

Object-Oriented Programming

Lab #04

Department: 응용물리학과

Student ID: 2017103038

Name: 권인회

A. Code explanation & output analysis (Write the source code and results)

A-1. Listing 5.11

: < 코드 >

```
#include <iostream>
int main() {
    std::cout << "Help! My computer doesn't work!\n";
    char choice;
    std::cout << "Does the computer make any sounds "
                << "(fans, etc.) or show any lights? (y/n):";
    std::cin >> choice;
    // The troubleshooting control logic
    if (choice == 'n') { // The computer does not have power
        std::cout << "Is it plugged in? (y/n):";
        std::cin >> choice;
        if (choice == 'n') { // It is not plugged in, plug it in
            std::cout << "Plug it in. If the problem persists, "
                        << "please run this program again.\n";
        }
        else { // It is plugged in
            std::cout << "Is the switch in the W'onW" position? (y/n):";
            std::cin >> choice;
            if (choice == 'n') { // The switch is off, turn it on!
                std::cout << "Turn it on. If the problem persists, "
                            << "please run this program again.\n";
            }
            else { // The switch is on
                std::cout << "Does the computer have a fuse? (y/n):";
                std::cin >> choice;
                if (choice == 'n') { // No fuse
                    std::cout << "Is the outlet OK? (y/n):";
                    std::cin >> choice;
                    if (choice == 'n') { // Fix outlet
                        std::cout << "Check the outlet's circuit "
                                    << "breaker or fuse. Move to a "
                                    << "new outlet, if necessary. "
                                    << "If the problem persists, "
                                    << "please run this program
again.\n";
                    }
                    else { // Beats me!
                        std::cout << "Please consult a service "
                                    << "technician.\n";
                    }
                }
            }
        }
    }
}
```

```

        else { // Check fuse
            std::cout << "Check the fuse. Replace if "
                << "necessary. If the problem "
                << "persists, then "
                << "please run this program again.\n";
        }
    }
}
else { // The computer has power
    std::cout << "Please consult a service technician.\n";
}
}

```

< 실행결과 예시 >

Help! My computer doesn't work!

Does the computer make any sounds (fans, etc.) or show any lights? (y/n):n

Is it plugged in? (y/n):y

Is the switch in the "on" position? (y/n):y

Does the computer have a fuse? (y/n):n

Is the outlet OK? (y/n):n

Check the outlet's circuit breaker or fuse. Move to a new outlet, if necessary.

If the problem persists, please run this program again.

C:\Users\Administrator\source\repos\Ex002\Debug\Ex002.exe(프로세스 44908개)이(가

) 종료되었습니다(코드: 0개).

이 창을 닫으려면 아무 키나 누르세요...

< 코드 설명 및 분석 >

이 코드는 아마 컴퓨터의 자가 진단 소스코드로 보인다. Choice 변수를 문자형으로 선언하고 사용자에게 y/n 둘 중의 문자를 입력 받아 할당한다. if문을 통해 만약 choice 변수에 n이 할당되면 if 조건문의 내용을 실행하고, y가 할당되면 else문의 내용을 실행하게 되는 것이다.

위에서부터 차례대로 살펴보면 컴퓨터가 어떤 소리를 내거나 불빛을 보여주는지 물어본다. 여기서 해당되면 y, 아니면 n을 입력해준다. 만약 y면 가장 바깥쪽 else문의 내용인 "Please consult a

service technician.”을 출력한다. 만약 n를 입력하면 첫번째 if문의 내용으로 들어가 다시 질문을 출력한다. 이런식으로 Nested Conditionals를 통해 문제점에 대한 조언을 출력한다.

