dấu * đặt trước).
- Bên trong hàm có thể truy nhập trực tiếp đến vùng nhớ các tham số thực thông qua các con trở này.



Chương 5: Hàm

Ví dụ:

```
/*Hàm hoán vị hai số nguyên */
```

```
void HoanVi(int *a, int *b) {  
    int tam;  
    tam = *a; *a = *b; *b = tam;  
}
```

```
void main() {  
    int x = 3, y = 7;  
    HoanVi(&x, &y); /*Lời gọi hàm HoanVi() với hai  
                    tham số thực &x và &y*/  
    <In x, y>; //x và y đã bị thay đổi  
}
```



8. Nguyên mẫu hàm

- Về nguyên tắc khi gọi một hàm thì hàm đó phải được khai báo trước, nếu không chương trình sẽ báo lỗi. Tuy nhiên cũng có thể gọi một hàm chưa được khai báo trước bằng cách khai báo trước nguyên mẫu hàm.
- Nguyên mẫu hàm thực chất là dòng đầu của hàm có thêm dấu chấm phẩy.



Chương 5: Hàm

Ví dụ:

```
void HoanVi(int *a, int *b); /*Khai báo nguyên  
mẫu hàm*/
```

```
void main() {  
    int x = 3, y = 7;  
    HoanVi(&x, &y); /*Gọi hàm HoanVi() được khai báo  
sau*/  
    <In x, y>  
}  
void HoanVi(int *a, int *b) {  
    int tam;  
    tam = *a; *a = *b; *b = tam;  
}
```

```
26
27     int n;
28
29     do
30     {
31         printf("\nNhap vao so nguyen duong (n > 0): ");
32         scanf("%d", &n);
33     } while (n <= 0);
34
35     /* 3.4.3.5.10. Hãy đếm số lượng chữ số của số nguyên dương n giống như chữ số đầu tiên
36     của số nguyên dương n.
37     Ví dụ: Nhập n = 10151. In ra màn hình: Có ba chữ số 1. */
38
39     // B1: Tìm k
40     int k = 1;
41     while (k <= n)
42     {
43         k *= 10;
44     }
45
46     k = k / 10;
47     printf("\nk = %d", k);
48
49     // B2: Dem
50     int So = n, Dem = 0;
51     int ChuSoDauTien = n / k;
52
53     while (So > 0)
54     {
55         int x = So % 10;
56
57         if (x == ChuSoDauTien)
58             Dem++;
59
60         So /= 10;
61     }
62
63     printf("\nCo %d chu so %d", Dem, ChuSoDauTien);
64
```

```
10
11     for (int i = 2; i < n; i++)
12     {
13         if (n % i == 0)
14         {
15             KT = 0;
16             break;
17         }
18     }
19
20     return KT;
21 }
```

```
22
23 void TangGiaTri(int &n)
24 {
25     n = n + 10;
26 }
```

