

[Dashboard](#) / [My courses](#) / [SFP:HK2-2019-2020](#) / [Con trỏ](#) / [Bài tập con trỏ](#)

Started on	Wednesday, 20 May 2020, 10:13 AM
State	Finished
Completed on	Wednesday, 10 June 2020, 8:16 PM
Time taken	21 days 10 hours
Marks	4.00/6.00
Grade	6.67 out of 10.00 (67%)

Question **1**

Correct

Mark 1.00 out of 1.00

Viết [hàm](#) tính tổng 2 số nguyên.

Khuôn dạng (Prototype)

- Tên [hàm](#) (function name): **add**
- Danh sách tham số (parameters):
 - a và b: int**
 - c: con trỏ int**
- Kiểu trả về (return type): **void**

Thân hàm (Body)

- Tính **a + b** và lưu kết quả vào ô nhớ mà c trỏ đến

Chú ý

- Chỉ viết [hàm](#) add, KHÔNG VIẾT TOÀN BỘ CHƯƠNG TRÌNH
- Xem thêm chi tiết trong phần **For example**.

For example:

Test	Result
int c; add(5, 2, &c); printf("%d\n", c);	7
int c; add(-5, 1, &c); printf("%d\n", c);	-4
int c; add(5, 2, &c); add(c, -8, &c); printf("%d\n", c);	-1

Answer: (penalty regime: 33.3, 66.7, ... %)

1

2

3

```
void add(int a, int b, int *c){  
    *c = a + b;  
}
```

	Test	Expected	Got	
✓	int c; add(5, 2, &c); printf("%d\n", c);	7	7	✓
✓	int c; add(-5, 1, &c); printf("%d\n", c);	-4	-4	✓
✓	int c; add(5, 2, &c); add(c, -8, &c); printf("%d\n", c);	-1	-1	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **2**

Correct

Mark 1.00 out of 1.00

Viết 1 [hàm](#) cho phép trả về n phần tử đầu tiên của 1 cấp số cộng với công sai là d (phần tử đầu tiên là 1).

Khuôn dạng (Prototype)

- Tên [hàm](#) (function name): **getSequence()**
- Tham số (parameters):
 - n** - kiểu int (n>0), số phần tử của cấp số cộng
 - d** - kiểu int, công sai
- Kiểu trả về (return type): **int*** (con trỏ int)

Thân hàm (Body)

- Con trỏ chỉ đến phần tử đầu tiên của dãy cấp số cộng gồm n phần tử.

Chú ý

- Giá trị của các tham số luôn hợp lệ, không cần kiểm tra.
- Chỉ viết [hàm](#), KHÔNG VIẾT TOÀN BỘ CHƯƠNG TRÌNH
- Xem thêm chi tiết trong phần **For example**.

Gợi ý

- Cấp phát vùng nhớ n số nguyên
- Số hạng thứ i của cấp số cộng chính là số nguyên thứ i của vùng nhớ được cấp phát.
- Kết quả trả về của [hàm](#) là địa chỉ của vùng nhớ được cấp phát

For example:

Test	Result
<pre>int *a; int n=5, d=2; int i; a=getSequence(n,d); for(i=0;i<=n-1;i++) printf("%d ",a[i]); free(a);</pre>	1 3 5 7 9
<pre>int *a; int n=6, d=3; int i; a=getSequence(n,d); for(i=0;i<=n-1;i++) printf("%d ",a[i]); free(a);</pre>	1 4 7 10 13 16
<pre>int *a; int n=1, d=2; int i; a=getSequence(n,d); for(i=0;i<=n-1;i++) printf("%d ",a[i]); free(a);</pre>	1

Answer: (penalty regime: 33.3, 66.7, ... %)