A-2. Listing 5.12

: < 코드 >

```
#include <iostream>
int main() {
    // Some useful conversion constants
    const int SECONDS_PER_MINUTE = 60,
              SECONDS_PER_HOUR = 60 * SECONDS_PER_MINUTE; // 3600
    int hours, minutes, seconds;
    std::cout << "Please enter the number of seconds:";
    std::cin >> seconds;
    // First, compute the number of hours in the given number
    // of seconds
    hours = seconds / SECONDS_PER_HOUR; // 3600 seconds = 1 hours
    // Compute the remaining seconds after the hours are
    // accounted for
    seconds = seconds % SECONDS_PER_HOUR;
    // Next, compute the number of minutes in the remaining
    // number of seconds
    minutes = seconds / SECONDS_PER_MINUTE; // 60 seconds = 1 minute
    // Compute the remaining seconds after the minutes are
    // accounted for
    seconds = seconds % SECONDS_PER_MINUTE;
    // Report the results
    std::cout << hours;
    // Decide between singular and plural form of hours
    if (hours == 1)
        std::cout << " hour ";
    else
        std::cout << " hours ";
    std::cout << minutes;
    // Decide between singular and plural form of minutes
    if (minutes == 1)
        std::cout << " minute ";
    else
        std::cout << " minutes ";
    std::cout << seconds;
    // Decide between singular and plural form of seconds
    if (seconds == 1)
        std::cout << " second";
    else
        std::cout << " seconds";
    std::cout << '\n';
}
```

< 실행결과 예시 >

Please enter the number of seconds:4560

1 hour 16 minutes 0 seconds

C:\Users\Administrator\source\repos\WEx002\Debug\WEx002.exe(프로세스 47844개)이(가)
) 종료되었습니다(코드: 0개).

이 창을 닫으려면 아무 키나 누르세요...

< 코드 설명 및 분석 >

이 코드는 '초' 시간을 입력 받아 시간과 분, 초로 자세한 시간으로 변환해주는 프로그램이다. 초기화이후 변경할 수 없는 변수 SECONDS_PER_MINUTE을 60으로 초기화하고 SECONDS_PER_HOUR를 3600으로 초기화한다. 아마도 1분에 60초, 1시간에 3600초를 나타내는 듯 하다. 이후 hours, minutes, seconds 변수를 정수형으로 선언해주고 seconds 변수를 입력 받는다. 입력 받은 seconds를 3600으로 나누어 hours 변수에 대입한다. 이 때, seconds가 3600이 되지 않는다면 몫이 0이기 때문에 hours에는 0이 들어간다. 이후 seconds는 3600으로 나눈 나머지의 값이 대입된다. 마찬가지로 minutes도 진행된다. 이후 if문에 따라 숫자가 단수이면 시간의 단수형 단어를 출력하고, 숫자가 복수이면 시간의 복수형 단어를 출력하도록 설정되어있다.

B. Exercises (Write the questions down on your answer sheet)

(pp. 117-121), Exercises 3, 4, 8, 9, 12, 13, 15, 16, 17, 18

(write output analysis for all exercises)

3. What is the integer equivalent to true in C++?

: 0을 제외한 모든 정수가 true와 동일하다.

4. What is the integer equivalent to false in C++?

: 오직 0만 false와 동일하다.

8. Given the following declarations: int x = 3, y = 5, z = 7; bool b1 = true, b2 = false, b3 = x == 3, b4 = y < 3; evaluate the following Boolean expressions:

