

Trạng thái	Đã xong
Bắt đầu vào lúc	Thứ Bảy, 15 tháng 2 2025, 4:11 PM
Kết thúc lúc	Thứ Bảy, 15 tháng 2 2025, 4:29 PM
Thời gian thực hiện	18 phút 4 giây
Điểm	100,00/100,00
Điểm	10,00 trên 10,00 (100%)

Câu hỏi 1

Đúng

Đạt điểm 10,00 trên 10,00

Cứ hết một năm học, giáo viên trong các trường trung học cơ sở sẽ phải tính toán điểm trung bình của học sinh theo hệ số điểm tương ứng.

Viết chương trình nhập vào điểm thành phần của một học sinh và in ra điểm trung bình tương ứng.

Biết các điểm lần lượt nhập theo thứ tự: 3 điểm hệ số 1, 2 điểm hệ số 2 và 1 điểm hệ số 3.

Kết quả làm tròn đến hai chữ số thập phân đằng sau dấu phẩy.

Đầu vào: Điểm thành phần của một học sinh gồm:

- 3 điểm hệ số 1;
- 2 điểm hệ số 2;
- 1 điểm hệ số 3.

Đầu ra: Điểm trung bình môn học của học sinh đó. Kết quả làm tròn đến hai chữ số sau dấu phẩy.

Dữ liệu vào nhập từ bàn phím và kết quả được in ra màn hình. (Chi tiết xem ở



).

For example:

Input	Result
1 2 3 4 5 6	4.20

Answer: (penalty regime: 0 %)

```
1 #include <iostream>
2 #include <vector>
3 #include <iomanip>
4 using namespace std;
5
6 signed main() {
7     double a, b, c, d, e, f;
8     cin >> a >> b >> c >> d >> e >> f;
9     double sum = (a + b + c) + 2 * (d + e) + 3 * f;
10    cout << fixed << setprecision(2);
11    cout << sum / 10;
12 }
```

	Input	Expected	Got	
✓	1 2 3 4 5 6	4.20	4.20	✓
✓	3 4 5 6 7 8	6.20	6.20	✓
✓	4.3 2.5 4.6 7.8 9.8 8.4	7.18	7.18	✓

Passed all tests! ✓

Đúng

Marks for this submission: 10,00/10,00.

## Câu hỏi 2

Đúng

Đạt điểm 10,00 trên 10,00

**Chỉ số khối cơ thể** - thường được biết đến với chữ viết tắt **BMI** theo tên tiếng Anh *Body Mass Index* - được dùng để đánh giá mức độ gầy hay béo của một người. Chỉ số này do nhà bác học người Bỉ Adolphe Quetelet đưa ra năm 1832.

Chỉ số khối cơ thể của một người tính bằng cân nặng của người đó (kg) chia cho bình phương chiều cao (m).

Chỉ số này có thể giúp xác định một người bị bệnh béo phì hay bị bệnh suy dinh dưỡng thông qua số liệu về hình dáng, chiều cao và cân nặng cơ thể.

Viết chương trình nhận vào cân nặng và chiều cao của một người và in ra chỉ số khối cơ thể của người đó.

Kết quả được làm tròn đến hai chữ số thập phân sau dấu phẩy sử dụng `std::cout << std::fixed << std::setprecision << number;`

**Dữ liệu nhập vào từ bàn phím và kết quả in ra màn hình.**

**For example:**

Input	Result
52 1.66	18.87

**Answer:** (penalty regime: 0 %)

```
1 #include <iostream>
2 #include <vector>
3 #include <iomanip>
4 using namespace std;
5
6 signed main() {
7     double kg, m;
8     cin >> kg >> m;
9
10    cout << fixed << setprecision(2);
11    cout << kg / (m * m);
12 }
```

	Input	Expected	Got	
✓	52 1.66	18.87	18.87	✓
✓	49 1.56	20.13	20.13	✓
✓	46 1.58	18.43	18.43	✓

Passed all tests! ✓

Đúng

Marks for this submission: 10,00/10,00.

Câu hỏi 3

Đúng

Đạt điểm 10,00 trên 10,00

Viết chương trình nhập một ngày bất kỳ từ bàn phím và in ra ngày đó là thứ mấy trong tuần.

Chương trình nhận 3 tham số đầu vào lần lượt là  $d$  (ngày),  $m$  (tháng) và  $y$  (năm).

In ra 0 cho Chủ nhật, 1 cho Thứ hai, ...

Sử dụng công thức dưới đây để tính:

$$y_0 = y - (14 - m) / 12$$

$$x = y_0 + y_0 / 4 - y_0 / 100 + y_0 / 400$$

$$m_0 = m + 12 \times ((14 - m) / 12) - 2$$

$$d_0 = (d + x + 31 \times m_0 / 12) \bmod 7$$

(Các phép chia đều là chia lấy phần nguyên).

Dữ liệu nhập vào từ bàn phím và kết quả in ra màn hình.

For example:

Input	Result
2 8 1953	0

Answer: (penalty regime: 0 %)

```
1 #include <iomanip>
2 #include <iostream>
3 #include <vector>
4
5 using namespace std;
6
7 signed main() {
8     int d, m, y;
9     cin >> d >> m >> y;
10
11     int y0 = y - (14 - m) / 12;
12     int x = y0 + y0 / 4 - y0 / 100 + y0 / 400;
13     int m0 = m + 12 * ((14 - m) / 12) - 2;
14     int d0 = (d + x + 31 * m0 / 12) % 7;
15
16     cout << d0;
17 }
```

	Input	Expected	Got	
✓	2 8 1953	0	0	✓
✓	1 1 2000	6	6	✓

	Input	Expected	Got	
✓	28 7 2017	5	5	✓
✓	1 1 1	1	1	✓

Passed all tests! ✓

Đúng

Marks for this submission: 10,00/10,00.