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  int *getSequence(int n, int d){
4      int *a;
5      a = (int*)malloc(sizeof(int)*n);
6      int i = 0;
7      *(a+i) = 1;
8      for(i = 1; i < n; i++){
9          *(a+i) = *(a+i-1) + d;
10     }
11     return a;
12 }
```

	Test	Expected	Got	
✓	<pre>int *a; int n=5, d=2; int i; a=getSequence(n,d); for(i=0;i<=n-1;i++) printf("%d ",a[i]); free(a);</pre>	1 3 5 7 9	1 3 5 7 9	✓
✓	<pre>int *a; int n=6, d=3; int i; a=getSequence(n,d); for(i=0;i<=n-1;i++) printf("%d ",a[i]); free(a);</pre>	1 4 7 10 13 16	1 4 7 10 13 16	✓
✓	<pre>int *a; int n=1, d=2; int i; a=getSequence(n,d); for(i=0;i<=n-1;i++) printf("%d ",a[i]); free(a);</pre>	1	1	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **3**

Correct

Mark 0.00 out of 1.00

Viết 1 [hàm](#) cho phép trả về giá trị lớn nhất và vị trí của số lớn nhất trong mảng A có n số thực.

Khuôn dạng (Prototype)

- Tên [hàm](#) (function name): **getMax()**
- Tham số (parameters):
 - A[]** - kiểu float,
 - N** - kiểu int, số phần tử của mảng
 - pMaxValue - con trỏ kiểu float
 - pMaxPos - con trỏ kiểu int
- Kiểu trả về (return type): **không**

Thân [hàm](#) (Body)

- Tìm giá trị lớn nhất và vị trí của giá trị này trong mảng A

Chú ý

- Giá trị của các tham số luôn hợp lệ, không cần kiểm tra.
- Chỉ viết [hàm](#), KHÔNG VIẾT TOÀN BỘ CHƯƠNG TRÌNH
- Xem thêm chi tiết trong phần **For example**.

Gợi ý

- Áp dụng giải thuật tìm giá trị lớn nhất trong mảng
- Sau khi kết thúc giải thuật cho nội dung của con trỏ pMaxValue chỉ tới chính là giá trị lớn nhất và nội dung con trỏ pMaxPos là vị trí lớn nhất.

For example:

Test	Result
float a[] = {1.2}; int n =sizeof(a)/sizeof(float); float Max; int vtMax; getMax(a,n,&Max, &vtMax); printf("%.4f at %d",Max,vtMax);	1.2000 at 0

Answer: (penalty regime: 33.3, 66.7, ... %)

```
1 #include <stdio.h>
2 void getMax(float A[], int N, float *pMaxvalue, int *pMaxPos){
3     int i;
4     float tmp;
5     int vt = 0;
6     tmp = A[0];
7     for(i = 0; i < N; i++){
8         if(tmp < A[i]){
9             tmp = A[i];
10            vt = i;
11        }
12    }
13    *pMaxvalue = tmp;
14    *pMaxPos = vt ;
15 }
```

Test	Expected	Got	
------	----------	-----	--

	Test	Expected	Got	
✓	float a[] = {1.2, 2.4,3.0,0.0,9.5,2.0}; int n =sizeof(a)/sizeof(float); float Max; int vtMax; getMax(a,n,&Max, &vtMax); printf("%.4f at %d",Max,vtMax);	9.5000 at 4	9.5000 at 4	✓
✓	float a[] = {1.2, 2.4,30.0,0.0,9.5,30.0}; int n =sizeof(a)/sizeof(float); float Max; int vtMax; getMax(a,n,&Max, &vtMax); printf("%.4f at %d",Max,vtMax);	30.0000 at 2	30.0000 at 2	✓
✓	float a[] = {1.2}; int n =sizeof(a)/sizeof(float); float Max; int vtMax; getMax(a,n,&Max, &vtMax); printf("%.4f at %d",Max,vtMax);	1.2000 at 0	1.2000 at 0	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00. Accounting for previous tries, this gives **0.00/1.00**.

Question **4**

Correct

Mark 0.00 out of 1.00

Cho biến a có kiểu int và một con trỏ p được khai báo như sau:

```
void *p;  
int a = 5, b = 2;  
p = &a;
```

Hãy viết ĐỦ lệnh cần thiết làm các công việc sau:

- Lấy giá trị của a lưu vào b.

Chú ý

- Chỉ viết các lệnh theo yêu cầu, KHÔNG VIẾT TOÀN BỘ CHƯƠNG TRÌNH
- Phải sử dụng p không được sử dụng tên gọi của biến a

Answer: (penalty regime: 33.3, 66.7, ... %)