*ví dụ tham số
và tham trị*

```
27
28 void main()
29 {
30     srand(unsigned int(time(NULL)));
31
32     int n = 10;
33
34     TangGiaTri(n);
35
36     printf("\nSau khi gọi hàm: n = %d", n);
37
38
39     /*do
40     {
41         printf("\nNhập vào số nguyên dương (n > 0): ");
42         scanf("%d", &n);
43     } while (n <= 0);*/
44 }
```



```
46     int x = So % 10;
47
48     if (x % 2 == 0)
49     {
50         KT = 0;
51         break;
52     }
53
54     So /= 10;
55 }
56
57 if (KT == 1)
58     printf("\n%d toan le", n);
59 else
60     printf("\n%d khong toan le", n);
61
62 // printf("\nCo %d chu so %d", Dem, ChuSoDauTien);
63
64
65 // Tich = 1, n = 6
66 // 1: tich = 1 x 2 = 2, 2 <= 6
67 // 2: tich = 2 x 2 = 4, 4 <= 6
68 // 3: tich = 4 x 2 = 8, 8 > 6
69 // tich = tich / 2 = 4, 4 != 6 => 6 khong phai 2k
70
71 // tich = 1, n = 8
72 // 1: tich = 2, 2 <= 8
73 // 2: tich = 4, 4 <= 8
74 // 3: tich = 8, 8 <= 8
75 // 4: tich = 16, 16 > 8
76 // tich = tich / 2 = 8, 8 == 8, 8 => 2k
77
78 }
79
```

```
1  #include<stdio.h>
2
3  int nhapso() {
4      unsigned int n;
5      do {
6          printf("moi nhap so nguyen duong: ");
7          scanf("%u", &n);
8      } while (n <= 0);
9      return (n);
10 }
11
12 int ktra2muk (unsigned int n) {
13     unsigned int tich=1, i =1, check = 0;
14
15     while (tich <= n) {
16         tich *= 2;
17     }
18     // printf("%u\n", tich);
19
20     if (tich / 2 == n) {
21         check = 1;
22     }
23     // printf("So nhap n: %u\n", n);
24     // printf("%u\n", check);
25     return check;
26 }
27
28
29 void main() {
30     unsigned int n, check = 0;
31     n = nhapso();
32     check = ktra2muk(n);
33
34     if (check == 1) {
35         printf("%u la so 2 mu k",n);
36     }
37     else {
38         printf("%u KHONG la so 2 mu k",n);
39     }
40 }
```

```
1 // 3.4.3.6trg27
2 #include<stdio.h>
3
4 int timUSCLN(unsigned int a, unsigned int b) {
5     unsigned int x = a;
6     unsigned int y = b;
7
8     while (x != y)
9     {
10         if (x > y) {
11             x = x - y;
12             //printf("%d\n", x);
13         }
14         else {
15             y = y - x;
16             //printf("%d\n", y);
17         }
18     }
19     return x;
20 }
21
22 int nhapso() {
23     unsigned int n;
24     do {
25         printf("moi nhap so nguyen duong > 0: ");
26         scanf("%u", &n);
27     } while (n <= 0);
28     return (n);
29 }
30
31 void main() {
32     unsigned int a, b, c;
33     a = nhapso();
34     b = nhapso();
35     c = timUSCLN(a, b);
36     printf("USCLN cua %u va %u : %u", a, b, c);
37 }
```

```
49     {
50         KT = 0;
51         break;
52     }
53
54     So /= 10;
55 }
56
57 if (KT == 1)
58     printf("\n%d toan le", n);
59 else
60     printf("\n%d khong toan le", n);
61
62 // printf("\nCo %d chu so %d", Dem, ChuSoDauTien);
63
64
65 // n = 12345 -> So Nghich Dao: 54321
66 // n = 12321 -> so Nghich Dao: 12321
67
68 // tong = 0
69 // b1: x = 5, tong = 0 * 10 + 5 = 5
70 // b2: x = 4, tong = 5 * 10 + 4 = 54
71 // b3: x = 3, tong = 54 * 10 + 3 = 543
72
73 /* 3.4.3.5.16. Hãy kiểm tra các chữ của số nguyên dương n có tăng dần từ trái sang phải
74 hay không?
75 Ví dụ: n = 12345 -> tăng dần từ trái -> phải, n = 12321 -> không tăng dần từ trái qua phải*/
76
77 }
78
```

```
1  #include<stdio.h>
2
