|  |
| --- |
| 2020년 1학기 |
| 프로그래밍 언어 과제 |
| 5주차 레포트 |

|  |
| --- |
| 사회학과  201721899 송지민 |

**[과제#1] 실습#3~#5까지 실습수행 및 실행화면 캡처**

#include <stdio.h>

int main(void)

{

int x=0;

int y=0;

int z=0;

int min=0;

printf("3개의 정수를 입력하시오:");

scanf("%d %d %d", &x, &y, &z);

if(x>y)

{if(y>z)

{min=z;}

else

{min=y;}

}

else

{if(x>z)

{min=z;}

else

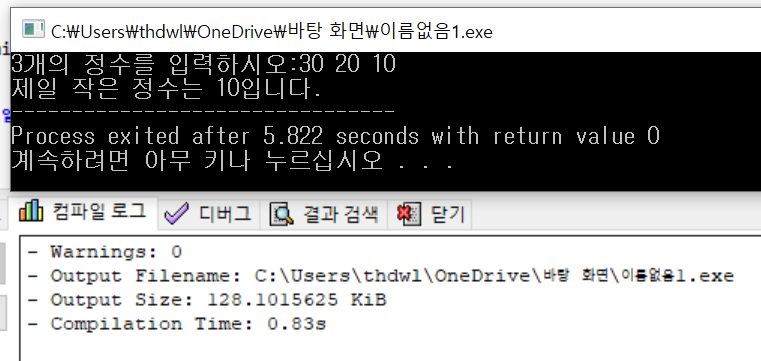
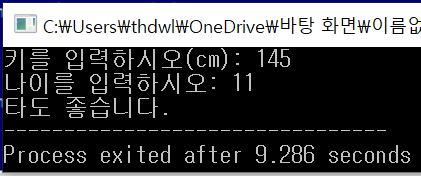
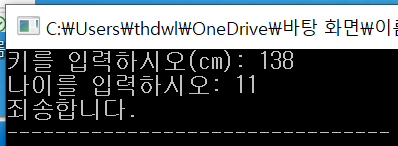
{min=x;}

}

printf("제일 작은 정수는 %d입니다.", min);

return 0;

}

* 실습#3 코드 및 실행화면
* 실습#4 코드 및 실행화면

#include <stdio.h>

int main(void)

{

int height = 0;

int age = 0;

printf("키를 입력하시오(cm): ");

scanf("%d", &height);

printf("나이를 입력하시오: ");

scanf("%d", &age);

if(height>=140)

{

if(age>=10)

{

printf("타도 좋습니다.");

}

else

{

printf("죄송합니다.");

}

}

else

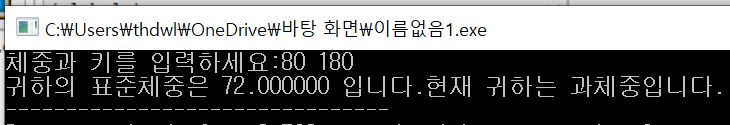
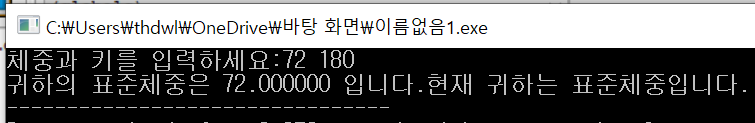
{

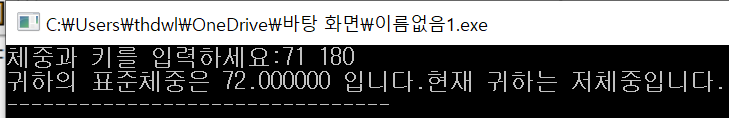
printf("죄송합니다.");

}

return 0;

}

* 실습#5 코드 및 실행화면



#include <stdio.h>

int main(void)

{

double height=0;

double weight=0;

double standard\_weight=0;

printf("체중과 키를 입력하세요:");

scanf("%lf %lf", &weight, &height);

standard\_weight=(height-100)\*0.9;

printf("귀하의 표준체중은 %lf 입니다.", standard\_weight);

if(weight>standard\_weight)

