Trạng thái	Đã xong
Bắt đầu vào lúc	Thứ Tư, 16 tháng 4 2025, 5:06 PM
Kết thúc lúc	Thứ Tư, 16 tháng 4 2025, 5:20 PM
Thời gian thực hiện	14 phút 32 giây

Câu hỏi **1**

Đúng

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:

```
SCP createSCP(int id, int objClass, string speConProcedures, string description, string* addendums, int numAddendums);
```

Hàm có chức năng khởi tạo instance của struct SCP với các thông tin đã truyền vào, sau đó trả về instance đó.

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:

```
SCP createSCP(int id, int objClass, string speConProcedures, string description, string* addendums, int numAddendums);
```

The function returns a new instance of struct `SCP` with the passed parameters.

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

For example:

Test	Result

string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
SCP obj = createSCP(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, 2);
 	
for (int i = 0; i < 2; i++) cout << obj.addendums[i] << "\n";
 	
delete [] addendums;
	Document #055-1: An Analysis of SCP-055
	The author puts forward the hypothesis that SCP-055 was never formally acquired.
	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.

Test	Result

string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
SCP obj = createSCP(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, 2);
 	
cout << obj.id;
 	
delete [] addendums;
	55

string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
SCP obj = createSCP(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, 2);
 	
cout << obj.objClass;
 	
delete [] addendums;
	2

string* addendums = new string[2]; addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired."; addendums[1] = "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."; SCP obj = createSCP(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, 2); cout << obj.speConProcedures;	Object is kept within a five (5) by five (5) by two point five (2.5) meter square room.

string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
SCP obj = createSCP(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, 2);
 	
cout << obj.description;
 	
delete [] addendums;
	All known information about the object is that it is not round.

string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
SCP obj = createSCP(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, 2);
 	
for (int i = 0; i < obj.numAddendums; i++) cout << obj.addendums[i] << "\n";
 	
delete [] addendums;
	Document #055-1: An Analysis of SCP-055
	The author puts forward the hypothesis that SCP-055 was never formally acquired.
	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.

Test	Result

string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
SCP obj = createSCP(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, 2);
 	
for (int i = 0; i < obj.numAddendums; i++) cout << obj.addendums[i] << "\n";
 	
delete [] addendums;
	Document #055-1: An Analysis of SCP-055
	The author puts forward the hypothesis that SCP-055 was never formally acquired.
	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.

string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
SCP obj = createSCP(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, 2);
 	
for (int i = 0; i < obj.numAddendums; i++) cout << obj.addendums[i] << "\n";
 	
delete [] addendums;
	Document #055-1: An Analysis of SCP-055
	The author puts forward the hypothesis that SCP-055 was never formally acquired.
	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.

string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
SCP obj = createSCP(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, 2);
 	
for (int i = 0; i < obj.numAddendums; i++) cout << obj.addendums[i] << "\n";
 	
delete [] addendums;
	Document #055-1: An Analysis of SCP-055
	The author puts forward the hypothesis that SCP-055 was never formally acquired.
	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.

string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
SCP obj = createSCP(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, 2);
 	
for (int i = 0; i < obj.numAddendums; i++) cout << obj.addendums[i] << "\n";
 	
delete [] addendums;
	Document #055-1: An Analysis of SCP-055
	The author puts forward the hypothesis that SCP-055 was never formally acquired.
	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.

Answer: (penalty regime: 0 %)

Reset answer

```
SCP createSCP(int id, int objClass, string speConProcedures, string description, string* addendums,
                                                 int numAddendums) {
    // Khởi tạo một đối tượng SCP mới
    SCP newSCP;

    // Gán các giá trị cơ bản
    newSCP.id = id;                              // Gán mã định danh
    newSCP.objClass = objClass;                  // Gán loại đối tượng
    newSCP.objClass = objClass;                // Gán loại đối tượng
    newSCP.speConProcedures = speConProcedures;  // Gán thủ tục chứa giữ đặc biệt
    newSCP.description = description;            // Gán mô tả

    // Cấp phát bộ nhớ cho mảng addendums
    newSCP.numAddendums = numAddendums;          // Lưu số lượng phụ lục
    newSCP.addendums = new string[numAddendums]; // Cấp phát mảng động mới để lưu các phụ lục

    // Sao chép từng chuỗi phụ lục từ mảng đầu vào sang mảng mới
    for (int i = 0; i < numAddendums; i++) {
        newSCP.addendums[i] = addendums[i];      // Sao chép nội dung phụ lục thứ i
    }
    // Trả về đối tượng SCP đã được khởi tạo đầy đủ
    return newSCP;
}
```