#### Câu hỏi 4

Đúng

Đạt điểm 10,00 trên 10,00

Viết chương trình nhận bốn số thực nhập từ bàn phím:  $x_1, y_1, x_2$  và  $y_2$ . In ra màn hình khoảng cách Euclide giữa hai điểm  $(x_1, y_1)$  và  $(x_2, y_2)$ .

Sử dụng `sqrt()` trong thư viện chuẩn `cmath`.

Kết quả làm tròn đến hai chữ số thập phân đằng sau dấu phẩy sử dụng `std::cout << std::fixed << std::setprecision(2) << number;`

Dữ liệu nhập vào từ bàn phím và kết quả in ra màn hình.

For example:

Input	Result
0 0 3 4	5.00

Answer: (penalty regime: 0 %)

```
1 #include <iomanip>
2 #include <iostream>
3 #include <vector>
4 #include <cmath>
5
6 using namespace std;
7
8 signed main() {
9     double a, b, c, d;
10    cin >> a >> b >> c >> d;
11
12    double dist = sqrt(pow(c-a, 2) + pow(d-b, 2));
13    cout << fixed << setprecision(2);
14    cout << dist;
15 }
```

	Input	Expected	Got	
✓	0 0 3 4	5.00	5.00	✓
✓	-12 11.2 3.4 2.5	17.69	17.69	✓

Passed all tests! ✓

Đúng

Marks for this submission: 10,00/10,00.



**Câu hỏi 5**

Đúng

Đạt điểm 10,00 trên 10,00

Nhập vào 3 số nguyên từ bàn phím và in ra màn hình theo thứ tự giảm dần.

*Hint: Sử dụng hàm `min()`, `max()`.*

**For example:**

Test	Input	Result
Test case 1	3 2 5	5 3 2

**Answer:** (penalty regime: 0 %)

```
1 #include <iomanip>
2 #include <iostream>
3 #include <vector>
4 #include <cmath>
5
6 using namespace std;
7
8 signed main() {
9     int a, b, c;
10    cin >> a >> b >> c;
11
12    int mn = min(a, min(b, c));
13    int mx = max(a, max(b, c));
14    int r = a + b + c - mx - mn;
15    cout << mx << ' ' << r << ' ' << mn;
16 }
```

	Test	Input	Expected	Got	
✓	Test case 1	3 2 5	5 3 2	5 3 2	✓
✓	Test case 2	1 -3 4	4 1 -3	4 1 -3	✓

Passed all tests! ✓

**Đúng**

Marks for this submission: 10,00/10,00.

Câu hỏi 6

Đúng

Đạt điểm 10,00 trên 10,00

Viết hàm `double calculate (double num1, char operat, double num2)` giả lập một máy tính bỏ túi.

Hàm thực hiện tính toán kết quả của phép toán giữa số *num1* và *num2* với toán tử *operat*

Biết các toán tử đơn giản thực hiện phép toán lần lượt là:

- Phép trừ `-`
- Phép cộng `+`
- Phép chia `/`
- Phép nhân `*`

Kết quả làm tròn đến chữ số thập phân thứ hai.

Đầu vào: Hai số thực và phép toán cần thực hiện (trong số các phép toán `+`, `-`, `*`, `/`).

Đầu ra: Kết quả phép toán.

For example:

Test	Input	Result
<code>calculate (num1, operat, num2)</code>	<code>2 + 3</code>	<code>5.00</code>

Answer: (penalty regime: 0 %)

```
1 double calculate (double num1, char operat, double num2) {
2     if (operat == '+') return num1 + num2;
3     if (operat == '-') return num1 - num2;
4     if (operat == '*') return num1 * num2;
5     return num1 / num2;
6 }
```

	Test	Input	Expected	Got	
✓	<code>calculate (num1, operat, num2)</code>	<code>2 + 3</code>	<code>5.00</code>	<code>5.00</code>	✓
✓	<code>calculate (num1, operat, num2)</code>	<code>12 / 4</code>	<code>3.00</code>	<code>3.00</code>	✓

	Test	Input	Expected	Got	
✓	calculate (num1, operat, num2)	23.89 * 23.12	552.34	552.34	✓
✓	calculate (num1, operat, num2)	82 - 12	70.00	70.00	✓
✓	calculate (num1, operat, num2)	12 * -23.004	-276.05	-276.05	✓

Passed all tests! ✓

Đúng

Marks for this submission: 10,00/10,00.

**Câu hỏi 7**

Đúng

Đạt điểm 10,00 trên 10,00

Viết hàm `void readArray(int a[], int n)` để đọc vào  $n$  số nguyên từ bàn phím và lưu vào mảng số nguyên  $a$  và hàm `void printArray(int a[], int n)` để in các phần tử của mảng  $a$  có số lượng phần tử là  $n$ , các phần tử cách nhau bởi dấu cách.

*Lưu ý:* chỉ cần viết 2 hàm như đề bài yêu cầu, không cần viết hàm `main()`, không cần viết các câu lệnh `#include`, `using namespace std` ...