{

printf("현재 귀하는 과체중입니다.");

}

else if(weight==standard\_weight)

{

printf("현재 귀하는 표준체중입니다.");

}

else

{

printf("현재 귀하는 저체중입니다.");

}

return 0;

}

**[과제#2] 놀이공원 입장요금 계산 프로그램 작성**

* 코드 및 실행화면

// 놀이공원 입장요금 계산프로그램

#include <stdio.h> // 표준 입출력 헤더파일을 포함하여 선언하기

int main(void) // 함수의 선언부

{

// 변수 선언 및 초기화

int time=0; // 현재 시각을 입력할 변수

int age=0; // 나이를 입력할 변수

int price=0; // 시각과 나이에 따른 요금을 저장할 변수

/\*

시간, 나이와 같은 자료이므로 정수형으로 선언하며, 쓰레기 값의 저장으로 인한 오류를 최소화하기 위하여 선언과 함께 초기화한다. \*/

// 자료 입력받기

printf("현재 시간과 나이를 입력하시오(시간, 나이): "); // printf() 함수 이용하여 출력

scanf("%d %d", &time, &age); // scanf() 함수 이용하여 입력받기

/\*

printf()안에 개행문자를 입력하면 출력된 문장의 다음 행에서 자료를 입력하게 되고 개행문자를 입력하지않으면 출력된 문장의 행에서 자료를 입력하게 된다.

scanf() 함수의 경우 “”안에 개행문자를 입력해서는 안 되며, 위와 같이 변환명세와 변환명세 사이에 띄어쓰기를 입력한 경우, 실행화면에서 자료를 입력할 때에도 12 8과 같이 자료 사이에 띄어쓰기를 해야 한다.

입력받은 자료를 저장하기 위해 주소 연산자를 변수명 앞에 써서 지정해 주어야 하며, 정수형이기 때문에 변환명세는 %d를 사용한다.\*/

// 조건문 작성하기

*if(time<17)* // 현재 시간이 오후 5시 전일 경우(16:59까지일 경우) : 자유이용권

{

if(3<=age) // 나이가 3살 미만일 경우에는 시간에 상관없이 0원이므로 3살일 때부터 고려

if(age<13||age>=65) // 나이가 3살 이상이며 13살 미만일 때, 혹은 65세 이상일 때(소인)

{

price=25000; // 조건식에 따른 자유이용권 요금 = 25000원

}

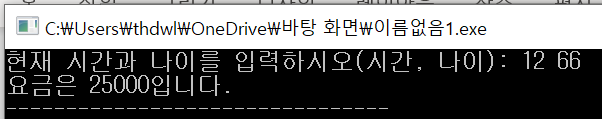
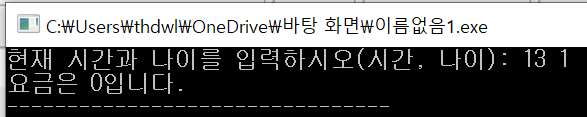
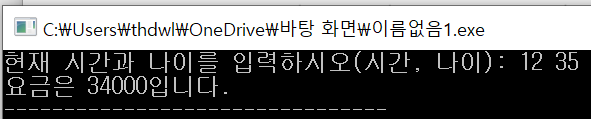
else // 나이가 13살 이상이거나 65세 미만일 때(대인)

{

price=34000; // 조건식에 따른 자유이용권 요금 = 34000원

}

}



*else* // 현재 시간이 오후 5시 이후일 경우 : 야간이용권

{

price=10000; // 야간이용권 요금은 나이에 상관 없이(3살 미만 제외) 10000원

}

if(age<3) // 나이가 3살 미만일 때 : 무료

{

price=0; // 조건식에 따른 요금

}

/\*

코드의 중복을 최대한 피하기 위하여, 모든 경우에 요금이 무료인 나이가 3살보다 적을 때는 따로 고려하였다.

또한 자유이용권의 경우에 3이상 12살 이하, 65세 이상일 경우 소인으로 간주하여 동일한 금액(25000원)을 지불해야 하므로 논리 연산자 ||을 사용하여 함께 고려하였다.