Passed all tests! ✓

Câu hỏi **2**
Đúng

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:

```
string convertToString(SCP obj);
```

Hàm trả về một string duy nhất, với format chuẩn được quy định bởi tổ chức:

```
Item #: SCP-<Mã định danh, thêm 0 vào trước nếu chưa đủ 3 chữ số>
Object Class: <Safe nếu objClass là 0, Euclid nếu là 1, Keter nếu là 2>
Special Containment Procedures: <Quy trình quản thúc đặc biệt>
Description: <Mô tả vật thể>
<Phụ lục 0>
<Phụ lục 1>
...
<Phụ lục n>
```

Tham khảo ví dụ để hiểu về format đầu ra.

Ghi chú: (Các) thư viện `iostream`, `sstream` 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:

```
string convertToString(SCP obj);
```

The function returns a string with the following format:

```
Item #: SCP-<Item number, zero-left-padded until it has 3 or more digits.>
Object Class: <Safe if objClass is 0, Euclid if 1, Keter if 2>
Special Containment Procedures: <The specification of the object's special containment procedures.>
Description: <The object's description.>
<Addendum 0>
<Addendum 1>
...
<Addendum n>
```

Refer to the example to understand the format.

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

For example:

Test	Result

string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
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, 2};
 	
cout << convertToString(obj);
 	
delete [] addendums;
	Item #: SCP-055
	Object Class: Keter
	Special Containment Procedures: Object is kept within a five (5) by five (5) by two point five (2.5) meter square room.
	Description: All known information about the object is that it is not round.
	Document #055-1: An Analysis of SCP-055
	The author puts forward the hypothesis that SCP-055 was never formally acquired.
	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.

Test	Result
SCP obj {2, 1, "SCP-002 is to remain connected to a suitable power supply at all times.", "SCP-002 resembles a tumorous, fleshy growth with a volume of roughly 60 cubed meter.", nullptr, 0}; cout << convertToString(obj);	Item #: SCP-002 Object Class: Euclid Special Containment Procedures: SCP-002 is to remain connected to a suitable power supply at all times. Description: SCP-002 resembles a tumorous, fleshy growth with a volume of roughly 60 cubed meter.
SCP obj {500, 0, "SCP-500 must be stored in a cool and dry place away from bright light.", "SCP-500 is a small plastic can which at the time of writing contains forty-seven (47) red pills.", nullptr, 0}; cout << convertToString(obj);	Item #: SCP-500 Object Class: Safe Special Containment Procedures: SCP-500 must be stored in a cool and dry place away from bright light. Description: SCP-500 is a small plastic can which at the time of writing contains forty-seven (47) red pills.
SCP obj {1487, 1, "SCP-1487 is to be kept in Humanoid Containment Chamber #3821 at Site 23.", "SCP-1487 is a female humanoid of Filipino descent. The entity is fifteen (15) years old and measures 1.7m in height and 70kg in weight.", nullptr, 0}; cout << convertToString(obj);	Item #: SCP-1487 Object Class: Euclid Special Containment Procedures: SCP-1487 is to be kept in Humanoid Containment Chamber #3821 at Site 23. Description: SCP-1487 is a female humanoid of Filipino descent. The entity is fifteen (15) years old and measures 1.7m in height and 70kg in weight.

string* addendums = new string[1];
addendums[0] = "Site 81/715 Protocol: [ACCESS RESTRICTED]\nPlease Access Using 715/5 Authorization Code";
 	