- (a) $x == 3$: x 에는 정수 3이 할당되어있으므로 True
- (b) $x < y$: y 에 할당된 정수 5는 x 에 할당된 정수 3보다 크므로 True
- (c) $x >= y$: x 에 할당된 정수 3은 y 에 할당된 정수 5보다 작으므로 False
- (d) $x <= y$: y 에 할당된 정수 5는 x 에 할당된 정수 3보다 크므로 True
- (e) $x != y - 2$: x 에 할당된 정수 3은 y 에 할당된 정수 5에서 2를 빼것과 같으므로 True
- (f) $x < 10$: x 에 할당된 정수 3은 10보다 작으므로 True
- (g) $x >= 0 \ \&\& \ x < 10$: x 에 할당된 정수 3은 0보다 크거나 같고, 10보다 작으므로 True
- (h) $x < 0 \ \&\& \ x < 10$: x 에 할당된 정수 3은 0보다 작지 않으므로 False (둘 다 만족해야함)
- (i) $x >= 0 \ \&\& \ x < 2$: x 에 할당된 정수 3은 2보다 작지 않으므로 False (상동)
- (j) $x < 0 \ || \ x < 10$: x 에 할당된 정수는 앞의 0보다 큰 조건을 만족하므로 True (하나만 만족하면 됨)
- (k) $x > 0 \ || \ x < 10$: x 에 할당된 정수는 10보다 작으므로 True (상동)
- (l) $x < 0 \ || \ x > 10$: x 에 할당된 정수는 0보다 크고 10보다 작으므로 False (범위 겹치는 곳이 없음)
- (m) $b1 : \text{boolean true}$ 값이 할당되어있으므로 True
- (n) $!b1$: True의 반대이므로 False
- (o) $!b2$: False의 반대이므로 True
- (p) $b1 \ \&\& \ b2$: True와 False가 $\&\&$ 연산자로 이루어지면 False (True가 되려면 둘 다 True)

9. Express the following Boolean expressions in simpler form; that is, use fewer operators. x is an int.

- (a) $!(x == 2) : x != 2$
- (b) $x < 2 \ || \ x == 2 : x <= 2$
- (c) $!(x < y) : x >= y$
- (d) $!(x <= y) : x > y$
- (e) $x < 10 \ \&\& \ x > 20 : \text{false}$

(f) $x > 10 \parallel x < 20$: true

(g) $x \neq 0$: x

(h) $x == 0$: !x

12. Write a C++ program that requests an integer value from the user. If the value is between 1 and 100 inclusive, print "OK;" otherwise, do not print anything.

: < 코드 >

```
#include <iostream>

int main() {
    int a;
    std::cin >> a;
    if (a >= 1 && a <= 100)
        std::cout << "OK";
}
```

< 실행결과 예시 >

99

OK

C:\Users\Administrator\source\repos\Ex002\Debug\Ex002.exe(프로세스 31996개)이(가

) 종료되었습니다(코드: 0개).

이 창을 닫으려면 아무 키나 누르세요...

< 설명 >

&&가 and 연산자이기 때문에 if문의 두 조건을 모두 만족해야 "OK"가 출력된다.

13. Write a C++ program that requests an integer value from the user. If the value is between 1 and 100 inclusive, print "OK;" otherwise, print "Out of range."

: < 코드 >

```
#include <iostream>

int main() {
    int a;
    std::cin >> a;
    if (a >= 1 && a <= 100)
```

```

        std::cout << "OK";
    else
        std::cout << "Out of range.";
}

```

< 실행결과 예시 >

101

Out of range.

C:\Users\Administrator\source\repos\Ex002\Debug\Ex002.exe(프로세스 2332개)이(가)

종료되었습니다(코드: 0개).

이 창을 닫으려면 아무 키나 누르세요...

< 설명 >

위의 12번 문제에서 else문을 추가하여 if문의 조건을 만족하지 못할 때 출력할 내용을 설정했다.

15. Consider the following section of C++ code: // i, j, and k are ints if (i < j) { if (j < k) i = j; else j = k; } else { if (j > k) j = i; else i = k; } std::cout << "i = " << i << " j = " << j << " k = " << k << '\n'; What will the code print if the variables i, j, and k have the following values?

