# CS 410 Project Two Security Report Template

Janera Dobson

CS-410-T3249

## Instructions

Fill in the table in step one. In steps two and three, replace the bracketed text with your answer in your own words.

## Identify where multiple security vulnerabilities are present within the blocks of C++ code. You may add columns and extend this table as you see fit.

| **Block of C++ Code** | **Identified Security Vulnerability** |
| --- | --- |
| string username = "";  string password = "";  string client = ""; | Environment variables |
| void CheckUserPermissionAccess() {  cout << "Enter your username: " << endl;  cin >> username;    cout << "Enter password: " << endl;  cin >> password;    if(password == "22420"){  cout <<"You chose" << endl;  } else {  cout << "Invalid Password. Please try again" << endl;  }  } | User privileges  Missing authentication for critical function  Missing authorization  Validation and Verification  Weak password  Sensitive data exposure (user input in format strings)  Insufficient testing  Lack of security |
| int main() {    cout << "Created by: Janera Dobson" << endl;    cout << "Hello, Welcome to our Investment Company!" << endl;    CheckUserPermissionAccess();    DisplayInfo();    ChangeCustomerChoice();  } | Insecure coding  Design flaw |

## Explain the *security vulnerabilities* that are found in the blocks of C++ code.

The first block of security vulnerabilities I found was within my environmental variables. They were vulnerable to overflows and would probably represent significant attacks on the surface. This comes about due to explicitly and implicitly trusting data from outside and thus can be controlled by the attacker. Subsequently, there is a risk of assuming that the environment variables provide restricted visibility, which they do not always do. Therefore, when I passed the sensitive information of what password must be required, other users on the system might see the environment variables' content and thus grab this sensitive information.

The second block of security vulnerabilities I found was within my CheckUserPermissionAccess method. Those specific vulnerabilities were user privileges, missing authentication for critical function, absent authorization, validation and verification, weak password, sensitive data exposure (user input in format strings), insufficient testing, and lack of security. All these vulnerabilities tie into one another because I coded improper access control, had weak credential management (for instance, the use of hard-coded passwords), issues with privileges, and security misconfiguration. These are security vulnerabilities because they determine the impact of whether or not a security breach is very high. Hence, the core problem with access control vulnerabilities is the system's incorrect security configuration and component access controls. So, as a result, I risked my program becoming vulnerable, an easy target for attackers. So, when an attacker takes advantage of my coding errors, password security is compromised, the hacker gains all access to control the system and device. Then exploiting security vulnerabilities through malicious code execution and can gain access to sensitive information.

The third block of security vulnerabilities occurred in my main method. I designed a program with insecure coding and design flaw. Insecure coding is a security vulnerability because of the number of codes I had hard-coded with credentials and other important information. Also, that there was not even a single layered-defense coded to be sure a user truly validated and authenticated themselves to the system. The design flaw for my program was an attacker's dream. All the information coded was pretty much handing over all rights to the program and system. I exposed my program to injection flaws, no authentication, insecure direct object references, no security configuration, no function level access control. All of which is vital to hold the integrity and authenticity of a program.

## Describe *recommendations* for how the security vulnerabilities can be fixed.

How I recommend the above security vulnerabilities to be fixed is by establishing a software design requirement. In doing so, you will be sure that your program has everything thing it needs to be secure and coded appropriately. You will also be aware of the things you are subjecting your program to without doing more or less than what was required.

This ties into using a coding standard to produce your program. Coding standards will help in the development of the software program to reduce errors and security vulnerabilities. If done right, codes would be easily followed, consistent, and easily maintained. Therefore, knowing why you are coding the program using the reference of your software design requirements goes a long way because understanding the intent of the program will allow you to prioritize your codes and code efficiently.

Testing your software is key to detecting software failures, so once discovered, they can be corrected. With the right intent of testing, you will discover what does not function correctly under specific conditions. Not only will it discover what does not function properly, but any errors, gaps, or missing requirements versus the actual requirements of the program is found.

Dynamic memory allocation to allocate memory is another recommendation I have. Allocating memory would give the flexibility to allocate and deallocate memory whenever it is needed or not. Therefore, assigning memory space during runtime or execution time will dictate the amount of memory needed for the program beforehand. Subsequently, dynamically allocating memory allows the program to use memory space more efficiently.

Lastly, having clean and readable codes is essential for others to understand, read, and add to your codes easily. Considering when coding, we are writing codes for human readability; we must write them in a way we understand and add comments so that if a question does arise, the answer to what the line of code's purpose is is explained. Thus, when a change is required to add features, fix bugs, or change codes, it is done without confusion.

Now how I would incorporate my recommendations is by first removing my hard-coded password. Then set up a function that tests whether or not the password passed meets the requirements I coded to be sure access is granted to the right user. Next, dynamically allocate the array of passwords to ensure only the required memory necessary is being used during execution. I also coded an extra password test to check the password credentials since I did not beforehand and then freed up the memory I allocated for the password. For clean and readable codes, I commented throughout my program and made sure the user was prompted to what was the next option the program required; this way, they understood the program's flow and selected the correct response to move on. The last and final thing I did was ended a program after the main method with a bool test that made sure the credentials coded within the program met the exact format of the password I asked for. The password required at least one uppercase letter, one lowercase letter, and at least one number to ensure my program ran with maximum security.

//

// main.cpp

// FinalProject.cpp

//

// Created by Janera Dobson on 2/19/21.