SCP obj {715, 0, "SCP-715 is to be contained at its point of origin within the ---- City Mall in ----, Ohio.", "SCP-715 is a \"Take Your Own Photo\" fotomat model photo booth manufactured by the Sony corporation in 1972.", addendums, 1};
 	
cout << convertToString(obj);
 	
delete [] addendums;
	Item #: SCP-715
	Object Class: Safe
	Special Containment Procedures: SCP-715 is to be contained at its point of origin within the ---- City Mall in ----, Ohio.
	Description: SCP-715 is a "Take Your Own Photo" fotomat model photo booth manufactured by the Sony corporation in 1972.
	Site 81/715 Protocol: [ACCESS RESTRICTED]
	Please Access Using 715/5 Authorization Code

string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
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, 2};
 	
cout << convertToString(obj);
 	
delete [] addendums;
	Item #: SCP-055
	Object Class: Keter
	Special Containment Procedures: Object is kept within a five (5) by five (5) by two point five (2.5) meter square room.
	Description: All known information about the object is that it is not round.
	Document #055-1: An Analysis of SCP-055
	The author puts forward the hypothesis that SCP-055 was never formally acquired.
	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.

SCP obj {2, 1, "SCP-002 is to remain connected to a suitable power supply at all times.", "SCP-002 resembles a tumorous, fleshy growth with a volume of roughly 60 cubed meter.", nullptr, 0}; cout << convertToString(obj);	Item #: SCP-002 Object Class: Euclid Special Containment Procedures: SCP-002 is to remain connected to a suitable power supply at all times. Description: SCP-002 resembles a tumorous, fleshy growth with a volume of roughly 60 cubed meter.

Test	Result

`SCP obj {500, 0, "SCP-500 must be stored in a cool and dry place away from bright light.", "SCP-500 is a small plastic can which at the time of writing contains forty-seven (47) red pills.", nullptr, 0};`
`cout << convertToString(obj);`
	Item #: SCP-500
	Object Class: Safe
	Special Containment Procedures: SCP-500 must be stored in a cool and dry place away from bright light.
	Description: SCP-500 is a small plastic can which at the time of writing contains forty-seven (47) red pills.

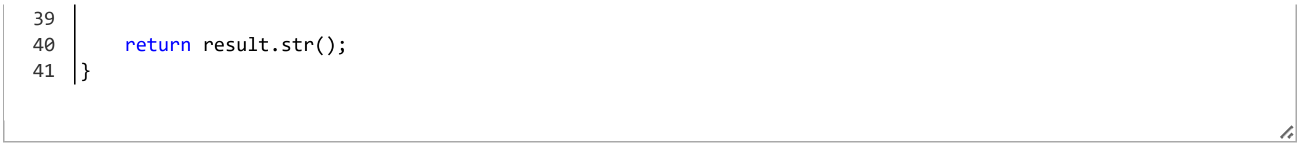
`SCP obj {1487, 1, "SCP-1487 is to be kept in Humanoid Containment Chamber #3821 at Site 23.", "SCP-1487 is a female humanoid of Filipino descent. The entity is fifteen (15) years old and measures 1.7m in height and 70kg in weight.", nullptr, 0};`
`cout << convertToString(obj);`
	Item #: SCP-1487
	Object Class: Euclid
	Special Containment Procedures: SCP-1487 is to be kept in Humanoid Containment Chamber #3821 at Site 23.
	Description: SCP-1487 is a female humanoid of Filipino descent. The entity is fifteen (15) years old and measures 1.7m in height and 70kg in weight.