대인은 위의 조건식에 해당하지 않는 경우이므로 else를 사용하였고 각각의 문장은 {}를 사용하여 묶어주었다.

조건식 행의 끝에 ;을 붙이지 않도록 주의해야 하며 조건에 따른 if문과 else문이 잘 연결되도록 작성하는 데 주의해야 한다.\*/

// 요금 출력하기

printf("요금은 %d입니다.", price); // printf() 함수를 통해 출력하기

/\*

모든 경우에 출력되는 문장이므로 if~else 문 밖에 작성해야 한다.

또한 요금에 해당하는 price 변수가 정수형이므로 변환명세는 %d를 사용한다.\*/

return 0; // 0을 반환하여 끝내기

}

* 부록 : 버스 요금 프로그램(응용)

#include <stdio.h>

int main(void)

{

int time=0;

int age=0;

int price=0;

int national\_merit=0; // 애국지사 혹은 국가유공자 여부를 저장할 변수

printf("현재 시간과 나이를 입력하시오.(시간, 나이): ");

scanf("%d %d", &time, &age);

if(age>=6)

if(age<65)

{

printf("애국지사 혹은 국가유공자이십니까? 맞으면 1, 아니라면 2를 입력해 주세요.");

scanf("%d", &national\_merit);

}

/\*

나이가 6살 미만일 때와 65세 이상일 때에는 애국지사 혹은 국가유공자 여부와 상관 없이 무료이므로 6살 이상 65세 미만일 때만 출력한다.

또한 첫차시간인 4시부터 6시까지는 조조 할인 요금이 적용되며 6살부터 12살까지는 어린이 요금, 13살부터 19살까지는 청소년 요금, 20살부터 64살까지는 일반 요금이 적용된다.\*/

if(4<=time)

if(time<=6) // 조조 할인 적용을 위한 if~else문

{

if(6<=age)

if(age<65)

if(age<13)

{

price=600; // 조조 할인된 어린이 요금

}

else if(age<19)

{

price=900; // 조조 할인된 청소년 요금

}

else

{

price=1300; // 조조 할인된 일반 요금

}

}

else

{

if(6<=age)

if(age<65)

if(age<13)

{

price=800; // 어린이 요금

}

else if(age<19)

{

price=1100; // 청소년 요금

}

else

{

price=1500; // 일반 요금

}

}

if(age<6||age>=65||national\_merit==1) // 6살 미만이거나 65세 이상이거나 애국지사 및 국가유공자일 경우

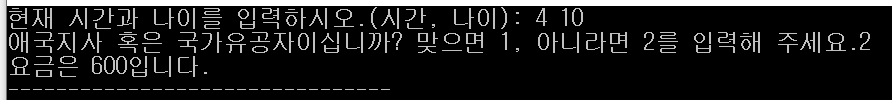
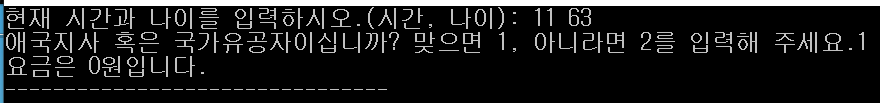
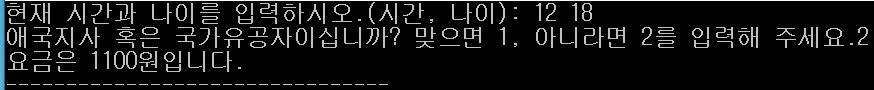
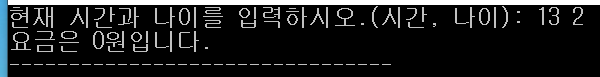
{

price=0; // 조건식에 해당하면 버스 요금은 무료

}

printf("요금은 %d원입니다.", price);

return 0;

}

코드와 실행결과는 위와 같다.

다만 중복되는 코드가 보여 수정이 필요할 것 같다.