//

/\*

#include <iostream>

#include <string>

using namespace std;

string username = "";

string password = "";

string client = "";

void DisplayInfo() {

cout << "What would you like to do?" << endl; // needed another line for readability

cout << "--------------- MENU ---------------" << endl;

cout << " DISPLAY the client list (enter 1)" << endl;

cout << " CHANGE a clients choice (enter 2)" << endl;

cout << " Exit program.. (enter 3)" << endl;

cout << "------------------------------------" << endl; //needed a few more lines for readability

}

//Function had serious security vulnerabilities

//Anyone could have eventually hacked my system becuase password was very easy and not authenticated correctly

//Therefore, I had to prompt the user twice to be sure they met the password requirements, check credentials with if statement and if not to re-enter until correct

// Also, I used delete[] to allocate space to memory

void CheckUserPermissionAccess() {

cout << "Enter your username: " << endl;

cin >> username;

cout << "Enter password: " << endl;

cin >> password;

if(password == "22420"){

cout <<"You chose" << endl;

} else {

cout << "Invalid Password. Please try again" << endl;

}

}

void ChangeCustomerChoice() {

cout << "Clients Name Service Selected: "<< endl;

string i;

string one = "1 = Brokerage";

string two = "2 = Retirement";

cin >> i;

if (i == one) {

cout << one;

}

if (i == two) {

cout << two;

}

cout << "Selected Option" << endl;

cout << "Enter the number of the client that you wish to change" << endl;

cin >> client;

cout << "Please enter the client's new service choice (1 = Brokerage, 2 = Retirement)" << endl;

cin >> i;

}

// main method needed the proper command line arguments to start program

// a while loop and switch statement that allowed the user to navigate through the menu and exit program appropriately

int main() {

cout << "Created by: Janera Dobson" << endl;

cout << "Hello! Welcome to our Investment Company" << endl;

CheckUserPermissionAccess();

DisplayInfo();

ChangeCustomerChoice();

}\_

// Lastly, a bool test to check each input and ensure that the password

contains a uppercase, lowercase, and digit

\*/

#include <iostream>

#include <string>

#include <cstring> // for std::strlen

#include <cctype>

using namespace std;

string username = "";

string password;

string client = "";

int choice;

//fuction proto to check pass

bool testPass(char []); // don't need the 2nd parameter

void DisplayInfo() {

cout << "What would you like to do?" << endl << endl;

cout << "-------------- MENU OPTIONS --------------" << endl;

cout << "DISPLAY the client list (enter 1)" << endl;

cout << "CHANGE a clients choice (enter 2)" << endl;

cout << "Exit program.. (enter 3)" << endl;

cout << "------------------------------------------" << endl << endl << endl;

}

void CheckUserPermissionAccess() {

char \*password; //dynamically allocating an array

int length; //assure requested length and pass length are the same

int numCharacters; //hold number of characters for password

cout << "Enter your username: " << endl;

cin >> username;

//get the password length from the user

cout << "Your password must be at least 6 characters long." << endl;

cout << "Please enter how many characters you would like your password to be: " << endl;

cin >> numCharacters;

//validate

while (numCharacters < 6)

{

cout << "Please enter a password length of at least 6 characters." << endl;

cin >> numCharacters;

}

//dynamically allocate the array for the password

password = new char[numCharacters+1]; /// every cstring must end with a '\0' else -> crash

cout << "Please enter a password that contains at least one uppercase letter, " << endl;

cout << "one\nlowercase letter, and at least one digit." << endl;

//get users password

cin >> password;

//convert pointer/array length to interger

length = strlen(password);

//check pointer/array length against user requested pointer/array size

//to ensure consistent data

while (length != numCharacters)

{

cout << "Your password is not the size you requested. ";

cout << "Please re-enter your password." << endl;

cin >> password;

length = strlen(password);

}

// checks password credentials

if (testPass(password))

cout << "Your password is valid." << endl;

else

{

cout << "Your password is not valid." << endl;

cout << "Please refer to the above warning message." << endl;

}

delete[] password ; // hey!! don't forget to free your allocated memory!!!

}

void ChangeCustomerChoice() {

cout << "Enter client's name: " << endl;

string j;

string one = "1 = Brokerage";

string two = "2 = Retirement";

cin >> j;

if (j == one) {

cout << one;

}

if (j == two) {

cout << two;

}

cout << "Selected Option" << endl;

cout << "Enter name of the client that you wish to change" << endl;

cin >> client;

cout << "Client selected";

cout << "Please enter the client's new service choice (1 = Brokerage, 2 = Retirement)" << endl;

cin >> j;

cout << "Client's service has been changed! Goodbye..." << endl;

}

int main(int argc, char\*\* argv) {

cout << "Created by: Janera Dobson" << endl << endl;

cout << "Hello, Welcome to our Investment Company!" << endl << endl;

CheckUserPermissionAccess();

DisplayInfo();

const int displayClient = 1,

changeClientChoice = 2,

exitProgram = 3;

while (choice < displayClient || choice > exitProgram) {

cout << "Please enter a valid menu choice: ";

cin >> choice;

}

if (choice != exitProgram) {

switch (choice) {

case displayClient:

cout << "Jay Dobson, Eko Dobson, Nate Smith, Justice Braxton, Dre Figueroa";

break;

case changeClientChoice:

ChangeCustomerChoice();

case exitProgram:

cout << "Program terminated" << endl;

default:

break;

}

} while (choice != exitProgram);

return 0;

}

/\*This function will check each input and ensure that the password

contains a uppercase, lowercase, and digit.\*/

bool testPass(char pass[])

{

// flags

bool aUpper = false,

aLower = false,

aDigit = false ;

for ( int i = 0 ; pass[i] ; ++i )

if ( isupper(pass[i]) )

aUpper = true ;

else if ( islower(pass[i]) )

aLower = true ;

else if ( isdigit(pass[i]) )

aDigit = true ;

if ( aUpper && aLower && aDigit )

return true;

else

return false ;