`string* addendums = new string[1];`
`addendums[0] = "Site 81/715 Protocol: [ACCESS RESTRICTED]\nPlease Access Using 715/5 Authorization Code";`
 	
`SCP obj {715, 0, "SCP-715 is to be contained at its point of origin within the ---- City Mall in ----, Ohio.", "SCP-715 is a \"Take Your Own Photo\" fotomat model photo booth manufactured by the Sony corporation in 1972.", addendums, 1};`
 	
`cout << convertToString(obj);`
 	
`delete [] addendums;`
	Item #: SCP-715
	Object Class: Safe
	Special Containment Procedures: SCP-715 is to be contained at its point of origin within the ---- City Mall in ----, Ohio.
	Description: SCP-715 is a "Take Your Own Photo" fotomat model photo booth manufactured by the Sony corporation in 1972.
	Site 81/715 Protocol: [ACCESS RESTRICTED]
	Please Access Using 715/5 Authorization Code

Answer: (penalty regime: 0 %)

Reset answer

```
string convertToString(SCP obj) {
    // Sử dụng stringstream để xây dựng chuỗi kết quả
    stringstream result;

    // Thêm mã định danh (zero-padded nếu cần)
    result << "Item #: SCP-";
    if (obj.id < 10) {
        result << "00" << obj.id;
    } else if (obj.id < 100) {
        result << "0" << obj.id;
    } else {
        result << obj.id;
    }
    result << endl;

    // Thêm loại đối tượng
    result << "Object Class: ";
    if (obj.objClass == 0) {
        result << "Safe";
    } else if (obj.objClass == 1) {
        result << "Euclid";
    } else if (obj.objClass == 2) {
        result << "Keter";
    }
    result << endl;

    // Thêm thủ tục quản thúc đặc biệt
    result << "Special Containment Procedures: " << obj.speConProcedures << endl;

    // Thêm mô tả
    result << "Description: " << obj.description;

    // Thêm các phụ lục (nếu có)
    if (obj.numAddendums > 0 && obj.addendums != nullptr) {
        for (int i = 0; i < obj.numAddendums; i++) {
            result << endl << obj.addendums[i];
        }
    }
```



```

    return result.str();
}
```

Passed all tests! ✓

Câu hỏi **3**

Đúng

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;
};
```

Một đối tượng dị thường là SCP-038 có khả năng tạo bản sao của các vật thể dị thường khác, ví dụ như SCP-500 (thuốc vạn ứng). Khi đối tượng gốc bị tiêu thụ/phá hủy, bản sao vẫn tồn tại độc lập.

Hiện thực một hàm với prototype sau:

```
SCP* cloneSCP(SCP* original);
```

Hàm có chức năng tạo một bản sao con trỏ loại SCP chứa các thông tin như đối tượng gốc original, sau đó trả về con trỏ đó. Lưu ý về shallow copy và deep copy.

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;
};
```

The paranormal object SCP-038 has the ability to clone other objects, such as SCP-500 (panacea). The clone is independent of the original, and the clone will remain the same whether the original is destroyed or modified.

Implement the function with the following prototype:

```
SCP* cloneSCP(SCP* original);
```

The function returns a pointer of the new cloned instance of the original object `original`. Choose the appropriate copy method.

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

For example:

Test	Result

string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
SCP* obj = new SCP {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, 2};
 	
SCP* newObj = cloneSCP(obj);
 	
obj->id = 1;
cout << obj->id << " " << newObj->id << "\n";
 	
delete [] obj->addendums;
delete obj;
 	
delete [] newObj->addendums;
delete newObj;
	1 55

string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
SCP* obj = new SCP {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, 2};
 	
SCP* newObj = cloneSCP(obj);
 	
obj->description = "The object cannot be described.";
cout << obj->description << " " << newObj->description << "\n";
 	
delete [] obj->addendums;
delete obj;
 	
delete [] newObj->addendums;
delete newObj;
	The object cannot be described. All known information about the object is that it is not round.

string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
SCP* obj = new SCP {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, 2};
 	
SCP* newObj = cloneSCP(obj);
 	
obj->addendums[0] = "Access authority required.";
cout << obj->addendums[0] << " " << newObj->addendums[0] << "\n";
 	
delete [] obj->addendums;
delete obj;
 	
delete [] newObj->addendums;
delete newObj;
	Access authority required. Document #055-1: An Analysis of SCP-055
	The author puts forward the hypothesis that SCP-055 was never formally acquired.