**[과제#3] 가위바위보 프로그램 작성**

* 코드 및 실행화면

// 가위바위보 프로그램

#include <stdio.h> // 표준 입출력 헤더파일 포함하기

#include <stdlib.h> // 표준 라이브러리

/\*

stdlib.h 은 C언어에서 stdio.h와 맞먹을 정도로 빈번하게 쓰이는 헤더 파일이며

문자열 변환, 의사 난수 생성, 동적 메모리 관리, 프로세스 제어, 검색/정렬, 정수 산술 등의 함수들을 포함하고 있다.

본 프로그램에서 사용할 함수는 rand(), srand() 함수로 의사 난수 생성 함수이다.

이밖에도 변환함수인 atoi(), atol(), atof() (차례대로 int, long, double 형 자료형으로 변환한다는 의미) 함수,

동적 메모리 관리 함수인 malloc(), calloc(), free() 등이 정의되어 있다.

동적 메모리 관리 함수는 포인터와 관련하여, 메모리 공간을 할당해주는 기능을 가지며 포인터 변수 = (자료형 \*)malloc(sizeof(자료형));과 같은 형태로 사용한다.\*/

int main(void) // 함수의 선언부

{

// 변수선언 및 초기화

int player = 0; // 사용자가 입력하는 정수를 저장할 변수

int result = 0; // 가위 바위 보의 승패

// 난수 생성과 관련된 코드

srand(time(NULL)); // 난수 발생의 정확도를 위해서 사용

/\*

void srand(unsigned int seed); 의사 난수 발생기를 seed로 초기화한다. 일반적으로 seed는 time(NULL)로 설정된다.

NULL은 널 포인터의 약어인 상수이며 메모리의 어떤 유효한 위치의 개체도 가리키지 않는 포인터 값이다.

따라서 srand(time(NULL));을 통하여 난수 발생의 정확도를 높일 수 있다.

만약 그렇게 하지 않고 변수를 생성하여 사용자에게 자료를 입력하게 하고 그 자료를 srand() 함수에 넣는다면 사용자가 seed값을 직접 입력하는 것이 되기 때문에 작업이 반복될 수 있으며 효율성도 떨어지게 된다.\*/

int computer = rand()%3+1; // rand() 함수를 이용한 난수 발생

/\*

int rand(void); 0부터 RAND\_MAX 사이의 의사 난수를 반환한다.

여기서 rand() 함수로 가져오는 랜덤 숫자는 항상 동일한 패턴으로 나타나기 때문에 srand() 함수가 필요하다.

rand() 함수의 값을 3으로 나누면 몫으로 0, 1, 2가 나오므로 여기에 1을 더하면 1, 2, 3으로 가위, 바위, 보가 된다.\*/

// 자료 입력받기

printf("선택하시오(1: 가위 2: 바위 3: 보)"); // printf() 함수를 이용하여 출력

scanf("%d", &player); // scanf() 함수를 이용하여 자료 입력받기

/\*

player 변수가 정수형이므로 변환명세는 %d를 사용하며, 1은 가위, 2는 바위, 3은 보로 간주한다.\*/

// 사용자가 1, 2, 3 중 하나의 값을 입력하였을 때

if(player==1||player==2||player==3)

{

// 자료 처리하기

result = player - computer;

/\*

가위, 바위, 보를 숫자로 입력받았기 때문에 수의 차이를 통해서 승패를 결정한다.

단, 차이를 바로 결과로 입력할 경우 1, -2 일 때는 승리, -1, 2일 때는 패배이므로 switch문에서 코드가 중복될 수 있어 과정 변수를 따로 만들어 준다.\*/

// if문으로 승패 가리기

if(result==1||result==-2) // 사용자 주먹, 컴퓨터 가위/ 사용자 보, 컴퓨터 주먹/ 사용자 가위, 컴퓨터 보

{

printf("사용자가 이겼음\n"); // printf() 함수 사용하여 출력

}

else if(result==0) // 사용자 == 컴퓨터, 같음을 의미하는 == 연산자를 쎠야 함을 유의

{

printf("비겼음\n"); // printf() 함수 사용하여 출력

}

else // 나머지 경우(result==2||result-1), 사용자 가위, 컴퓨터 주먹/ 사용자 주먹, 컴퓨터 보/ 사용자 보, 컴퓨터 가위