**For example:**

Test	Input	Result
<pre>int n = 5; int a[n]; readArray(a, n); printArray(a, n);</pre>	1 2 3 4 5	1 2 3 4 5

**Answer:** (penalty regime: 0 %)

```
1 void readArray(int a[], int n) {
2     for (int i = 0; i < n; ++i) {
3         cin >> a[i];
4     }
5 }
6
7 void printArray(int a[], int n) {
8     for (int i = 0; i < n; ++i) {
9         cout << a[i] << ' ';
10    }
11 }
```

	Test	Input	Expected	Got	
✓	<pre>int n = 5; int a[n]; readArray(a, n); printArray(a, n);</pre>	1 2 3 4 5	1 2 3 4 5	1 2 3 4 5	✓

	Test	Input	Expected	Got	
✓	<pre>int n = 10; int a[n]; readArray(a, n); printArray(a, n);</pre>	-1 -2 -3 -4 -5 -6 -7 -8 -9 -10	-1 -2 -3 -4 -5 -6 -7 -8 -9 -10	-1 -2 -3 -4 -5 -6 -7 -8 -9 -10	✓
✓	<pre>int n = 50; int a[n]; readArray(a, n); printArray(a, n);</pre>	-945 740 -865 918 -766 613 449 148 367 287 -502 -43 -84 657 617 179 -763 537 801 -562 279 -459 851 -28 -986 -808 -255 360 -44 702 17 -519 -90 -910 -838 300 -390 757 -287 731 -129 -958 -955 512 -931 201 213 995 65 62	-945 740 -865 918 -766 613 449 148 367 287 -502 -43 -84 657 617 179 -763 537 801 -562 279 -459 851 -28 -986 -808 -255 360 -44 702 17 -519 -90 -910 -838 300 -390 757 -287 731 -129 -958 -955 512 -931 201 213 995 65 62	-945 740 -865 918 -766 613 449 148 367 287 -502 -43 -84 657 617 179 -763 537 801 -562 279 -459 851 -28 -986 -808 -255 360 -44 702 17 -519 -90 -910 -838 300 -390 757 -287 731 -129 -958 -955 512 -931 201 213 995 65 62	✓
✓	<pre>int n = 100; int a[n]; readArray(a, n); printArray(a, n);</pre>	948 530 492 -484 -692 -552 661 153 -838 -978 559 -626 -983 66 594 -408 497 -11 -587 -877 -54 -280 -683 554 -819 -306 -328 245 -510 -390 585 -40 787 -712 809 -888 -971 -70 832 -42 156 -136 -119 -717 -694 151 -22 555 530 -456 422 961 115 -411 -57 -273 597 911 -864 -157 -765 315 642 939 -494 -410 -253 -664 781 -393 -469 -205 713 984 724 997 -95 954 809 -468 -298 -591 -532 609 311 -597 198 -718 436 183 -721 608 638 -576 455 13 867 -477 127 -983	948 530 492 -484 -692 -552 661 153 -838 -978 559 -626 -983 66 594 -408 497 -11 -587 -877 -54 -280 -683 554 -819 -306 -328 245 -510 -390 585 -40 787 -712 809 -888 -971 -70 832 -42 156 -136 -119 -717 -694 151 -22 555 530 -456 422 961 115 -411 -57 -273 597 911 -864 -157 -765 315 642 939 -494 -410 -253 -664 781 -393 -469 -205 713 984 724 997 -95 954 809 -468 -298 -591 -532 609 311 -597 198 -718 436 183 -721 608 638 -576 455 13 867 -477 127 -983	948 530 492 -484 -692 -552 661 153 -838 -978 559 -626 -983 66 594 -408 497 -11 -587 -877 -54 -280 -683 554 -819 -306 -328 245 -510 -390 585 -40 787 -712 809 -888 -971 -70 832 -42 156 -136 -119 -717 -694 151 -22 555 530 -456 422 961 115 -411 -57 -273 597 911 -864 -157 -765 315 642 939 -494 -410 -253 -664 781 -393 -469 -205 713 984 724 997 -95 954 809 -468 -298 -591 -532 609 311 -597 198 -718 436 183 -721 608 638 -576 455 13 867 -477 127 -983	✓