}

2F 2F 0A 2F 2F 20 20 6D 61 69 6E 2E 63 70 70 0A 2F 2F 20 20 46 69 6E 61 6C 50 72 6F 6A 65 63 74 2E 63 70 70 0A 2F 2F 0A 2F 2F 20 20 43 72 65 61 74 65 64 20 62 79 20 4A 61 6E 65 72 61 20 44 6F 62 73 6F 6E 20 6F 6E 20 32 2F 31 39 2F 32 31 2E 0A 2F 2F 0A 0A 2F 2A 0A 20 23 69 6E 63 6C 75 64 65 20 3C 69 6F 73 74 72 65 61 6D 3E 0A 20 23 69 6E 63 6C 75 64 65 20 3C 73 74 72 69 6E 67 3E 0A 20 75 73 69 6E 67 20 6E 61 6D 65 73 70 61 63 65 20 73 74 64 3B 0A 0A 20 73 74 72 69 6E 67 20 75 73 65 72 6E 61 6D 65 20 3D 20 22 22 3B 0A 20 73 74 72 69 6E 67 20 70 61 73 73 77 6F 72 64 20 3D 20 22 22 3B 0A 20 73 74 72 69 6E 67 20 63 6C 69 65 6E 74 20 3D 20 22 22 3B 0A 0A 20 76 6F 69 64 20 44 69 73 70 6C 61 79 49 6E 66 6F 28 29 20 7B 0A 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 57 68 61 74 20 77 6F 75 6C 64 20 79 6F 75 20 6C 69 6B 65 20 74 6F 20 64 6F 3F 22 20 3C 3C 20 65 6E 64 6C 3B 20 2F 2F 20 6E 65 65 64 65 64 20 61 6E 6F 74 68 65 72 20 6C 69 6E 65 20 66 6F 72 20 72 65 61 64 61 62 69 6C 69 74 79 0A 20 20 20 20 20 0A 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 20 4D 45 4E 55 20 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 20 20 44 49 53 50 4C 41 59 20 74 68 65 20 63 6C 69 65 6E 74 20 6C 69 73 74 20 28 65 6E 74 65 72 20 31 29 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 20 20 43 48 41 4E 47 45 20 61 20 63 6C 69 65 6E 74 73 20 63 68 6F 69 63 65 20 28 65 6E 74 65 72 20 32 29 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 20 20 45 78 69 74 20 70 72 6F 67 72 61 6D 2E 2E 20 28 65 6E 74 65 72 20 33 29 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 22 20 3C 3C 20 65 6E 64 6C 3B 20 2F 2F 6E 65 65 64 65 64 20 61 20 66 65 77 20 6D 6F 72 65 20 6C 69 6E 65 73 20 66 6F 72 20 72 65 61 64 61 62 69 6C 69 74 79 0A 20 20 20 20 20 0A 20 7D 0A 0A 2F 2F 46 75 6E 63 74 69 6F 6E 20 68 61 64 20 73 65 72 69 6F 75 73 20 73 65 63 75 72 69 74 79 20 76 75 6C 6E 65 72 61 62 69 6C 69 74 69 65 73 0A 2F 2F 41 6E 79 6F 6E 65 20 63 6F 75 6C 64 20 68 61 76 65 20 65 76 65 6E 74 75 61 6C 6C 79 20 68 61 63 6B 65 64 20 6D 79 20 73 79 73 74 65 6D 20 62 65 63 75 61 73 65 20 70 61 73 73 77 6F 72 64 20 77 61 73 20 76 65 72 79 20 65 61 73 79 20 61 6E 64 20 6E 6F 74 20 61 75 74 68 65 6E 74 69 63 61 74 65 64 20 63 6F 72 72 65 63 74 6C 79 0A 2F 2F 54 68 65 72 65 66 6F 72 65 2C 20 49 20 68 61 64 20 74 6F 20 70 72 6F 6D 70 74 20 74 68 65 20 75 73 65 72 20 74 77 69 63 65 20 74 6F 20 62 65 20 73 75 72 65 20 74 68 65 79 20 6D 65 74 20 74 68 65 20 70 61 73 73 77 6F 72 64 20 72 65 71 75 69 72 65 6D 65 6E 74 73 2C 20 63 68 65 63 6B 20 63 72 65 64 65 6E 74 69 61 6C 73 20 77 69 74 68 20 69 66 20 73 74 61 74 65 6D 65 6E 74 20 61 6E 64 20 69 66 20 6E 6F 74 20 74 6F 20 72 65 2D 65 6E 74 65 72 20 75 6E 74 69 6C 20 63 6F 72 72 65 63 74 0A 2F 2F 20 41 6C 73 6F 2C 20 49 20 75 73 65 64 20 64 65 6C 65 74 65 5B 5D 20 74 6F 20 61 6C 6C 6F 63 61 74 65 20 73 70 61 63 65 20 74 6F 20 6D 65 6D 6F 72 79 0A 20 0A 20 76 6F 69 64 20 43 68 65 63 6B 55 73 65 72 50 65 72 6D 69 73 73 69 6F 6E 41 63 63 65 73 73 28 29 20 7B 0A 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 45 6E 74 65 72 20 79 6F 75 72 20 75 73 65 72 6E 61 6D 65 3A 20 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 20 63 69 6E 20 3E 3E 20 75 73 65 72 6E 61 6D 65 3B 0A 20 20 20 20 20 0A 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 45 6E 74 65 72 20 70 61 73 73 77 6F 72 64 3A 20 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 20 63 69 6E 20 3E 3E 20 70 61 73 73 77 6F 72 64 3B 0A 20 20 20 20 20 0A 20 20 20 20 20 69 66 28 70 61 73 73 77 6F 72 64 20 3D 3D 20 22 32 32 34 32 30 22 29 7B 0A 20 20 20 20 20 20 20 20 20 63 6F 75 74 20 3C 3C 22 59 6F 75 20 63 68 6F 73 65 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 20 7D 20 65 6C 73 65 20 7B 0A 20 20 20 20 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 49 6E 76 61 6C 69 64 20 50 61 73 73 77 6F 72 64 2E 20 50 6C 65 61 73 65 20 74 72 79 20 61 67 61 69 6E 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 20 7D 0A 20 7D 0A 0A 20 76 6F 69 64 20 43 68 61 6E 67 65 43 75 73 74 6F 6D 65 72 43 68 6F 69 63 65 28 29 20 7B 0A 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 43 6C 69 65 6E 74 73 20 4E 61 6D 65 20 53 65 72 76 69 63 65 20 53 65 6C 65 63 74 65 64 3A 20 22 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 20 0A 20 20 20 20 20 73 74 72 69 6E 67 20 69 3B 0A 20 20 20 20 20 73 74 72 69 6E 67 20 6F 6E 65 20 3D 20 22 31 20 3D 20 42 72 6F 6B 65 72 61 67 65 22 3B 0A 20 20 20 20 20 73 74 72 69 6E 67 20 74 77 6F 20 3D 20 22 32 20 3D 20 52 65 74 69 72 65 6D 65 6E 74 22 3B 0A 20 20 20 20 20 0A 20 20 20 20 20 63 69 6E 20 3E 3E 20 69 3B 0A 0A 20 20 20 20 20 0A 20 20 20 20 20 69 66 20 28 69 20 3D 3D 20 6F 6E 65 29 20 7B 0A 20 20 20 20 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 6F 6E 65 3B 0A 20 20 20 20 20 7D 0A 20 20 20 20 20 