

Trạng thái	Đã xong
Bắt đầu vào lúc	Thứ Sáu, 18 tháng 4 2025, 7:16 PM
Kết thúc lúc	Thứ Sáu, 18 tháng 4 2025, 7:34 PM
Thời gian thực hiện	17 phút 49 giây

Câu hỏi 1

Đúng

Đạt điểm 1,00

Mô tả tiếng Việt:

Kho lưu trữ của tổ chức SCP chứa hàng loạt các vật thể dị thường. Mỗi vật thể dị thường được lưu trữ dưới struct **SCP** với các thông tin như sau:

- id**: kiểu int, là mã định danh (hay mã vật thể), phân biệt giữa các vật thể với nhau.
- objClass**: kiểu int, là phân loại của vật thể đó.
- speConProcedures**: kiểu string, mô tả quy trình quản thúc đặc biệt của vật thể đó.
- description**: kiểu string, mô tả về các đặc điểm của vật thể đó.
- addendums**: kiểu string*, là một tập hợp của các phụ lục đính kèm, mô tả các thông tin bổ sung cho vật thể đó.
- numAddendums**: kiểu int, là số lượng phụ lục đính kèm.

Hiện thực struct **SCP** với các yêu cầu trên.

Ghi chú: (Các) thư viện **iostream** và **string** đã được khai báo, và **namespace std** đã được sử dụng.

English version:

SCP Foundation's classified archives consist of records of paranormal objects. The information of each object is stored using struct **SCP** with following requirements:

- id**: integer, the identifier (or item number) of the object.
- objClass**: integer, the object class.
- speConProcedures**: string, specification of the object's special containment procedures.
- description**: string, description of the object.
- addendums**: string*, array of addendums describing additional information about the object.
- numAddendums**: integer, the number of addendums attached.

Declare struct **SCP** with mentioned requirements.

Note: Libraries **iostream** and **string** have been imported, and **namespace std** has been used.

Answer: (penalty regime: 0 %)

Reset answer

```
1 struct SCP {
2     // Student answer
3     int id;
4     int objClass;
5     string speConProcedures;
6     string description;
7     string* addendums;
8     int numAddendums;
9 };
```



Passed all tests! ✓

Câu hỏi 2

Đúng

Đạt điểm 1,00

Mô tả tiếng Việt:

Cho struct **SCP** lưu trữ thông tin các vật thể dị thường được mô tả như sau:

```
struct SCP {
    int id;
    int objClass;
    string speConProcedures;
    string description;
    string* addendums;
    int numAddendums;
};
```

Hiện thực một hàm với prototype sau:

```
void addAddendum(SCP &obj, string addendum);
```

Hàm thực hiện bổ sung một phụ lục **addendum** vào cuối danh sách phụ lục (addendums) của **obj**.

Ghi chú: (Các) thư viện **iostream** và **string** đã được khai báo, và namespace **std** đã được sử dụng.

English version:

Struct **SCP** used to store information about paranormal objects is declared as below:

```
struct SCP {
    int id;
    int objClass;
    string speConProcedures;
    string description;
    string* addendums;
    int numAddendums;
};
```

Implement the function with the following prototype:

```
void addAddendum(SCP &obj, string addendum);
```

The function appends a new **addendum** into the array of addendums of the object **obj**.

Note: Libraries **iostream** and **string** have been imported, and namespace **std** has been used.

For example:

Test	Result

<pre>string* addendums = new string[1]; addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired."; SCP obj {55, 2, "Object is kept within a five (5) by five (5) by two point five (2.5) meter square room.", "All known information about the object is that it is not round.", addendums, 1}; addAddendum(obj, "Document #055-2: Report of Dr. John Marachek\nSurvey team #19-055-127BXE was successfully able to enter SCP- 055's container and ascertain the appearance."); cout << obj.addendums[1]; delete [] obj.addendums;</pre>	<p>Document #055-2: Report of Dr. John Marachek Survey team #19- 055-127BXE was successfully able to enter SCP-055's container and ascertain the appearance.</p>
---	--

Answer: (penalty regime: 0 %)

Reset answer

```
1 void addAddendum(SCP &obj, string addendum) {
2     // Student answer
3     string *arr = new string[obj.numAddendums + 1];
4     for(int i = 0; i < obj.numAddendums; ++i)
5         arr[i] = obj.addendums[i];
6
7     arr[obj.numAddendums] = addendum;
8
9     delete [] obj.addendums;
10
11     obj.addendums = arr;
12     obj.numAddendums++;
13 }
```

Passed all tests! ✓