

Forum:[Hack Forums](#)

Thread: [Sleep\(time\)](#)

Exodus. - February 21, 2015, 10:01 pm

Code

```
int exitFunction()
```

```
{
    string str;

    cout << "Enter a line of text: ";
    getline(cin, str);
    cout << "You entered: " << str << endl;
    if (str == "Exit")
    {
        return 0;
    }
    else
        Sleep(1000);
}
```

Basically this means unless you enter Exit it will wait 1000 seconds before returning to int main. But instead it exits.

Tell me if you want me to post my full source code.

g4143 - February 21, 2015, 10:21 pm

How does this compile? All the function paths should be returning an integer.

Exodus. - February 21, 2015, 10:33 pm

Code // Testing Different types of declared functions.

```
#include "stdafx.h"
#include "targetver.h"

#include <string>
#include <iostream>
#include <Windows.h>
using namespace std;
```

```
int calculation()
{
    int x;
    int y;

    cout << "Enter your first number: \n";

    cin >> x;

    cout << "Enter your second number: \n";

    cin >> y;
```

```

    system("cls");

    cout << "The product is: " << x * y << endl;

    return 0;
}

int exitFunction()
{
    string str;

    cout << "Enter a line of text: ";
    getline(cin, str);
    cout << "You entered: " << str << endl;
    if (str == "Exit")
    {
        return 0;
    }
    else
        Sleep(1000);
}

int main()
{

    cout << "Multiply: ";

    calculation();

    system("pause");
    system("cls");

    exitFunction();

    cin.get();
    return 0;
}My full source code.

```

Kantos - February 21, 2015, 10:33 pm

You need a loop or a recursive call in order for it to jump back to user input.

Exodus. - February 21, 2015, 10:39 pm

No, I just want it to be idle on the screen for 1000 seconds.

Kantos - February 21, 2015, 10:53 pm

Oh, I misread. Sleep takes milliseconds as an argument, so Sleep(1000) sleeps for 1 second and not 1000.

Also your exit function should be void and not int since it's not returning anything useful.

(Feb 21, 2015 11:21 PM)g4143 Wrote: [\[CITE\]](#) How does this compile? All the function paths should be returning an integer.

C/C++ don't consider it an error for reasons of backward compatibility with pre-ANSI standards

since not all compilers had void. They'll still warn you about it though.

Exodus. - February 21, 2015, 11:41 pm

(Feb 21, 2015 11:53 PM)Kantos Wrote: [\[CITE\]](#) Oh, I misread. Sleep takes milliseconds as an argument, so Sleep(1000) sleeps for 1 second and not 1000.

Also your exit function should be void and not int since it's not returning anything useful.

C/C++ don't consider it an error for reasons of backward compatibility with pre-ANSI standards since not all compilers had void. They'll still warn you about it though.

Yeah I knew it was void but I kept getting an error so instead I tried int.

I tried again and I get the following errors on debug: Spoiler [\(Click to View\)](#)

0xDEAD10CC - February 21, 2015, 11:44 pm

elementaL-CSGO... Read what the error says. It should be blatantly obvious what the issue is, and what the fix would be, if you keep the function return as 'void'.

Exodus. - February 22, 2015, 12:29 am

Okay, since I was having a difficult time with the string system I decided to this using integers a,b,c. In this case a=1 b=0 c=userinput. This worked out extremely well.

Considering I'm only on part 19 of this video series I've been watching, do you guys think this is well done? I made this program/script to test different types of functions other than void (which in this case I used int).

Code // Testing Different types of declared functions.

```
#include "stdafx.h"
#include "targetver.h"
```

```
#include <string>
#include <iostream>
#include <Windows.h>
using namespace std;
```

```
int calculation()
{
    int x;
    int y;

    cout << "Enter your first number: \n";

    cin >> x;

    cout << "Enter your second number: \n";

    cin >> y;

    system("cls");

    cout << "The product is: " << x * y << endl;

    return 0;
```

```

}

int exitFunction()
{
    int a = 1;
    int b = 0;
    int c;

    cout << "Enter 1 to Exit and 0 to Stay: ";
    cin >> c;

    if (c == 1)
        return 0;
    else
    {
        Sleep(9000);
        cout << "Success! We have waited 9 seconds!\n";
    }
}

int main()
{
    cout << "Multiply: ";

    calculation();

    system("pause");
    system("cls");

    exitFunction();

    cin.get();
    return 0;
}

```

0xDEAD10CC - February 22, 2015, 1:30 am

No, this is not very good..

calculation() is not a descriptive identifier for a function that displays a product. You have it return an int, but you return 0 which is meaningless.

exitFunction() only returns a value if c == 1, but not all code branches, which is wrong.

And the system() function calls are horrible.

Part 19 and this is the code you can write? Stop with the video series and grab a real book or take a real course.

Exodus. - February 22, 2015, 3:22 am

What do you use instead of system() function calls?

Also c only needs to return 1, if they enter 0 it still works since else.

0xDEAD10CC - February 22, 2015, 12:04 pm

(Feb 22, 2015 04:22 AM)elementaL-CSGO Wrote: [\[CITE\]](#) What do you use instead of system() function calls?

Also c only needs to return 1, if they enter 0 it still works since else.

Why are you trying to tell me what "C" needs? You don't know what you're talking about at all. This looks like C++ too, not C. You're missing a return statement from that code branch... There's lots of things that "work" in C, but that doesn't mean they are proper or are not undefined behavior.

Code warning: control reaches end of non-void function [-Wreturn-type]
Furthermore, you declare a, and b, which aren't even used in that function...

Exodus. - February 22, 2015, 12:40 pm

C is an integer in my script int c. I haven't mentioned anything about C.

Also I modified the script code. Check it out now.
Code // Testing Different types of declared functions.

```
#include "stdafx.h"
#include "targetver.h"
```

```
#include <string>
#include <iostream>
#include <Windows.h>
using namespace std;
```

```
void calculation()
{
    int x;
    int y;

    cout << "Enter your first number: \n";

    cin >> x;

    cout << "Enter your second number: \n";

    cin >> y;

    system("cls");

    cout << "The product is: " << x * y << endl;
}
```

```
void exitFunction()
{
    cout << "Enter 1 to Exit or 0 to Stay.\n";

    int z;

    cin >> z;
```

```

if (z == 1)
{
    cout << "You chose to leave. \n";
    system("pause");
    exit(1);
}
if (z == 0)
{
    Sleep(9000);
}
else
    cout << "You did not enter one or zero.";
}

int main()
{

    cout << "Multiply: ";

    calculation();

    system("pause");
    system("cls");

    exitFunction();
    cout << "You chose to stay. \n";

    cin.get();
    return 0;
}

```

0xDEAD10CC - February 22, 2015, 1:16 pm

You just made it even worse... Stop learning anything more until you understand basic control structures and variables.

Quote:Also c only needs to return 1,

'c' doesn't return anything, the function does, and only having to return 1 is an entirely false statement...

Exodus. - February 22, 2015, 2:03 pm

I didn't make it worse. It works perfectly fine.