69 66 20 28 69 20 3D 3D 20 74 77 6F 29 20 7B 0A 20 20 20 20 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 74 77 6F 3B 0A 20 20 20 20 20 7D 0A 20 20 20 20 20 0A 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 53 65 6C 65 63 74 65 64 20 4F 70 74 69 6F 6E 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 20 0A 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 45 6E 74 65 72 20 74 68 65 20 6E 75 6D 62 65 72 20 6F 66 20 74 68 65 20 63 6C 69 65 6E 74 20 74 68 61 74 20 79 6F 75 20 77 69 73 68 20 74 6F 20 63 68 61 6E 67 65 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 20 63 69 6E 20 3E 3E 20 63 6C 69 65 6E 74 3B 0A 20 20 20 20 20 0A 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 50 6C 65 61 73 65 20 65 6E 74 65 72 20 74 68 65 20 63 6C 69 65 6E 74 27 73 20 6E 65 77 20 73 65 72 76 69 63 65 20 63 68 6F 69 63 65 20 28 31 20 3D 20 42 72 6F 6B 65 72 61 67 65 2C 20 32 20 3D 20 52 65 74 69 72 65 6D 65 6E 74 29 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 20 63 69 6E 20 3E 3E 20 69 3B 0A 20 7D 0A 0A 2F 2F 20 6D 61 69 6E 20 6D 65 74 68 6F 64 20 6E 65 65 64 65 64 20 74 68 65 20 70 72 6F 70 65 72 20 63 6F 6D 6D 61 6E 64 20 6C 69 6E 65 20 61 72 67 75 6D 65 6E 74 73 20 74 6F 20 73 74 61 72 74 20 70 72 6F 67 72 61 6D 0A 2F 2F 20 61 20 77 68 69 6C 65 20 6C 6F 6F 70 20 61 6E 64 20 73 77 69 74 63 68 20 73 74 61 74 65 6D 65 6E 74 20 74 68 61 74 20 61 6C 6C 6F 77 65 64 20 74 68 65 20 75 73 65 72 20 74 6F 20 6E 61 76 69 67 61 74 65 20 74 68 72 6F 75 67 68 20 74 68 65 20 6D 65 6E 75 20 61 6E 64 20 65 78 69 74 20 70 72 6F 67 72 61 6D 20 61 70 70 72 6F 70 72 69 61 74 65 6C 79 0A 20 69 6E 74 20 6D 61 69 6E 28 29 20 7B 0A 20 20 20 20 20 0A 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 43 72 65 61 74 65 64 20 62 79 3A 20 4A 61 6E 65 72 61 20 44 6F 62 73 6F 6E 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 20 0A 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 48 65 6C 6C 6F 21 20 57 65 6C 63 6F 6D 65 20 74 6F 20 6F 75 72 20 49 6E 76 65 73 74 6D 65 6E 74 20 43 6F 6D 70 61 6E 79 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 20 0A 20 20 20 20 20 43 68 65 63 6B 55 73 65 72 50 65 72 6D 69 73 73 69 6F 6E 41 63 63 65 73 73 28 29 3B 0A 20 20 20 20 20 0A 20 20 20 20 20 44 69 73 70 6C 61 79 49 6E 66 6F 28 29 3B 0A 20 20 20 20 20 0A 20 20 20 20 20 43 68 61 6E 67 65 43 75 73 74 6F 6D 65 72 43 68 6F 69 63 65 28 29 3B 0A 20 7D 04 0A 20 0A 20 2F 2F 20 4C 61 73 74 6C 79 2C 20 61 20 62 6F 6F 6C 20 74 65 73 74 20 74 6F 20 63 68 65 63 6B 20 65 61 63 68 20 69 6E 70 75 74 20 61 6E 64 20 65 6E 73 75 72 65 20 74 68 61 74 20 74 68 65 20 70 61 73 73 77 6F 72 64 0A 20 63 6F 6E 74 61 69 6E 73 20 61 20 75 70 70 65 72 63 61 73 65 2C 20 6C 6F 77 65 72 63 61 73 65 2C 20 61 6E 64 20 64 69 67 69 74 0A 20 2A 2F 0A 0A 0A 0A 23 69 6E 63 6C 75 64 65 20 3C 69 6F 73 74 72 65 61 6D 3E 0A 23 69 6E 63 6C 75 64 65 20 3C 73 74 72 69 6E 67 3E 0A 23 69 6E 63 6C 75 64 65 20 3C 63 73 74 72 69 6E 67 3E 20 2F 2F 20 66 6F 72 20 73 74 64 3A 3A 73 74 72 6C 65 6E 0A 23 69 6E 63 6C 75 64 65 20 3C 63 63 74 79 70 65 3E 0A 75 73 69 6E 67 20 6E 61 6D 65 73 70 61 63 65 20 73 74 64 3B 0A 0A 73 74 72 69 6E 67 20 75 73 65 72 6E 61 6D 65 20 3D 20 22 22 3B 0A 73 74 72 69 6E 67 20 70 61 73 73 77 6F 72 64 3B 0A 73 74 72 69 6E 67 20 63 6C 69 65 6E 74 20 3D 20 22 22 3B 0A 69 6E 74 20 63 68 6F 69 63 65 3B 0A 0A 2F 2F 66 75 63 74 69 6F 6E 20 70 72 6F 74 6F 20 74 6F 20 63 68 65 63 6B 20 70 61 73 73 0A 62 6F 6F 6C 20 74 65 73 74 50 61 73 73 28 63 68 61 72 20 5B 5D 29 3B 20 2F 2F 20 64 6F 6E 27 74 20 6E 65 65 64 20 74 68 65 20 32 6E 64 20 70 61 72 61 6D 65 74 65 72 0A 0A 76 6F 69 64 20 44 69 73 70 6C 61 79 49 6E 66 6F 28 29 20 7B 0A 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 57 68 61 74 20 77 6F 75 6C 64 20 79 6F 75 20 6C 69 6B 65 20 74 6F 20 64 6F 3F 22 20 3C 3C 20 65 6E 64 6C 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 0A 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 20 4D 45 4E 55 20 4F 50 54 49 4F 4E 53 20 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 44 49 53 50 4C 41 59 20 74 68 65 20 63 6C 69 65 6E 74 20 6C 69 73 74 20 28 65 6E 74 65 72 20 31 29 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 43 48 41 4E 47 45 20 61 20 63 6C 69 65 6E 74 73 20 63 68 6F 69 63 65 20 28 65 6E 74 65 72 20 32 29 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 45 78 69 74 20 70 72 6F 67 72 61 6D 2E 2E 20 28 65 6E 74 65 72 20 33 29 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 2D 22 20 3C 3C 20 65 6E 64 6C 20 3C 3C 20 65 6E 64 6C 20 3C 3C 20 65 6E 64 6C 3B 0A 7D 0A 0A 76 6F 69 64 20 43 68 65 63 6B 55 73 65 72 50 65 72 6D 69 73 73 69 6F 6E 41 63 63 65 73 73 28 29 20 7B 0A 20 20 20 20 0A 20 20 20 20 63 68 61 72 20 2A 70 61 73 73 77 6F 72 64 3B 20 2F 2F 64 79 6E 61 6D 69 63 61 6C 6C 79 20 61 6C 6C 6F 63 61 74 69 6E 67 20 61 6E 20