3  int nhapso() {
4      unsigned int n;
5      do {
6          printf("moi nhap so nguyen duong: ");
7          scanf("%u", &n);
8      } while (n <= 0);
9      return (n);
10 }
11
12 unsigned int tongDonSo(unsigned int n) {
13     unsigned int a = 0, tong = 0;
14     while (n > 0) {
15         a = n % 10;
16         tong = tong * 10 + a;
17         n /= 10;
18     }
19     return tong;
20 }
21
22 void main() {
23     unsigned int n, s=0;
24     n = nhapso();
25     s = tongDonSo(n);
26     printf("So dao: %u", s);
27 }
```

```
1      #include<stdio.h>
2
3      unsigned int nhapsoND() {
4          unsigned int n;
5          do {
6              printf("moi nhap so nguyen duong > 0: ");
7              scanf("%u", &n);
8          } while (n <= 0);
9          return (n);
10     }
11
12     bool ktraSNT(unsigned int n) {
13         int ktra = 1; // 1 la SNT
14         for (int i = 2; i < n; i++) {
15             if (n % i == 0) {
16                 ktra = 0;
17                 break;
18             }
19         }
20
21         if (ktra == 1 && n > 1) {
22             return true;
23         }
24         else {
25             return false;
26         }
27     }
28
29     void main() {
30         unsigned int n;
31         n = nhapsoND();
32         for (unsigned int i = n; i > 0; i--) {
33             if (ktraSNT(i)) {
34                 printf("So nguyen to gan %u nhat la %u", n, i);
35                 break;
36             }
37         }
38     }
```

```
1  #include <stdio.h>
2  #include <math.h>
3
4  // 3.4.3.5.1
5  void main() {
6      unsigned int n, tmp=0, tmp1=0, i=0, k=0;
7      printf("vui long nhap so nguyen bat ky: ");
8      scanf("%d", &n);
9      tmp = n;
10
11     while (tmp > 0) {
12         k += 1;
13         tmp /= 10;
14     }
15
16     tmp1 = pow(10, k-1);
17
18     int sL = n / tmp1;
19     printf("\nSo hang %d: %d", tmp1, sL);
20     n %= tmp1;
21     do {
22         sL = n / (tmp1*=0.1);
23         printf("\nSo hang %d: %d", tmp1, sL);
24         n %= tmp1;
25     } while (n > 0);
26 }
27
```

```
1  #include <stdio.h>
2
3  bool ktraSNT(unsigned int n) {
4      int kt = 1; // SNT
5      for (int i = 2; i < n; i++) {
6          if (n % i == 0) {
7              kt = 0;
8              break;
9          }
10     }
11
12     if (kt == 1 && n > 1) {
13         return true;
14     }
15     else {
16         return false;
17     }
18 }
19
20 unsigned int nhapsoND() {
21     unsigned int n;
22     do {
23         printf("Vui long nhap so nguyen duong > 0: ");
24         scanf("%u", &n);
25     } while (n <= 0);
26
27     return n;
28 }
29
```



```
19
20 unsigned int nhapsoND() {
21     unsigned int n;
22     do {
23         printf("Vui long nhap so nguyen duong > 0: ");
24         scanf("%u", &n);
25     } while (n <= 0);
26
27     return n;
28 }
29
30 void main() {
31     unsigned int n;
32     n = nhapsoND();
33
34     // cach 1
35     //for (unsigned int i = n+1;; i++) {
36     //    if (ktraSNT(i)) {
37     //        printf("---> %u la nguyen to lon hon va gan voi %u nhat", i, n);
38     //        break;
39     //    }
40     //}
41
42     // cach 2:
43     unsigned int i = n;
44     do {
45         i++;
46     } while (ktraSNT(i) == false);
47     printf("---> %u la nguyen to lon hon va gan voi %u nhat", i, n);
48 }
```