	Test	Input	Expected	Got	
✓	int n = 1000; int a[n]; readArray(a, n); printArray(a, n);	-309 390 423 541 26 318 -733 -649 631 774 -285 -850 -837 -68 -967 -56 510 -76 168 -822 351 74 -459 206 872 368 746 -745 554 462 531 -820 -297 -709 -217 732 -190 -136 -797 -561 497 -738 -697 195 916 58 -853 401 -440 219 608 229 -242 -185 141 -931 660 -598 -462 317 -721 -42 -750 -84 25 -53 835 727 -555 -626 681 -133 -697 -562 118 -443 -620 -268 720 -346 -180 785 376 618 557 -794 -18 961 814 122 644 -456 830 785 899 -843 -981 721 351 -700 343 923 644 77 -494 -862 467 -530 431 -762 -624 -529 -60 332 -349 -665 -865 183 734 -555 111 -672 -272 785 -143 -485 -146 -873 949 -628 -37 -640 996 -993 97 928 -712 37 -382 299 247 -871 898 -54 568 -377 -385 596 683 900 856 -743 22 101 -512 255 -388 779 465 -106 79 876 -911 -470 532 109 636 -423 870 573 698 -588 -284 -912 -745 -553 176 108 -622 -307 -407 -428 21 -84 -853 -195 -675 82 -154 750 -465 -731 75 -53 -629 724 482 854 501 556 725 -26 -605 122 -746 -822 535 13 -914 -597 -215 437 -224 -889 -979 -686 -345 7 902 -340 -971 256 -581 -972 -204 -999 267 -750 -757 -187 -847 754 410 -170 836 -435 -149 -178 291 45 488 38 308 -449 777 -240 236 -241 108 167 100 -981 -242 137 56 44 -811 -50 -220 -108 -438 -724 237 -252 -50 450 896 -877 -339 -25 504 -117 652 336 674 -342 -264 244 -942 656 -494 -87 333 776 -665 -213 -519 -643 -725 -650 453 258 631 225 163 781 -509 302 453 484 613 -801 -323 -501 357 880 -801 320 -936 280 355 466 -874 614 59 364 803 -916 909 647 481 319 508 868 -577 -43 116 31 715 -763 -986 -775 -148 -995 55 603 25 -896 -484 779 62 -363 377 378 335 -596 10 120 -681 224 991 -205 161 735 444 280 134 948 37 -677 -43 -57 -87 455 612 -320 161 -485	-309 390 423 541 26 318 -733 -649 631 774 -285 -850 -837 -68 -967 -56 510 -76 168 -822 351 74 -459 206 872 368 746 -745 554 462 531 -820 -297 -709 -217 732 -190 -136 -797 -561 497 -738 -697 195 916 58 -853 401 -440 219 608 229 -242 -185 141 -931 660 -598 -462 317 -721 -42 -750 -84 25 -53 835 727 -555 -626 681 -133 -697 -562 118 -443 -620 -268 720 -346 -180 785 376 618 557 -794 -18 961 814 122 644 -456 830 785 899 -843 -981 721 351 -700 343 923 644 77 -494 -862 467 -530 431 -762 -624 -529 -60 332 -349 -665 -865 183 734 -555 111 -672 -272 785 -143 -485 -146 -873 949 -628 -37 -640 996 -993 97 928 -712 37 -382 299 247 -871 898 -54 568 -377 -385 596 683 900 856 -743 22 101 -512 255 -388 779 465 -106 79 876 -911 -470 532 109 636 -423 870 573 698 -588 -284 -912 -745 -553 176 108 -622 -307 -407 -428 21 -84 -853 -195 -675 82 -154 750 -465 -731 75 -53 -629 724 482 854 501 556 725 -26 -605 122 -746 -822 535 13 -914 -597 -215 437 -224 -889 -979 -686 -345 7 902 -340 -971 256 -581 -972 -204 -999 267 -750 -757 -187 -847 754 410 -170 836 -435 -149 -178 291 45 488 38 308 -449 777 -240 236 -241 108 167 100 -981 -242 137 56 44 -811 -50 -220 -108 -438 -724 237 -252 -50 450 896 -877 -339 -25 504 -117 652 336 674 -342 -264 244 -942 656 -494 -87 333 776 -665 -213 -519 -643 -725 -650 453 258 631 225 163 781 -509 302 453 484 613 -801 -323 -501 357 880 -801 320 -936 280 355 466 -874 614 59 364 803 -916 909 647 481 319 508 868 -577 -43 116 31 715 -763 -986 -775 -148 -995 55 603 25 -896 -484 779 62 -363 377 378 335 -596 10 120 -681 224 991 -205 161 735 444 280 134 948 37 -677 -43 -57 -87 455 612 -320 161 -485 38 89 744 -467 475 -127 819 -406 -964 -420 -462 96 -252 777 -368 -565 634 -926 505 918 -484 -154 309 596 -894 235 -115 -745 -601 -570 560 -376 74 817 -150 -983 -196 983 -582	-309 390 423 541 26 318 -733 -649 631 774 -285 -850 -837 -68 -967 -56 510 -76 168 -822 351 74 -459 206 872 368 746 -745 554 462 531 -820 -297 -709 -217 732 -190 -136 -797 -561 497 -738 -697 195 916 58 -853 401 -440 219 608 229 -242 -185 141 -931 660 -598 -462 317 -721 -42 -750 -84 25 -53 835 727 -555 -626 681 -133 -697 -562 118 -443 -620 -268 720 -346 -180 785 376 618 557 -794 -18 961 814 122 644 -456 830 785 899 -843 -981 721 351 -700 343 923 644 77 -494 -862 467 -530 431 -762 -624 -529 -60 332 -349 -665 -865 183 734 -555 111 -672 -272 785 -143 -485 -146 -873 949 -628 -37 -640 996 -993 97 928 -712 37 -382 299 247 -871 898 -54 568 -377 -385 596 683 900 856 -743 22 101 -512 255 -388 779 465 -106 79 876 -911 -470 532 109 636 -423 870 573 698 -588 -284 -912 -745 -553 176 108 -622 -307 -407 -428 21 -84 -853 -195 -675 82 -154 750 -465 -731 75 -53 -629 724 482 854 501 556 725 -26 -605 122 -746 -822 535 13 -914 -597 -215 437 -224 -889 -979 -686 -345 7 902 -340 -971 256 -581 -972 -204 -999 267 -750 -757 -187 -847 754 410 -170 836 -435 -149 -178 291 45 488 38 308 -449 777 -240 236 -241 108 167 100 -981 -242 137 56 44 -811 -50 -220 -108 -438 -724 237 -252 -50 450 896 -877 -339 -25 504 -117 652 336 674 -342 -264 244 -942 656 -494 -87 333 776 -665 -213 -519 -643 -725 -650 453 258 631 225 163 781 -509 302 453 484 613 -801 -323 -501 357 880 -801 320 -936 280 355 -916 909 647 481 319 508 868 -577 -43 116 31 715 -763 -986 -775 -148 -995 55 603 25 -896 -484 779 62 -363 377 378 335 -596 10 120 -681 224 991 -205 161	✓