```
1 // void *p;  
2 // int a = 5, b = 2;  
3 // p = &a;  
4 b = *(int*) p;  
5  
6  
7
```

Test	Expected	Got

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00. Accounting for previous tries, this gives **0.00/1.00**.

Question **5**

Correct

Mark 1.00 out of 1.00

Viết 1 [hàm](#) cho phép chèn x vào vị trí p trong mảng gồm n số nguyên.

Khuôn dạng (Prototype)

- Tên [hàm](#) (function name): **Insert()**
- Tham số (parameters):
 - x** - kiểu int
 - p** - kiểu int
 - A[]** - kiểu int,
 - pn** - con trỏ kiểu int, số phần tử của mảng
- Kiểu trả về (return type): **không**

Thân [hàm](#) (Body)

- Chèn x vào vị trí p trong mảng n số nguyên

Chú ý

- Giá trị của các tham số luôn hợp lệ, không cần kiểm tra.
- Chỉ viết [hàm](#), KHÔNG VIẾT TOÀN BỘ CHƯƠNG TRÌNH
- Xem thêm chi tiết trong phần **For example**.

Gợi ý

- Gọi n là nội dung vùng nhớ mà con trỏ pn chỉ tới.
- Các phần tử từ vị trí p trở ra sau được dời về sau 1 vị trí
- $\forall i = n-1, p$

$a[i+1] = a[i]$

- Đặt x vào vị trí p trong mảng A: $A[p]=x$;
- Số phần tử tăng lên 1: $n=n+1$
- Cho nội dung vùng nhớ mà con trỏ pn chỉ tới nhận giá trị là n

For example:

Test	Result
<pre>int a[5]; a[0]=1;a[1]=5;a[2]=2;a[3]=0; int n = 4; int x,i,p; x=100; p=4; Insert(x,p,a, &n); for(i=0;i<=n-1;i++) printf("%d ",a[i]);</pre>	1 5 2 0 100
<pre>int a[5]; a[0]=1;a[1]=5;a[2]=2;a[3]=0; int n = 4; int x,i,p; x=100; p=2; Insert(x,p,a, &n); for(i=0;i<=n-1;i++) printf("%d ",a[i]);</pre>	1 5 100 2 0
<pre>int a[7]; a[0]=1;a[1]=5;a[2]=2;a[3]=0;a[4]=7;a[5]=6; int n = 6; int x,i,p; x=100; p=0; Insert(x,p,a, &n); for(i=0;i<=n-1;i++) printf("%d ",a[i]);</pre>	100 1 5 2 0 7 6

Answer: (penalty regime: 33.3, 66.7, ... %)

```
1 #include <stdio.h>
2 void Insert( int x, int p, int A[], int *pn){
3     int i;
```