(a) i is 3, j is 5, and k is 7 : i = 5 j = 5 k = 7

- 첫번째 if문 조건 만족하여 두번째 if문으로 넘어갔고, 이도 만족하여 i=j가 처리되고 출력

(b) i is 3, j is 7, and k is 5 : i = 3 j = 5 k = 5

- 첫번째 if문 조건 만족하여 두번째 if문으로 넘어갔고, 이는 만족하지 못하여 두번째 else문에서 j=k가 처리되고 출력

(c) i is 5, j is 3, and k is 7 : i = 7 j = 3 k = 7

- 첫번째 if문 조건 만족하지 못하여 else문으로 넘어간 뒤, 이 안의 if문도 만족하지 못하여 else문의 내용인 i=k가 처리되고 출력

(d) i is 5, j is 7, and k is 3 : i = 5 j = 3 k = 3

- 첫번째 if문 조건 만족하여 두번째 if문으로 넘어갔고, 이는 만족하지 못하여 두번째 else문에서 j=k가 처리되고 출력

(e) i is 7, j is 3, and k is 5 : i = 5 j = 3 k = 5

- 첫번째 if문 조건 만족하지 못하여 else문으로 넘어간 뒤, 이 안의 if문도 만족하지 못하여 else

문의 내용인 $i=k$ 가 처리되고 출력

(f) i is 7, j is 5, and k is 3 : $i = 7$ $j = 7$ $k = 3$

- 첫번째 if문 조건 만족하지 못하여 else문으로 넘어간 뒤, 이 안의 if문은 만족하여 $j=i$ 가 처리되고 출력

16. Consider the following C++ program that prints one line of text: `#include <iostream> int main() { int input; std::cin >> input; if (input < 10) { if (input != 5) std::cout << "wow "; else input++; } else { if (input == 17) input += 10; else std::cout << "whoa "; } std::cout << input << '\n'; }` What will the program print if the user provides the following input?

(a) 3 : wow 3

- 첫번째 if문을 만족하였고 그 안의 두번째 if문도 만족하여 wow가 출력된 것이다.

(b) 21 : whoa 21

- 첫번째 if문을 만족하지 못하여 else문 안에 있는 두번째 if문도 만족하지 못하여 whoa를 출력한 것이다.

(c) 5 : 6

- 첫번째 if문을 만족하였으나 그 안의 두번째 if문은 만족하지 못하여 else문 안에 있는 `input++`가 실행되어 후위증가에 의해 6이 출력된 것이다.

(d) 17 : 27

- 첫번째 if문을 만족하지 못하여 else문 안에 있는 두번째 if문은 만족하여 `input += 10`이 실행된 채로 출력된 것이다.

(e) -5 : wow -5

- 첫번째 if문을 만족하였고 그 안의 두번째 if문도 만족하여 wow가 출력된 것이다.

17. Why does the following section of code always print "ByeHi"? `int x; std::cin >> x; if (x < 0); std::cout << "Bye"; std::cout << "Hi\n";`

: if문의 조건 뒤에 바로 세미콜론(;)이 붙어 if문을 통해 나오는 결과가 없기 때문이다.

`if (x < 0)`

`;`


```
std::cout << "Bye";
```

와 같은 코드이다.

18. Write a C++ program that requests five integer values from the user. It then prints the maximum and minimum values entered. If the user enters the values 3, 2, 5, 0, and 1, the program would indicate that 5 is the maximum and 0 is the minimum. Your program should handle ties properly; for example, if the user enters 2, 4, 2, 3, and 3, the program should report 2 as the minimum and 4 as maximum.

: < 코드 >

```
#include <iostream>

int main() {
    int a, b, c, d, e, max, min;
    std::cout << "정수 5개를 차례대로 입력하세요." << std::endl;
    std::cin >> a >> b >> c >> d >> e;
    max = a;
    min = a;
    if (b >= max)
        max = b;
    else if (b <= min)
        min = b;
    if (c >= max)
        max = c;
    else if (c <= min)
        min = c;
    if (d >= max)
        max = d;
    else if (d <= min)
        min = d;
    if (e >= max)
        max = e;
    else if (e <= min)
        min = e;
    std::cout << "최대값은 " << max << ", 최소값은 " << min << "이다.";
}
```

< 실행결과 예시 >

정수 5개를 차례대로 입력하세요.

2 4 2 3 3

최대값은 4, 최소값은 2이다.

C:\Users\WAdministrator\source\repos\WEx002\Debug\WEx002.exe(프로세스 41208개)이(가

) 종료되었습니다(코드: 0개).