61 72 72 61 79 0A 20 20 20 20 69 6E 74 20 6C 65 6E 67 74 68 3B 20 2F 2F 61 73 73 75 72 65 20 72 65 71 75 65 73 74 65 64 20 6C 65 6E 67 74 68 20 61 6E 64 20 70 61 73 73 20 6C 65 6E 67 74 68 20 61 72 65 20 74 68 65 20 73 61 6D 65 0A 20 20 20 20 69 6E 74 20 6E 75 6D 43 68 61 72 61 63 74 65 72 73 3B 20 2F 2F 68 6F 6C 64 20 6E 75 6D 62 65 72 20 6F 66 20 63 68 61 72 61 63 74 65 72 73 20 66 6F 72 20 70 61 73 73 77 6F 72 64 0A 20 20 20 20 0A 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 45 6E 74 65 72 20 79 6F 75 72 20 75 73 65 72 6E 61 6D 65 3A 20 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 63 69 6E 20 3E 3E 20 75 73 65 72 6E 61 6D 65 3B 0A 20 20 20 20 0A 20 20 20 20 2F 2F 67 65 74 20 74 68 65 20 70 61 73 73 77 6F 72 64 20 6C 65 6E 67 74 68 20 66 72 6F 6D 20 74 68 65 20 75 73 65 72 0A 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 59 6F 75 72 20 70 61 73 73 77 6F 72 64 20 6D 75 73 74 20 62 65 20 61 74 20 6C 65 61 73 74 20 36 20 63 68 61 72 61 63 74 65 72 73 20 6C 6F 6E 67 2E 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 50 6C 65 61 73 65 20 65 6E 74 65 72 20 68 6F 77 20 6D 61 6E 79 20 63 68 61 72 61 63 74 65 72 73 20 79 6F 75 20 77 6F 75 6C 64 20 6C 69 6B 65 20 79 6F 75 72 20 70 61 73 73 77 6F 72 64 20 74 6F 20 62 65 3A 20 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 63 69 6E 20 3E 3E 20 6E 75 6D 43 68 61 72 61 63 74 65 72 73 3B 0A 20 20 20 20 0A 20 20 20 20 2F 2F 76 61 6C 69 64 61 74 65 0A 20 20 20 20 77 68 69 6C 65 20 28 6E 75 6D 43 68 61 72 61 63 74 65 72 73 20 3C 20 36 29 0A 20 20 20 20 7B 0A 20 20 20 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 50 6C 65 61 73 65 20 65 6E 74 65 72 20 61 20 70 61 73 73 77 6F 72 64 20 6C 65 6E 67 74 68 20 6F 66 20 61 74 20 6C 65 61 73 74 20 36 20 63 68 61 72 61 63 74 65 72 73 2E 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 20 20 20 20 63 69 6E 20 3E 3E 20 6E 75 6D 43 68 61 72 61 63 74 65 72 73 3B 0A 20 20 20 20 7D 0A 20 20 20 20 0A 20 20 20 20 2F 2F 64 79 6E 61 6D 69 63 61 6C 6C 79 20 61 6C 6C 6F 63 61 74 65 20 74 68 65 20 61 72 72 61 79 20 66 6F 72 20 74 68 65 20 70 61 73 73 77 6F 72 64 0A 20 20 20 20 70 61 73 73 77 6F 72 64 20 3D 20 6E 65 77 20 63 68 61 72 5B 6E 75 6D 43 68 61 72 61 63 74 65 72 73 2B 31 5D 3B 20 2F 2F 2F 20 65 76 65 72 79 20 63 73 74 72 69 6E 67 20 6D 75 73 74 20 65 6E 64 20 77 69 74 68 20 61 20 27 5C 30 27 20 65 6C 73 65 20 2D 3E 20 63 72 61 73 68 0A 20 20 20 20 0A 20 20 20 20 0A 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 50 6C 65 61 73 65 20 65 6E 74 65 72 20 61 20 70 61 73 73 77 6F 72 64 20 74 68 61 74 20 63 6F 6E 74 61 69 6E 73 20 61 74 20 6C 65 61 73 74 20 6F 6E 65 20 75 70 70 65 72 63 61 73 65 20 6C 65 74 74 65 72 2C 20 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 6F 6E 65 5C 6E 6C 6F 77 65 72 63 61 73 65 20 6C 65 74 74 65 72 2C 20 61 6E 64 20 61 74 20 6C 65 61 73 74 20 6F 6E 65 20 64 69 67 69 74 2E 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 0A 20 20 20 20 2F 2F 67 65 74 20 75 73 65 72 73 20 70 61 73 73 77 6F 72 64 0A 20 20 20 20 63 69 6E 20 3E 3E 20 70 61 73 73 77 6F 72 64 3B 0A 20 20 20 20 0A 20 20 20 20 2F 2F 63 6F 6E 76 65 72 74 20 70 6F 69 6E 74 65 72 2F 61 72 72 61 79 20 6C 65 6E 67 74 68 20 74 6F 20 69 6E 74 65 72 67 65 72 0A 20 20 20 20 6C 65 6E 67 74 68 20 3D 20 73 74 72 6C 65 6E 28 70 61 73 73 77 6F 72 64 29 3B 0A 20 20 20 20 0A 20 20 20 20 2F 2F 63 68 65 63 6B 20 70 6F 69 6E 74 65 72 2F 61 72 