Test	Result

string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
SCP* obj = new SCP {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, 2};
 	
SCP* newObj = cloneSCP(obj);
 	
delete [] obj->addendums;
delete obj;
 	
cout << newObj->id << "\n";
 	
delete [] newObj->addendums;
delete newObj;
	55

{
string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
SCP* obj = new SCP {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, 2};
 	
SCP* newObj = cloneSCP(obj);
 	
delete [] obj->addendums;
delete obj;
 	
cout << newObj->addendums[0] << "\n";
 	
delete [] newObj->addendums;
delete newObj;
}
	Document #055-1: An Analysis of SCP-055
	The author puts forward the hypothesis that SCP-055 was never formally acquired.

string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
SCP* obj = new SCP {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, 2};
 	
SCP* newObj = cloneSCP(obj);
 	
obj->id = 1;
cout << obj->id << " " << newObj->id << "\n";
 	
delete [] obj->addendums;
delete obj;
 	
delete [] newObj->addendums;
delete newObj;
	1 55

Test	Result

string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
SCP* obj = new SCP {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, 2};
 	
SCP* newObj = cloneSCP(obj);
 	
obj->description = "The object cannot be described.";
cout << obj->description << " " << newObj->description << "\n";
 	
delete [] obj->addendums;
delete obj;
 	
delete [] newObj->addendums;
delete newObj;
	The object cannot be described. All known information about the object is that it is not round.

string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
SCP* obj = new SCP {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, 2};
 	
SCP* newObj = cloneSCP(obj);
 	
obj->addendums[0] = "Access authority required.";
cout << obj->addendums[0] << " " << newObj->addendums[0] << "\n";
 	
delete [] obj->addendums;
delete obj;
 	
delete [] newObj->addendums;
delete newObj;
	Access authority required. Document #055-1: An Analysis of SCP-055
	The author puts forward the hypothesis that SCP-055 was never formally acquired.

string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
SCP* obj = new SCP {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, 2};
 	
SCP* newObj = cloneSCP(obj);
 	
delete [] obj->addendums;
delete obj;
 	
cout << newObj->id << "\n";
 	
delete [] newObj->addendums;
delete newObj;
	55

Test	Result

{
string* addendums = new string[2];
addendums[0] = "Document #055-1: An Analysis of SCP-055\nThe author puts forward the hypothesis that SCP-055 was never formally acquired.";
addendums[1] = "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.";
 	
SCP* obj = new SCP {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, 2};
 	
SCP* newObj = cloneSCP(obj);
 	
delete [] obj->addendums;
delete obj;
 	
cout << newObj->addendums[0] << "\n";
 	
delete [] newObj->addendums;
delete newObj;
}
	Document #055-1: An Analysis of SCP-055
	The author puts forward the hypothesis that SCP-055 was never formally acquired.

Answer: (penalty regime: 0 %)

Reset answer

```
SCP* cloneSCP(SCP* original) {
    // Kiểm tra con trỏ đầu vào có hợp lệ hay không
    if (original == nullptr) {
        return nullptr;
    }

    // Cấp phát bộ nhớ cho đối tượng SCP mới
    SCP* clone = new SCP;

    // Sao chép các thông tin cơ bản
    clone->id = original->id;
    clone->objClass = original->objClass;
    clone->speConProcedures = original->speConProcedures;
    clone->description = original->description;

    // Xử lý các phụ lục
    clone->numAddendums = original->numAddendums;

    // Nếu có phụ lục, cần thực hiện deep copy
    if (original->numAddendums > 0 && original->addendums != nullptr) {
        // Cấp phát mảng mới cho các phụ lục
        clone->addendums = new string[original->numAddendums];

        // Sao chép từng phụ lục
        for (int i = 0; i < original->numAddendums; i++) {
            clone->addendums[i] = original->addendums[i];
        }
    } else {
        // Nếu không có phụ lục, đặt con trỏ là nullptr
        clone->addendums = nullptr;
    }

    return clone;
}
```

Passed all tests! ✓