Test	Input	Expected	Got
	38 89 744 -467 475 -127 819	-876 -808 -655 -142 214 744 149	735 444 280 134 948 37 -677
	-406 -964 -420 -462 96 -252	357 937 -26 984 407 -135 307 -657	-43 -57 -87 455 612 -320 161
	777 -368 -565 634 -926 505 918	56 675 -417 410 -595 -273 -443	-485 38 89 744 -467 475 -127
	-484 -154 309 596 -894 235	-505 -863 955 77 -188 21 -335 230	819 -406 -964 -420 -462 96
	-115 -745 -601 -570 560 -376	966 -254 -689 -94 -784 -960 562	-252 777 -368 -565 634 -926
	74 817 -150 -983 -196 983 -582	799 452 -666 -616 478 710 644	505 918 -484 -154 309 596
	-876 -808 -655 -142 214 744	-608 -710 -952 -469 40 -593 -908	-894 235 -115 -745 -601 -570
	149 357 937 -26 984 407 -135	111 -164 127 -462 53 564 421 40	560 -376 74 817 -150 -983
	307 -657 56 675 -417 410 -595	543 -284 -543 -376 480 -137 -319	-196 983 -582 -876 -808 -655
	-273 -443 -505 -863 955 77	-748 845 -724 898 761 690 997 586	-142 214 744 149 357 937 -26
	-188 21 -335 230 966 -254 -689	-819 -16 -573 799 -812 12 -832 47	984 407 -135 307 -657 56 675
	-94 -784 -960 562 799 452 -666	837 545 356 -672 759 629 653 -513	-417 410 -595 -273 -443 -505
	-616 478 710 644 -608 -710	-128 912 -165 56 -923 845 259 555	-863 955 77 -188 21 -335 230
	-952 -469 40 -593 -908 111	-958 -148 -572 -560 294 -619 687	966 -254 -689 -94 -784 -960
	-164 127 -462 53 564 421 40	-515 963 -90 -264 992 881 870	562 799 452 -666 -616 478 710
	543 -284 -543 -376 480 -137	-775 -497 -645 -724 401 25 828	644 -608 -710 -952 -469 40
	-319 -748 845 -724 898 761 690	653 176 247 -311 -912 469 26 973	-593 -908 111 -164 127 -462
	997 586 -819 -16 -573 799 -812	89 -513 939 636 602 279 383 -473	53 564 421 40 543 -284 -543
	12 -832 47 837 545 356 -672	-293 717 659 -79 -204 399 811 834	-376 480 -137 -319 -748 845
	759 629 653 -513 -128 912 -165	720 -117 189 -459 30 -327 -659	-724 898 761 690 997 586 -819
	56 -923 845 259 555 -958 -148	-827 523 -288 -235 580 290 156	-16 -573 799 -812 12 -832 47
	-572 -560 294 -619 687 -515	-476 671 -550 936 82 502 -947 386	837 545 356 -672 759 629 653
	963 -90 -264 992 881 870 -775	-378 343 -436 697 66 -736 -272	-513 -128 912 -165 56 -923
	-497 -645 -724 401 25 828 653	-865 14 -226 700 709 -362 -467	845 259 555 -958 -148 -572
	176 247 -311 -912 469 26 973	-861 -530 27 -751 997 -632 510 62	-560 294 -619 687 -515 963
	89 -513 939 636 602 279 383	472 -864 -569 -169 745 -754 -786	-90 -264 992 881 870 -775
	-473 -293 717 659 -79 -204 399	391 251 -462 -126 -292 -337 -385	-497 -645 -724 401 25 828 653
	811 834 720 -117 189 -459 30	334 -250 163 834 -479 678 944 605	176 247 -311 -912 469 26 973
	-327 -659 -827 523 -288 -235	740 932 895 774 -779 -868 926	89 -513 939 636 602 279 383
	580 290 156 -476 671 -550 936	-760 -153 226 497 -112 -660 232	-473 -293 717 659 -79 -204
	82 502 -947 386 -378 343 -436	-802 279 42 173 -78 -681 -317	399 811 834 720 -117 189 -459
	697 66 -736 -272 -865 14 -226	-892 -1000 588 42 989 -106 25	30 -327 -659 -827 523 -288
	700 709 -362 -467 -861 -530 27	-835 -685 -665 -850 311 -248 -815	-235 580 290 156 -476 671
	-751 997 -632 510 62 472 -864	-319 -500 194 -184 -634 -621 508	-550 936 82 502 -947 386 -378
	-569 -169 745 -754 -786 391	319 -875 -484 -535 495 -82 -699 2	343 -436 697 66 -736 -272
	251 -462 -126 -292 -337 -385	-998 -397 71 -416 -426 -67 -218	-865 14 -226 700 709 -362
	334 -250 163 834 -479 678 944	-196 221 963 472 329 -302 166 901	-467 -861 -530 27 -751 997
	605 740 932 895 774 -779 -868	-677 -254 28 196 -787 156 -361 19	-632 510 62 472 -864 -569
	926 -760 -153 226 497 -112	-515 398 20 -312 810 -123 849 345	-169 745 -754 -786 391 251
	-660 232 -802 279 42 173 -78	813 -933 272 490 -154 -505 742	-462 -126 -292 -337 -385 334
	-681 -317 -892 -1000 588 42	913 -921 -467 -539 291 -190 336	-250 163 834 -479 678 944 605
	989 -106 25 -835 -685 -665	43 -597 -581 828 570 386 -542 217	740 932 895 774 -779 -868 926
	-850 311 -248 -815 -319 -500	-905 946 196 261 -565 101 -928	-760 -153 226 497 -112 -660
	194 -184 -634 -621 508 319	995 -241 529 249 436 111 546 -275	232 -802 279 42 173 -78 -681
	-875 -484 -535 495 -82 -699 2	514 288 -68 -298 -133 -778 701	-317 -892 -1000 588 42 989
	-998 -397 71 -416 -426 -67	-230 47 -942 -770 396 764 -61 749	-106 25 -835 -685 -665 -850
	-218 -196 221 963 472 329 -302	116 -909 -242 962 -591 406 962	311 -248 -815 -319 -500 194
	166 901 -677 -254 28 196 -787	-582 -819 766 207 -95 -225 -667	-184 -634 -621 508 319 -875
	156 -361 19 -515 398 20 -312	-406 722 989 828 944 498 -367 931	-484 -535 495 -82 -699 2 -998
	810 -123 849 345 813 -933 272	347 -600 -594 -819 780 756 770	-397 71 -416 -426 -67 -218
	490 -154 -505 742 913 -921	-638 -506 -163 -587 -73 -660 380	-196 221 963 472 329 -302 166
	-467 -539 291 -190 336 43 -597	-421 -438 -302 -532 619 22 57	901 -677 -254 28 196 -787 156
	-581 828 570 386 -542 217 -905	-258 917 -479 929 954 938 -293	-361 19 -515 398 20 -312 810
	946 196 261 -565 101 -928 995	-684 1 -638 -431 -316 0 945 -697	-123 849 345 813 -933 272 490
	-241 529 249 436 111 546 -275	-434 -748 -584 -738 594 361 32	-154 -505 742 913 -921 -467