72 61 79 20 6C 65 6E 67 74 68 20 61 67 61 69 6E 73 74 20 75 73 65 72 20 72 65 71 75 65 73 74 65 64 20 70 6F 69 6E 74 65 72 2F 61 72 72 61 79 20 73 69 7A 65 0A 20 20 20 20 2F 2F 74 6F 20 65 6E 73 75 72 65 20 63 6F 6E 73 69 73 74 65 6E 74 20 64 61 74 61 0A 20 20 20 20 77 68 69 6C 65 20 28 6C 65 6E 67 74 68 20 21 3D 20 6E 75 6D 43 68 61 72 61 63 74 65 72 73 29 0A 20 20 20 20 7B 0A 20 20 20 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 59 6F 75 72 20 70 61 73 73 77 6F 72 64 20 69 73 20 6E 6F 74 20 74 68 65 20 73 69 7A 65 20 79 6F 75 20 72 65 71 75 65 73 74 65 64 2E 20 22 3B 0A 20 20 20 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 50 6C 65 61 73 65 20 72 65 2D 65 6E 74 65 72 20 79 6F 75 72 20 70 61 73 73 77 6F 72 64 2E 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 20 20 20 20 63 69 6E 20 3E 3E 20 70 61 73 73 77 6F 72 64 3B 0A 20 20 20 20 20 20 20 20 6C 65 6E 67 74 68 20 3D 20 73 74 72 6C 65 6E 28 70 61 73 73 77 6F 72 64 29 3B 0A 20 20 20 20 7D 0A 20 20 20 20 0A 20 20 20 20 2F 2F 20 63 68 65 63 6B 73 20 70 61 73 73 77 6F 72 64 20 63 72 65 64 65 6E 74 69 61 6C 73 0A 20 20 20 20 69 66 20 28 74 65 73 74 50 61 73 73 28 70 61 73 73 77 6F 72 64 29 29 0A 20 20 20 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 59 6F 75 72 20 70 61 73 73 77 6F 72 64 20 69 73 20 76 61 6C 69 64 2E 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 65 6C 73 65 0A 20 20 20 20 7B 0A 20 20 20 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 59 6F 75 72 20 70 61 73 73 77 6F 72 64 20 69 73 20 6E 6F 74 20 76 61 6C 69 64 2E 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 50 6C 65 61 73 65 20 72 65 66 65 72 20 74 6F 20 74 68 65 20 61 62 6F 76 65 20 77 61 72 6E 69 6E 67 20 6D 65 73 73 61 67 65 2E 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 7D 0A 20 20 20 20 0A 20 20 20 20 64 65 6C 65 74 65 5B 5D 20 70 61 73 73 77 6F 72 64 20 3B 20 2F 2F 20 68 65 79 21 21 20 64 6F 6E 27 74 20 66 6F 72 67 65 74 20 74 6F 20 66 72 65 65 20 79 6F 75 72 20 61 6C 6C 6F 63 61 74 65 64 20 6D 65 6D 6F 72 79 21 21 21 0A 7D 0A 0A 76 6F 69 64 20 43 68 61 6E 67 65 43 75 73 74 6F 6D 65 72 43 68 6F 69 63 65 28 29 20 7B 0A 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 45 6E 74 65 72 20 63 6C 69 65 6E 74 27 73 20 6E 61 6D 65 3A 20 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 0A 20 20 20 20 73 74 72 69 6E 67 20 6A 3B 0A 20 20 20 20 73 74 72 69 6E 67 20 6F 6E 65 20 3D 20 22 31 20 3D 20 42 72 6F 6B 65 72 61 67 65 22 3B 0A 20 20 20 20 73 74 72 69 6E 67 20 74 77 6F 20 3D 20 22 32 20 3D 20 52 65 74 69 72 65 6D 65 6E 74 22 3B 0A 20 20 20 20 0A 20 20 20 20 63 69 6E 20 3E 3E 20 6A 3B 0A 20 20 20 20 0A 20 20 20 20 69 66 20 28 6A 20 3D 3D 20 6F 6E 65 29 20 7B 0A 20 20 20 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 6F 6E 65 3B 0A 20 20 20 20 7D 0A 20 20 20 20 69 66 20 28 6A 20 3D 3D 20 74 77 6F 29 20 7B 0A 20 20 20 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 74 77 6F 3B 0A 20 20 20 20 7D 0A 20 20 20 20 0A 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 53 65 6C 65 63 74 65 64 20 4F 70 74 69 6F 6E 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 0A 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 45 6E 74 65 72 20 6E 61 6D 65 20 6F 66 20 74 68 65 20 63 6C 69 65 6E 74 20 74 68 61 74 20 79 6F 75 20 77 69 73 68 20 74 6F 20 63 68 61 6E 67 65 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 63 69 6E 20 3E 3E 20 63 6C 69 65 6E 74 