7. Truyền tham số cho hàm

a. Truyền tham trị

- **Giá trị** của các tham số thực được truyền cho các tham số hình thức tương ứng của hàm.
- Mọi thay đổi trong hàm trên các tham số hình thức sẽ không ảnh hưởng tới giá trị ban đầu của của các tham số thực.



Chương 5: Hàm



Ví dụ:

//Hàm hoán vị hai số nguyên

```
void HoanVi(int a, int b) {  
    int tam; //khai báo biến cục bộ tam  
    tam = a; a = b; b = tam;  
}
```

```
void main() {  
    int x = 3, y = 7;  
    HoanVi(x, y); /*Lời gọi hàm HoanVi với hai tham số  
                  thực x và y*/  
    <In x, y>; //x và y không thay đổi  
}
```



b. Truyền tham trở

- **Địa chỉ** của các tham số thực (có dấu & đặt trước) được truyền cho các tham số hình thức tương ứng của hàm và các tham số hình thức phải được khai báo dưới dạng **con trở** (có dấu * đặt trước).
- Bên trong hàm có thể truy nhập trực tiếp đến vùng nhớ các tham số thực thông qua các con trở này.



Chương 5: Hàm

Ví dụ:

```
/*Hàm hoán vị hai số nguyên */
```

```
void HoanVi(int *a, int *b) {  
    int tam;  
    tam = *a; *a = *b; *b = tam;  
}
```

```
void main() {  
    int x = 3, y = 7;  
    HoanVi(&x, &y); /*Lời gọi hàm HoanVi() với hai  
                    tham số thực &x và &y*/  
    <In x, y>; //x và y đã bị thay đổi  
}
```



8. Nguyên mẫu hàm

- Về nguyên tắc khi gọi một hàm thì hàm đó phải được khai báo trước, nếu không chương trình sẽ báo lỗi. Tuy nhiên cũng có thể gọi một hàm chưa được khai báo trước bằng cách khai báo trước nguyên mẫu hàm.
- Nguyên mẫu hàm thực chất là dòng đầu của hàm có thêm dấu chấm phẩy.



Chương 5: Hàm

Ví dụ:

```
void HoanVi(int *a, int *b); /*Khai báo nguyên  
mẫu hàm*/
```

```
void main() {  
    int x = 3, y = 7;  
    HoanVi(&x, &y); /*Gọi hàm HoanVi() được khai báo  
sau*/  
    <In x, y>  
}  
void HoanVi(int *a, int *b) {  
    int tam;  
    tam = *a; *a = *b; *b = tam;  
}
```

```
26
27     int n;
28
29     do
30     {
31         printf("\nNhap vao so nguyen duong (n > 0): ");
32         scanf("%d", &n);
33     } while (n <= 0);
34
35     /* 3.4.3.5.10. Hãy đếm số lượng chữ số của số nguyên dương n giống như chữ số đầu tiên
36     của số nguyên dương n.
37     Ví dụ: Nhập n = 10151. In ra màn hình: Có ba chữ số 1. */
38
39     // B1: Tìm k
40     int k = 1;
41     while (k <= n)
42     {
43         k *= 10;
44     }
45
46     k = k / 10;
47     printf("\nk = %d", k);
48
49     // B2: Dem
50     int So = n, Dem = 0;
51     int ChuSoDauTien = n / k;
52
53     while (So > 0)
54     {
55         int x = So % 10;
56
57         if (x == ChuSoDauTien)
58             Dem++;
59
60         So /= 10;
61     }
62
63     printf("\nCo %d chu so %d", Dem, ChuSoDauTien);
64
```



```
10
11     for (int i = 2; i < n; i++)
12     {
13         if (n % i == 0)
14         {
15             KT = 0;
16             break;
17         }
18     }
19
20     return KT;
21 }
```

```
22
23 void TangGiaTri(int &n)
24 {
25     n = n + 10;
26 }
```