이 창을 닫으려면 아무 키나 누르세요...

< 설명 >

앞에서부터 차례대로 이전의 최대값, 최소값과 끝까지 비교해가며 최대값, 최소값을 갱신한 뒤에
마지막에 출력한다.

C. Additional exercises (Write the questions down on your answer sheet)

C-1. If the variable `flag` is true, read the integer variables `a` and `b`, Then calculate and print the sum and average of both inputs.

< 코드 >

```
#include <iostream>

int main() {
    bool flag;
    std::cout << "합계와 평균을 구하시려면 1, 종료하려면 0을 입력하세요." << std::endl;
    std::cin >> flag;
    if (flag == true)
    {
        std::cout << "구하실 두 정수를 차례대로 입력하세요." << std::endl;
        int a, b, sum;
        double avg;
        std::cin >> a >> b;
        sum = a + b;
        avg = sum / 2.0;
        std::cout << "합계는 " << sum << std::endl;
        std::cout << "평균은 " << avg << "입니다.";
    }
}
```

< 실행결과 예시 >

합계와 평균을 구하시려면 1, 종료하려면 0을 입력하세요.

1

구하실 두 정수를 차례대로 입력하세요.

99 98

합계는 197

평균은 98.5입니다.

C:\Users\Administrator\source\repos\Ex002\Debug\Ex002.exe(프로세스 28008개)이(가)
) 종료되었습니다(코드: 0개).

이 창을 닫으려면 아무 키나 누르세요...

< 설명 >

Flag의 boolean이 true일 때, 입력받은 두 정수의 합계와 평균을 계산하는 코드를 작성한다.

C-2. Write a program that determines and prints a student's grade. The program will read three types of scores (quiz, mid-term, and final scores) and determine the grade based on the following rules:

- 1) If the average score $\geq 90\%$, then grade=A
- 2) If the average score $\geq 70\%$ and $<90\%$, then grade=B
- 3) If the average score $\geq 50\%$ and $<70\%$, then grade=C
- 4) If the average score $<50\%$, then grade=F

< 코드 >

```
#include <iostream>

int main() {
    int quiz, mid, final;
    double avg;
    char grade;
    std::cout << "<quiz점수 mid-term점수 final점수> 형식으로 입력하세요." << std::endl;
    std::cin >> quiz >> mid >> final;
    avg = (quiz + mid + final) / 3.0;
    if (avg >= 90)
        grade = 'A';
    else if (avg < 90 && avg >= 70)
        grade = 'B';
    else if (avg < 70 && avg >= 50)
        grade = 'C';
    else
        grade = 'F';
    std::cout << grade;
}
```

< 실행결과 예시 >

<quiz점수 mid-term점수 final점수> 형식으로 입력하세요.

90 85 80

B

C:\Users\Administrator\source\repos\Ex002\Debug\Ex002.exe(프로세스 24584개)이(가)
) 종료되었습니다(코드: 0개).

이 창을 닫으려면 아무 키나 누르세요...

< 설명 >

세 점수를 받아 평균 계산을 한 뒤, 소수점까지 고려하여 if문을 통해 학점을 계산한다.

C-3. Write a program that defines and prints whether an entered integer is odd or even.

< 코드 >

```
#include <iostream>

int main() {
    int num;
    std::cout << "숫자를 입력하세요." << std::endl;
    std::cin >> num;
    if (num % 2 == 1)
        std::cout << "odd number";
    else
        std::cout << "even number";
}
```

< 실행결과 예시 >

숫자를 입력하세요.

23

odd number

C:\Users\Administrator\source\repos\Ex002\Debug\Ex002.exe(프로세스 25404개)이(가)
) 종료되었습니다(코드: 0개).

이 창을 닫으려면 아무 키나 누르세요...

< 설명 >

숫자를 입력받고 2로 나눈 나머지가 1이면 홀수, 0이면 짝수로 판명한다.