3B 0A 20 20 20 20 0A 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 43 6C 69 65 6E 74 20 73 65 6C 65 63 74 65 64 22 3B 0A 20 20 20 20 0A 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 50 6C 65 61 73 65 20 65 6E 74 65 72 20 74 68 65 20 63 6C 69 65 6E 74 27 73 20 6E 65 77 20 73 65 72 76 69 63 65 20 63 68 6F 69 63 65 20 28 31 20 3D 20 42 72 6F 6B 65 72 61 67 65 2C 20 32 20 3D 20 52 65 74 69 72 65 6D 65 6E 74 29 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 63 69 6E 20 3E 3E 20 6A 3B 0A 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 43 6C 69 65 6E 74 27 73 20 73 65 72 76 69 63 65 20 68 61 73 20 62 65 65 6E 20 63 68 61 6E 67 65 64 21 20 47 6F 6F 64 62 79 65 2E 2E 2E 22 20 3C 3C 20 65 6E 64 6C 3B 0A 7D 0A 0A 0A 69 6E 74 20 6D 61 69 6E 28 69 6E 74 20 61 72 67 63 2C 20 63 68 61 72 2A 2A 20 61 72 67 76 29 20 7B 0A 20 20 20 20 0A 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 43 72 65 61 74 65 64 20 62 79 3A 20 4A 61 6E 65 72 61 20 44 6F 62 73 6F 6E 22 20 3C 3C 20 65 6E 64 6C 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 0A 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 48 65 6C 6C 6F 2C 20 57 65 6C 63 6F 6D 65 20 74 6F 20 6F 75 72 20 49 6E 76 65 73 74 6D 65 6E 74 20 43 6F 6D 70 61 6E 79 21 22 20 3C 3C 20 65 6E 64 6C 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 0A 20 20 20 20 43 68 65 63 6B 55 73 65 72 50 65 72 6D 69 73 73 69 6F 6E 41 63 63 65 73 73 28 29 3B 0A 20 20 20 20 0A 20 20 20 20 44 69 73 70 6C 61 79 49 6E 66 6F 28 29 3B 0A 20 20 20 20 0A 20 20 20 20 63 6F 6E 73 74 20 69 6E 74 20 64 69 73 70 6C 61 79 43 6C 69 65 6E 74 20 3D 20 31 2C 0A 20 20 20 20 63 68 61 6E 67 65 43 6C 69 65 6E 74 43 68 6F 69 63 65 20 3D 20 32 2C 0A 20 20 20 20 65 78 69 74 50 72 6F 67 72 61 6D 20 3D 20 33 3B 0A 20 20 20 20 0A 20 20 20 20 77 68 69 6C 65 20 28 63 68 6F 69 63 65 20 3C 20 64 69 73 70 6C 61 79 43 6C 69 65 6E 74 20 7C 7C 20 63 68 6F 69 63 65 20 3E 20 65 78 69 74 50 72 6F 67 72 61 6D 29 20 7B 0A 20 20 20 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 50 6C 65 61 73 65 20 65 6E 74 65 72 20 61 20 76 61 6C 69 64 20 6D 65 6E 75 20 63 68 6F 69 63 65 3A 20 22 3B 0A 20 20 20 20 20 20 20 20 63 69 6E 20 3E 3E 20 63 68 6F 69 63 65 3B 0A 20 20 20 20 7D 0A 20 20 20 20 0A 20 20 20 20 0A 20 20 20 20 69 66 20 28 63 68 6F 69 63 65 20 21 3D 20 65 78 69 74 50 72 6F 67 72 61 6D 29 20 7B 0A 20 20 20 20 20 20 20 20 73 77 69 74 63 68 20 28 63 68 6F 69 63 65 29 20 7B 0A 20 20 20 20 20 20 20 20 20 20 20 20 63 61 73 65 20 64 69 73 70 6C 61 79 43 6C 69 65 6E 74 3A 0A 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 4A 61 79 20 44 6F 62 73 6F 6E 2C 20 45 6B 6F 20 44 6F 62 73 6F 6E 2C 20 4E 61 74 65 20 53 6D 69 74 68 2C 20 4A 75 73 74 69 63 65 20 42 72 61 78 74 6F 6E 2C 20 44 72 65 20 46 69 67 75 65 72 6F 61 22 3B 0A 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 62 72 65 61 6B 3B 0A 20 20 20 20 20 20 20 20 20 20 20 20 63 61 73 65 20 63 68 61 6E 67 65 43 6C 69 65 6E 74 43 68 6F 69 63 65 3A 0A 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 43 68 61 6E 67 65 43 75 73 74 6F 6D 65 72 43 68 6F 69 63 65 28 29 3B 0A 20 20 20 20 20 20 20 20 20 20 20 20 63 61 73 65 20 65 78 69 74 50 72 6F 67 72 61 6D 3A 0A 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 63 6F 75 74 20 3C 3C 20 22 50 72 6F 67 72 61 6D 20 74 65 72 6D 69 6E 61 74 65 64 22 20 3C 3C 20 65 6E 64 6C 3B 0A 20 20 20 20 20 20 20 20 20 20 20 20 64 65 66 61