*ví dụ tham số
và tham trị*

```
27
28 void main()
29 {
30     srand(unsigned int(time(NULL)));
31
32     int n = 10;
33
34     TangGiaTri(n);
35
36     printf("\nSau khi gọi hàm: n = %d", n);
37
38
39     /*do
40     {
41         printf("\nNhập vào số nguyên dương (n > 0): ");
42         scanf("%d", &n);
43     } while (n <= 0);*/
44 }
```

```
46     int x = So % 10;
47
48     if (x % 2 == 0)
49     {
50         KT = 0;
51         break;
52     }
53
54     So /= 10;
55 }
56
57 if (KT == 1)
58     printf("\n%d toan le", n);
59 else
60     printf("\n%d khong toan le", n);
61
62 // printf("\nCo %d chu so %d", Dem, ChuSoDauTien);
63
64
65 // Tich = 1, n = 6
66 // 1: tich = 1 x 2 = 2, 2 <= 6
67 // 2: tich = 2 x 2 = 4, 4 <= 6
68 // 3: tich = 4 x 2 = 8, 8 > 6
69 // tich = tich / 2 = 4, 4 != 6 => 6 khong phai 2k
70
71 // tich = 1, n = 8
72 // 1: tich = 2, 2 <= 8
73 // 2: tich = 4, 4 <= 8
74 // 3: tich = 8, 8 <= 8
75 // 4: tich = 16, 16 > 8
76 // tich = tich / 2 = 8, 8 == 8, 8 => 2k
77
78 }
79
```

```
1  #include<stdio.h>
2
3  int nhapso() {
4      unsigned int n;
5      do {
6          printf("moi nhap so nguyen duong: ");
7          scanf("%u", &n);
8      } while (n <= 0);
9      return (n);
10 }
11
12 int ktra2muk (unsigned int n) {
13     unsigned int tich=1, i =1, check = 0;
14
15     while (tich <= n) {
16         tich *= 2;
17     }
18     // printf("%u\n", tich);
19
20     if (tich / 2 == n) {
21         check = 1;
22     }
23     // printf("So nhap n: %u\n", n);
24     // printf("%u\n", check);
25     return check;
26 }
27
28
29 void main() {
30     unsigned int n, check = 0;
31     n = nhapso();
32     check = ktra2muk(n);
33
34     if (check == 1) {
35         printf("%u la so 2 mu k",n);
36     }
37     else {
38         printf("%u KHONG la so 2 mu k",n);
39     }
40 }
```

```
1 // 3.4.3.6trg27
2 #include<stdio.h>
3
4 int timUSCLN(unsigned int a, unsigned int b) {
5     unsigned int x = a;
6     unsigned int y = b;
7
8     while (x != y)
9     {
10         if (x > y) {
11             x = x - y;
12             //printf("%d\n", x);
13         }
14         else {
15             y = y - x;
16             //printf("%d\n", y);
17         }
18     }
19     return x;
20 }
21
22 int nhapso() {
23     unsigned int n;
24     do {
25         printf("moi nhap so nguyen duong > 0: ");
26         scanf("%u", &n);
27     } while (n <= 0);
28     return (n);
29 }
30
31 void main() {
32     unsigned int a, b, c;
33     a = nhapso();
34     b = nhapso();
35     c = timUSCLN(a, b);
36     printf("USCLN cua %u va %u : %u", a, b, c);
37 }
```

```
49     {
50         KT = 0;
51         break;
52     }
53
54     So /= 10;
55 }
56
57 if (KT == 1)
58     printf("\n%d toan le", n);
59 else
60     printf("\n%d khong toan le", n);
61
62 // printf("\nCo %d chu so %d", Dem, ChuSoDauTien);
63
64
65 // n = 12345 -> So Nghich Dao: 54321
66 // n = 12321 -> so Nghich Dao: 12321
67
68 // tong = 0
69 // b1: x = 5, tong = 0 * 10 + 5 = 5
70 // b2: x = 4, tong = 5 * 10 + 4 = 54
71 // b3: x = 3, tong = 54 * 10 + 3 = 543
72
73 /* 3.4.3.5.16. Hãy kiểm tra các chữ của số nguyên dương n có tăng dần từ trái sang phải
74 hay không?
75 Ví dụ: n = 12345 -> tăng dần từ trái -> phải, n = 12321 -> không tăng dần từ trái qua phải*/
76
77 }
78
```

```
1  #include<stdio.h>
2
3  int nhapso() {
4      unsigned int n;
5      do {
6          printf("moi nhap so nguyen duong: ");
7          scanf("%u", &n);
8      } while (n <= 0);
9      return (n);
10 }
11
12 unsigned int tongDonSo(unsigned int n) {