	Test	Input	Expected	Got	
		514 288 -68 -298 -133 -778 701 -230 47 -942 -770 396 764 -61 749 116 -909 -242 962 -591 406 962 -582 -819 766 207 -95 -225 -667 -406 722 989 828 944 498 -367 931 347 -600 -594 -819 780 756 770 -638 -506 -163 -587 -73 -660 380 -421 -438 -302 -532 619 22 57 -258 917 -479 929 954 938 -293 -684 1 -638 -431 -316 0 945 -697 -434 -748 -584 -738 594 361 32 -393 -84 24 72 -245 593 -118 -654 -221 -909 601 -509 -520 -107 -390 604 -109 -39 132 -404 -326 332 -733 831 -246 380 660 -345 -662 -517 -226 154 -735 638 -324 354 928 -172 18 486 729 -983 -541 -272 -46 -251 208 -989 353 235 421 53 -952 -613 -44 914 830 639 833 713 -711 -564 519 -776 486 780 600 -145 773 994 782 574 428 -198 874 78 -103 -871 -594 -460 867 849 -276 766 -778 76 -958 453 -162 -240 994 -511 69 -863 827 -340 -410 -353 154 260 -255 914 -375 -617 -229 979 415 821 301 -318 446 -788 -335 -372 697 711 -725 749 137 387 699 883 -395 56 180 -265 -518 -965 -502 -85 668 763 -199 -669 -712 132 -145 -29 585 135 562 -816 -198 -753 -225 289 -800 -580 -301 52 347 305 162 -815 623 875 498 815 -865 720 -470 311 574 -481 -470 293 -580 -613 719 612 -852 -603 526 -6 -695 14 -820 99 464 -562 -161 -869 -854 835 -103 432	-393 -84 24 72 -245 593 -118 -654 -221 -909 601 -509 -520 -107 -390 604 -109 -39 132 -404 -326 332 -733 831 -246 380 660 -345 -662 -517 -226 154 -735 638 -324 354 928 -172 18 486 729 -983 -541 -272 -46 -251 208 -989 353 235 421 53 -952 -613 -44 914 830 639 833 713 -711 -564 519 -776 486 780 600 -145 773 994 782 574 428 -198 874 78 -103 -871 -594 -460 867 849 -276 766 -778 76 -958 453 -162 -240 994 -511 69 -863 827 -340 -410 -353 154 260 -255 914 -375 -617 -229 979 415 821 301 -318 446 -788 -335 -372 697 711 -725 749 137 387 699 883 -395 56 180 -265 -518 -965 -502 -85 668 763 -199 -669 -712 132 -145 -29 585 135 562 -816 -198 -753 -225 289 -800 -580 -301 52 347 305 162 -815 623 875 498 815 -865 720 -470 311 574 -481 -470 293 -580 -613 719 612 -852 -603 526 -6 -695 14 -820 99 464 -562 -161 -869 -854 835 -103 432	-539 291 -190 336 43 -597 -581 828 570 386 -542 217 -905 946 196 261 -565 101 -928 995 -241 529 249 436 111 546 -275 514 288 -68 -298 -133 -778 701 -230 47 -942 -770 396 764 -61 749 116 -909 -242 962 -591 406 962 -582 -819 766 207 -95 -225 -667 -406 722 989 828 944 498 -367 931 347 -600 -594 -819 780 756 770 -638 -506 -163 -587 -73 -660 380 -421 -438 -302 -532 619 22 57 -258 917 -479 929 954 938 -293 -684 1 -638 -431 -316 0 945 -697 -434 -748 -584 -738 594 361 32 -393 -84 24 72 -245 593 -118 -654 -221 -909 601 -509 -520 -107 -390 604 -109 -39 132 -404 -326 332 -733 831 -246 380 660 -345 -662 -517 -226 154 -735 638 -324 354 928 -172 18 486 729 -983 -541 -272 -46 -251 208 -989 353 235 421 53 -952 -613 -44 914 830 639 833 713 -711 -564 519 -776 486 780 600 -145 773 994 782 574 428 -198 874 78 -103 -871 -594 -460 867 849 -276 766 -778 76 -958 453 -162 -240 994 -511 69 -863 827 -340 -410 -353 154 260 -255 914 -375 -617 -229 979 415 821 301 -318 446 -788 -335 -372 697 711 -725 749 137 387 699 883 -395 56 180 -265 -518 -965 -502 -85 668 763 -199 -669 -712 132 -145 -29 585 135 562 -816 -198 -753 -225 289 -800 -580 -301 52 347 305 162 -815 623 875 498 815 -865 720 -470 311 574 -481 -470 293 -580 -613 719 612 -852 -603 526 -6 -695 14 -820 99 464 -562 -161 -869 -854 835 -103 432	