75 6C 74 3A 0A 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 62 72 65 61 6B 3B 0A 20 20 20 20 20 20 20 20 7D 0A 20 20 20 20 7D 20 77 68 69 6C 65 20 28 63 68 6F 69 63 65 20 21 3D 20 65 78 69 74 50 72 6F 67 72 61 6D 29 3B 0A 20 20 20 20 72 65 74 75 72 6E 20 30 3B 0A 7D 0A 0A 2F 2A 54 68 69 73 20 66 75 6E 63 74 69 6F 6E 20 77 69 6C 6C 20 63 68 65 63 6B 20 65 61 63 68 20 69 6E 70 75 74 20 61 6E 64 20 65 6E 73 75 72 65 20 74 68 61 74 20 74 68 65 20 70 61 73 73 77 6F 72 64 0A 20 63 6F 6E 74 61 69 6E 73 20 61 20 75 70 70 65 72 63 61 73 65 2C 20 6C 6F 77 65 72 63 61 73 65 2C 20 61 6E 64 20 64 69 67 69 74 2E 2A 2F 0A 0A 62 6F 6F 6C 20 74 65 73 74 50 61 73 73 28 63 68 61 72 20 70 61 73 73 5B 5D 29 0A 7B 0A 20 20 20 20 2F 2F 20 66 6C 61 67 73 0A 20 20 20 20 62 6F 6F 6C 20 61 55 70 70 65 72 20 3D 20 66 61 6C 73 65 2C 0A 20 20 20 20 61 4C 6F 77 65 72 20 3D 20 66 61 6C 73 65 2C 0A 20 20 20 20 61 44 69 67 69 74 20 3D 20 66 61 6C 73 65 20 3B 0A 20 20 20 20 66 6F 72 20 28 20 69 6E 74 20 69 20 3D 20 30 20 3B 20 70 61 73 73 5B 69 5D 20 3B 20 2B 2B 69 20 29 0A 20 20 20 20 69 66 20 28 20 69 73 75 70 70 65 72 28 70 61 73 73 5B 69 5D 29 20 29 0A 20 20 20 20 20 20 20 20 61 55 70 70 65 72 20 3D 20 74 72 75 65 20 3B 0A 20 20 20 20 65 6C 73 65 20 69 66 20 28 20 69 73 6C 6F 77 65 72 28 70 61 73 73 5B 69 5D 29 20 29 0A 20 20 20 20 20 20 20 20 61 4C 6F 77 65 72 20 3D 20 74 72 75 65 20 3B 0A 20 20 20 20 65 6C 73 65 20 69 66 20 28 20 69 73 64 69 67 69 74 28 70 61 73 73 5B 69 5D 29 20 29 0A 20 20 20 20 20 20 20 20 61 44 69 67 69 74 20 3D 20 74 72 75 65 20 3B 0A 20 20 20 20 69 66 20 28 20 61 55 70 70 65 72 20 26 26 20 61 4C 6F 77 65 72 20 26 26 20 61 44 69 67 69 74 20 29 0A 20 20 20 20 20 20 20 20 72 65 74 75 72 6E 20 74 72 75 65 3B 0A 20 20 20 20 65 6C 73 65 0A 20 20 20 20 20 20 20 20 72 65 74 75 72 6E 20 66 61 6C 73 65 20 3B 0A 7D 0A