13     unsigned int a = 0, tong = 0;
14     while (n > 0) {
15         a = n % 10;
16         tong = tong * 10 + a;
17         n /= 10;
18     }
19     return tong;
20 }
21
22 void main() {
23     unsigned int n, s=0;
24     n = nhapso();
25     s = tongDonSo(n);
26     printf("So dao: %u", s);
27 }
```

```
1      #include<stdio.h>
2
3      unsigned int nhapsoND() {
4          unsigned int n;
5          do {
6              printf("moi nhap so nguyen duong > 0: ");
7              scanf("%u", &n);
8          } while (n <= 0);
9          return (n);
10     }
11
12     bool ktraSNT(unsigned int n) {
13         int ktra = 1; // 1 la SNT
14         for (int i = 2; i < n; i++) {
15             if (n % i == 0) {
16                 ktra = 0;
17                 break;
18             }
19         }
20
21         if (ktra == 1 && n > 1) {
22             return true;
23         }
24         else {
25             return false;
26         }
27     }
28
29     void main() {
30         unsigned int n;
31         n = nhapsoND();
32         for (unsigned int i = n; i > 0; i--) {
33             if (ktraSNT(i)) {
34                 printf("So nguyen to gan %u nhat la %u", n, i);
35                 break;
36             }
37         }
38     }
```

```
1  #include <stdio.h>
2  #include <math.h>
3
4  // 3.4.3.5.1
5  void main() {
6      unsigned int n, tmp=0, tmp1=0, i=0, k=0;
7      printf("vui long nhap so nguyen bat ky: ");
8      scanf("%d", &n);
9      tmp = n;
10
11     while (tmp > 0) {
12         k += 1;
13         tmp /= 10;
14     }
15
16     tmp1 = pow(10, k-1);
17
18     int sL = n / tmp1;
19     printf("\nSo hang %d: %d", tmp1, sL);
20     n %= tmp1;
21     do {
22         sL = n / (tmp1*=0.1);
23         printf("\nSo hang %d: %d", tmp1, sL);
24         n %= tmp1;
25     } while (n > 0);
26 }
27
```



```
1  #include <stdio.h>
2
3  bool ktraSNT(unsigned int n) {
4      int kt = 1; // SNT
5      for (int i = 2; i < n; i++) {
6          if (n % i == 0) {
7              kt = 0;
8              break;
9          }
10     }
11
12     if (kt == 1 && n > 1) {
13         return true;
14     }
15     else {
16         return false;
17     }
18 }
19
20 unsigned int nhapsoND() {
21     unsigned int n;
22     do {
23         printf("Vui long nhap so nguyen duong > 0: ");
24         scanf("%u", &n);
25     } while (n <= 0);
26
27     return n;
28 }
29
```

```
19
20 unsigned int nhapsoND() {
21     unsigned int n;
22     do {
23         printf("Vui long nhap so nguyen duong > 0: ");
24         scanf("%u", &n);
25     } while (n <= 0);
26
27     return n;
28 }
29
30 void main() {
31     unsigned int n;
32     n = nhapsoND();
33
34     // cach 1
35     //for (unsigned int i = n+1;; i++) {
36     //    if (ktraSNT(i)) {
37     //        printf("---> %u la nguyen to lon hon va gan voi %u nhat", i, n);
38     //        break;
39     //    }
40     //}
41
42     // cach 2:
43     unsigned int i = n;
44     do {
45         i++;
46     } while (ktraSNT(i) == false);
47     printf("---> %u la nguyen to lon hon va gan voi %u nhat", i, n);
48 }
```

```
19
20     return KT;
21 }
22
23 void NhapGiaTri(int &n)
24 {
25     do
26     {
27         printf("\nNhap vao so nguyen duong (n > 0): ");
28         scanf("%d", &n);
29     } while (n <= 0);
30 }
31
32 void main()
33 {
34     srand(unsigned int(time(NULL)));
35
36     int n = 0;
37
38     NhapGiaTri(n);
39
40     printf("\nn = %d", n);
41
42     // n = 20
43     // 1 2 4 5 10
44
45     // 24 I
46     // 20
47
48
49
50 }
51
```