Passed all tests! ✓

**Đúng**

Marks for this submission: 10,00/10,00.



**Câu hỏi 8**

Đúng

Đạt điểm 10,00 trên 10,00

Cho hai số A và B. Đảo vị trí của chúng và in ra.

Bạn phải cài đặt thêm để chương trình test ở main() chạy được. Phần nhập xuất và test ở main() có sẵn và không thể sửa đổi được.

Ví dụ1:

\* Input

2 3

\* Output:

3 2

Code của các phần có sẵn như sau:

```
#include <iostream>
#include <fstream>
#include <string>
#include <cmath>
#include <vector>
#include <algorithm>
#include <stdlib.h>
```

```
using namespace std;
```

```
//Bo sung code o day. Phan ham main() co san o cho khac va khong sua doi duoc
```

```
int main()
{
    int a, b;
    cin >> a >> b;
    mySwap(a, b);
    cout << a << " " << b;

    return 0;
}
```

**For example:**

Test	Input	Result
<pre>int a, b; cin &gt;&gt; a &gt;&gt; b; mySwap(a, b); cout &lt;&lt; a &lt;&lt; " " &lt;&lt; b;</pre>	2 1	1 2

**Answer:** (penalty regime: 0 %)

Reset answer

```
1 //Bo sung code o day. Phan ham main() co san o cho khac va khong sua doi duoc
2 void mySwap(int &a, int &b) {
3     int t = a;
4     a = b;
5     b = t;
```

```
6 | }  
7 |
```

	Test	Input	Expected	Got	
✓	<pre>int a, b; cin &gt;&gt; a &gt;&gt; b; mySwap(a, b); cout &lt;&lt; a &lt;&lt; " " &lt;&lt; b;</pre>	2 1	1 2	1 2	✓
✓	<pre>int a, b; cin &gt;&gt; a &gt;&gt; b; mySwap(a, b); cout &lt;&lt; a &lt;&lt; " " &lt;&lt; b;</pre>	0 0	0 0	0 0	✓
✓	<pre>int a, b; cin &gt;&gt; a &gt;&gt; b; mySwap(a, b); cout &lt;&lt; a &lt;&lt; " " &lt;&lt; b;</pre>	12 110	110 12	110 12	✓
✓	<pre>int a, b; cin &gt;&gt; a &gt;&gt; b; mySwap(a, b); cout &lt;&lt; a &lt;&lt; " " &lt;&lt; b;</pre>	3 432	432 3	432 3	✓

Passed all tests! ✓

► SHOW/HIDE QUESTION AUTHOR'S SOLUTION (CPP)

Đúng

Marks for this submission: 10,00/10,00.

**Câu hỏi 9**

Đúng

Đạt điểm 10,00 trên 10,00

Viết hàm `void printArrow(int n, bool left)` để in một hình mũi tên, trong đó `n` là bậc của mũi tên, và `left` quy định hướng của mũi tên. Nếu `left` là `true` thì hướng của mũi tên sẽ quay sang trái, là `false` thì sẽ quay sang phải.