{

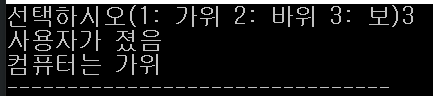
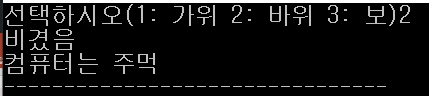
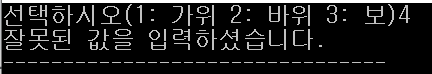
printf("사용자가 졌음\n"); // printf() 함수 사용하여 출력

}

/\*

result 변수의 값에 따라 사용자가 이겼을 때, 비겼을 때, 졌을 때와 같은 형태로 나누어서 표현한다.

개행문자를 입력해야 뒤에 올 '컴퓨터는~' 문장과 바로 이어지지 않는다.\*/



// switch문 이용하여 경우에 따른 컴퓨터의 값 나타내기

switch(computer) // 컴퓨터의 값에 따라 케이스를 나누기 때문에 () 안에는 computer가 들어감

{

case 1 : // 1은 가위로 간주하므로 case 1은 가위인 경우

printf("컴퓨터는 가위"); // printf() 함수를 이용하여 출력

break; // break; 로 빠져나오기

case 2 : // 2는 주먹으로 간주하므로 case 2은 주먹인 경우

printf("컴퓨터는 주먹"); // printf() 함수를 이용하여 출력

break; // break; 로 빠져나오기

case 3 : // 3은 보자기로 간주하므로 case 3은 보인 경우

printf("컴퓨터는 보"); // printf() 함수를 이용하여 출력

break; // break; 로 빠져나오기

}

/\*

switch문은 제어식의 값에 따라서 여러 경로 중 하나를 선택할 수 있는 제어구조이며,

break를 만날 때까지 계속 문장을 실행하므로 주의해야 한다.

default 값은 어떤 case문과도 일치하지 않는 경우 선택된다.

case \_ 에는 변수, 변수가 들어간 수식, 실수, 문자열은 사용이 불가하며 정수, 단문자만 사용 가능하다.\*/

}

else // 사용자가 1, 2, 3 외의 값을 입력하였을 때

{

printf("잘못된 값을 입력하셨습니다.");

}

return 0;

}

* 부록 : 난수를 이용한 숫자 맞추기 게임(예습)

// 숫자 맞추기 게임

#include <stdio.h> // 표준 입출력 헤더파일 포함하기

#include <stdlib.h> // 표준 라이브러리

int main(void) // 함수의 선언부

{

// 변수선언 및 초기화

int guess = 0; // 사용자가 입력하는 정수를 저장할 변수

int tries = 0; // 사용자의 시도횟수

// 난수 생성과 관련된 코드

srand(time(NULL)); // 난수 발생의 정확도를 위해서 사용

int answer = rand()%10+1; // rand() 함수를 이용한 난수 발생

// 자료 입력받기

printf("===========숫자 맞추기 게임===========\n\n\n"); // printf() 함수를 이용하여 출력

// 반복문을 이용하여 숫자 맞추기 과정 작성

do // do while 문은 반복문을 적어도 한 번 수행

{

printf("정답을 맞추시오.(정답은 1부터 10까지의 정수 중 하나)\n\n");

scanf("%d", &guess); // 값 입력받기

tries++; // 수행할 때마다 1씩 추가(첫 시도에 1)

if(guess<answer) // if문 이용하여 경우에 따라 출력하기

{

printf("%d는 정답보다 작습니다.\n\n", guess);

}

else if(guess==answer)

{

printf("축하합니다. 정답입니다.\n\n");

}

else

{

printf("%d는 정답보다 큽니다.\n\n", guess);

}

}

while (guess != answer); // 사용자의 입력값과 정답이 일치하지 않는 동안 반복한다는 의미

printf("정답은 %d입니다. 시도횟수는 %d번입니다.", answer, tries); // 정답과 시도횟수 출력하기

return 0; // 0반환하여 끝내기

}