```
4      int n = *pn;
5
6      for(i = n-1; i > p-1; i--){
7          A[i+1] = A[i];
8      }
9      n++;
10     A[p] = x;
11     *pn = n;
12 }
```

	Test	Expected	Got	
✓	<pre>int a[5]; a[0]=1;a[1]=5;a[2]=2;a[3]=0; int n = 4; int x,i,p; x=100; p=4; Insert(x,p,a, &n); for(i=0;i<=n-1;i++) printf("%d ",a[i]);</pre>	1 5 2 0 100	1 5 2 0 100	✓
✓	<pre>int a[5]; a[0]=1;a[1]=5;a[2]=2;a[3]=0; int n = 4; int x,i,p; x=100; p=2; Insert(x,p,a, &n); for(i=0;i<=n-1;i++) printf("%d ",a[i]);</pre>	1 5 100 2 0	1 5 100 2 0	✓
✓	<pre>int a[7]; a[0]=1;a[1]=5;a[2]=2;a[3]=0;a[4]=7;a[5]=6; int n = 6; int x,i,p; x=100; p=0; Insert(x,p,a, &n); for(i=0;i<=n-1;i++) printf("%d ",a[i]);</pre>	100 1 5 2 0 7 6	100 1 5 2 0 7 6	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Question **6**

Correct

Mark 1.00 out of 1.00

Viết 2 hàm, hàm thứ nhất cho phép chép các số nguyên lẻ của 1 mảng sang mảng thứ 2 còn hàm *thứ hai* cho phép hiển thị một mảng 1 chiều gồm n số nguyên

Khuôn dạng (Prototype)

- Tên hàm (function name) 1: **copyOddNumbers()**
- Tham số (parameters) của hàm 1:
 - A[]** - kiểu **int**, mảng nguồn
 - N** - kiểu **int**, số phần tử của mảng nguồn A
 - B[]** - kiểu **int**, mảng kết quả
 - pM** - con trỏ kiểu **int**, con trỏ chỉ đến số phần tử của mảng kết quả
- Kiểu trả về (return type): không

- Tên hàm (function name) 2: **printList()**
- Tham số (parameters) của hàm 2:
 - A[]** - kiểu **int**
 - N** - kiểu **int**, số phần tử của mảng A
- Kiểu trả về (return type): không

Thân hàm (Body)

- Hàm **copyOddNumbers()**: Đưa các số lẻ trong A vào mảng B, pM chỉ đến địa chỉ của số phần tử của B
- Hàm **printList()**: hiển thị các số trong mảng lên màn hình

Chú ý

- Giá trị của các tham số luôn hợp lệ, không cần kiểm tra.
- Chỉ viết 2 hàm, KHÔNG VIẾT TOÀN BỘ CHƯƠNG TRÌNH
- Xem thêm chi tiết trong phần **For example**.

Gợi ý cho hàm sao chép

- Duyệt qua mảng A, nếu gặp phần tử là số lẻ thì đưa vào trong mảng B.
- Con trỏ pM chỉ đến địa chỉ của số phần tử của mảng B

For example:

Test	Result
<pre>int a[]={1,5,6,8,71,0,12,13,11,18}; int b[1000]; int m,n; n = sizeof(a)/sizeof(int); printList(a,n); copyOddNumbers(a,n,b,&m); printList(b,m);</pre>	<pre>1 5 6 8 71 0 12 13 11 18 1 5 71 13 11</pre>
<pre>int a[]={1,3,5,7}; int b[1000]; int m,n; n = sizeof(a)/sizeof(int); printList(a,n); copyOddNumbers(a,n,b,&m); printList(b,m);</pre>	<pre>1 3 5 7 1 3 5 7</pre>

Answer: (penalty regime: 33.3, 66.7, ... %)

```
1 #include <stdio.h>
2 void copyOddNumbers(int a[], int n, int b[], int *pM){
3     int i,j=0;
4     for(i = 0; i < n; i++){
5         if(a[i] % 2 != 0){
6             b[j] = a[i];
```

```
7         j++;
8     }
9 }
10 *pM = j;
11 }
12 void printList(int a[],int n){
13     int i;
14     for(i = 0; i < n; i++){
15         printf("%d ", a[i]);
16     }
17     printf("\n");
18 }
```

	Test	Expected	Got	
✓	<pre>int a[]={1,5,6,8,71,0,12,13,11,18}; int b[1000]; int m,n; n = sizeof(a)/sizeof(int); printList(a,n); copyOddNumbers(a,n,b,&m); printList(b,m);</pre>	<pre>1 5 6 8 71 0 12 13 11 18 1 5 71 13 11</pre>	<pre>1 5 6 8 71 0 12 13 11 18 1 5 71 13 11</pre>	✓
✓	<pre>int a[]={2, 4, 6, 8}; int b[1000]; int m,n; n = sizeof(a)/sizeof(int); printList(a,n); copyOddNumbers(a,n,b,&m); printList(b,m);</pre>	<pre>2 4 6 8</pre>	<pre>2 4 6 8</pre>	✓
✓	<pre>int a[]={1,3,5,7}; int b[1000]; int m,n; n = sizeof(a)/sizeof(int); printList(a,n); copyOddNumbers(a,n,b,&m); printList(b,m);</pre>	<pre>1 3 5 7 1 3 5 7</pre>	<pre>1 3 5 7 1 3 5 7</pre>	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[◀ Một số thao tác trên mảng 1 chiều](#)

Jump to...

[Một số vấn đề cơ bản trong lập trình ▶](#)