**For example:**

Test	Input	Result
<pre>int n, left; cin &gt;&gt; n &gt;&gt; left; printArrow(n, left);</pre>	5 0	<pre>*****   ****    ***     **      *     **    ***   **** *****</pre>
<pre>int n, left; cin &gt;&gt; n &gt;&gt; left; printArrow(n, left);</pre>	5 1	<pre>*****   ****    ***     **      *     **    ***   **** *****</pre>

**Answer:** (penalty regime: 0 %)

```
1 #include <iostream>
2 using namespace std;
3
4 void printArrow(int n, bool left) {
5     if (left) {
6         int space = n - 1;
7         for (int ast = n; ast >= 1; --ast, --space) {
8             cout << string(space, ' ') << string(ast, '*') << '\n';
9         }
10        space += 2;
11        for (int ast = 2; ast <= n; ++ast, ++space) {
12            cout << string(space, ' ') << string(ast, '*') << '\n';
13        }
14    } else {
15        int space = 0;
16        for (int ast = n; ast >= 1; --ast, space += 2) {
17            cout << string(space, ' ') << string(ast, '*') << '\n';
18        }
19        space -= 4;
20        for (int ast = 2; ast <= n; ++ast, space -= 2) {
21            cout << string(space, ' ') << string(ast, '*') << '\n';
22        }
23    }
24 }
25 }
```

	Test	Input	Expected	Got	
✓	<pre>int n, left; cin &gt;&gt; n &gt;&gt; left; printArrow(n, left);</pre>	5 0	<pre>***** **** *** ** * ** *** **** *****</pre>	<pre>***** **** *** ** * ** *** **** *****</pre>	✓
✓	<pre>int n, left; cin &gt;&gt; n &gt;&gt; left; printArrow(n, left);</pre>	5 1	<pre>***** **** *** ** * ** *** **** *****</pre>	<pre>***** **** *** ** * ** *** **** *****</pre>	✓
✓	<pre>int n, left; cin &gt;&gt; n &gt;&gt; left; printArrow(n, left);</pre>	10 0	<pre>***** ***** ***** ***** ***** **** *** ** * ** *** **** ***** ***** ***** ***** ***** ***** *****</pre>	<pre>***** ***** ***** ***** ***** **** *** ** * ** *** **** ***** ***** ***** ***** ***** ***** *****</pre>	✓

[illegible]



Câu hỏi 10

Đúng

Đạt điểm 10,00 trên 10,00

Sắp xếp là nhu cầu thường thấy khi làm việc với dữ liệu, dù là kiểu kí tự hay kiểu số.  
Viết hàm `void sort(float array[], int size, bool isAscending)` nhận tham số đầu vào là một mảng số thực có kích cỡ `size`.  
Nếu `isAscending` nhận giá trị `true`, sắp xếp mảng theo thứ tự tăng dần, ngược lại, sắp xếp mảng theo thứ tự giảm dần.

For example:

Test	Input	Result
sort(array, n, true)	4 2 3 1 6	1.00 2.00 3.00 6.00

Answer: (penalty regime: 0 %)

```
1 void sort(float array[], int size, bool isAscending) {
2     for (int i = 0; i < size-1; ++i) {
3         for (int j = i+1; j < size; ++j) {
4             if ((isAscending && array[j] < array[i]) || (!isAscending && array[j] > array[i])) {
5                 swap(array[i], array[j]);
6             }
7         }
8     }
9 }
```

	Test	Input	Expected	Got	
✓	sort(array, n, true)	4 2 3 1 6	1.00 2.00 3.00 6.00	1.00 2.00 3.00 6.00	✓
✓	sort(array, n, true)	10 10.6579176171 -7.44530488421 -15.2651157251 -1.10895469456 -18.4388600308 4.9880183654 3.94724364022 16.2997304164 -3.30393854802 5.07280092833	-18.44 -15.27 -7.45 -3.30 -1.11 3.95 4.99 5.07 10.66 16.30	-18.44 -15.27 -7.45 -3.30 -1.11 3.95 4.99 5.07 10.66 16.30	✓

	Test	Input	Expected	Got	
✓	sort(array, n, false)	20 -14.9901942752 19.9444193075 5.63863739243 -1.20820585544 -8.46026052206 21.5954131858 4.20688426553 3.13787226372 9.28173072626 10.8288242252 11.998278715 -16.6635788226 4.88816372857 5.06370205692 4.94745541204 -4.11308394768 -8.87581617955 -1.79038201065 -12.2754818716 8.15407638295	21.60 19.94 12.00 10.83 9.28 8.15 5.64 5.06 4.95 4.89 4.21 3.14 -1.21 -1.79 -4.11 -8.46 -8.88 -12.28 -14.99 -16.66	21.60 19.94 12.00 10.83 9.28 8.15 5.64 5.06 4.95 4.89 4.21 3.14 -1.21 -1.79 -4.11 -8.46 -8.88 -12.28 -14.99 -16.66	✓
✓	sort(array, n, false)	25 14.8934039584 -0.243600485587 -3.08316266829 -7.30696165787 8.28652890274 -0.593930434044 -27.669322874 2.54311532448 0.530580383512 -3.11106906328 -5.35475589182 -13.5145275252 7.95320983418 -12.4114171679 -24.1169997076 13.2799786583 -12.8804061711 -13.0915248803 4.28232477424 4.01351356736 -8.01188399809 23.8761836184 4.42337570597 -4.00886028421 -6.67621010736	23.88 14.89 13.28 8.29 7.95 4.42 4.28 4.01 2.54 0.53 -0.24 -0.59 -3.08 -3.11 -4.01 -5.35 -6.68 -7.31 -8.01 -12.41 -12.88 -13.09 -13.51 -24.12 -27.67	23.88 14.89 13.28 8.29 7.95 4.42 4.28 4.01 2.54 0.53 -0.24 -0.59 -3.08 -3.11 -4.01 -5.35 -6.68 -7.31 -8.01 -12.41 -12.88 -13.09 -13.51 -24.12 -27.67	✓

Passed all tests! ✓

Đúng

Marks for this submission: 10,00/10,00.

Trở lại Khoá học