C++ 과제 #1

201810754 조윤직

제출일: 2019. 9. 15

문제 0: 메인 메뉴 구현

====== 201810754 조윤직 ========

1. 과제1 2)과제2 3)과제3 4)종료

메뉴를 선택하세요 :

이런 식으로 메뉴 구현

int main()

{

int menu = 0;

cout << "====== 201810754 조윤직 ========" << endl;

while (menu != 4) {

menu = 0;

while (menu > 4 || menu < 1) {

system(“cls”);

cout << "1) 과제1 2) 과제2 3) 과제3 4) 종료" << endl;

cout << "메뉴를 선택하세요 : ";

cin >> menu;

}

switch (menu)

{

case 1:

\_HW01();

break;

case 2:

\_HW02();

break;

case 3:

\_HW03();

break;

case 4:

break;

}

system("pause");

}

}

처음에는 이런 식으로 구현 했으나. cin 함수에서 정수가 아닌 값을 받을 경우 무한 루프가 발생해, 정수인 값만 받고 잘못 받을 경우 무한루프를 발생하지 않기 위해 cin.clear() 에러 상태 초기화, cin.ignore() 입력 버퍼 비우기, cin.fail() cin 에러 확인 활용하여 정수만을 받는 함수 추가.

int get\_num() { //정수만을 받기 위한 함수 정수가 아닌 값을 받으면 다시 입력 받게

int num;

cin >> num;

if (cin.fail()) {

cout << "다시 입력하세요. : ";

cin.clear();

cin.ignore(256, '\n');

return get\_num();

}

return num; }

get\_num함수를 문제 1~3에서도 입력부에서 활용

문제 1: 거스름돈 출력하기

**-문제정의**: 모든 상품의 가격이 500원 이하라고 할 때 고객은 무조건 한 개의 물건을 사며 500원을 지출한다 이때 받는 거스름돈을 100원, 50원 10원 1원 짜리 동전으로 받는다. 고객의 구매 금액이 입력 될 때 받을 각 동전 개수를 출력(단, 동전의 개수를 최소화 하여야 한다.)

-문제 분석 및 해결책 : 먼저 구매금액으로 거스름돈의 금액을 계산하고 그 뒤 큰 단위의 동전부터 몫이 계산 되는 int형 끼리의 ‘/’ 연산을 이용, 나머지는 ‘%’ 연산을 활용하여 각 동전 개수를 구한다.

-소스코드

//정수만을 받기 위한 함수 정수가 아닌 값을 받으면 다시 입력 받게

int get\_num() {

int num;

cin >> num;

if (cin.fail()) {

cout << "다시 입력하세요. : ";

cin.clear();

cin.ignore(256, '\n');

return get\_num();

}

return num;

}

void \_HW01() {

int money = 500;

int buy\_v;

cout << "====== 과제 1 ========" << endl;

do {

cout << "구매 금액을 입력하세요[0~500] : ";

buy\_v = get\_num();

} while (buy\_v > 500 || buy\_v < 0);

money -= buy\_v;

cout << money << "의 거스름돈은 다음과 같다.\n";

int cou\_100, cou\_50, cou\_10, cou\_1;

cou\_100 = money / 100;

cou\_50 = (money % 100) / 50;

cou\_10 = (money % 50) / 10;

cou\_1 = money % 10;

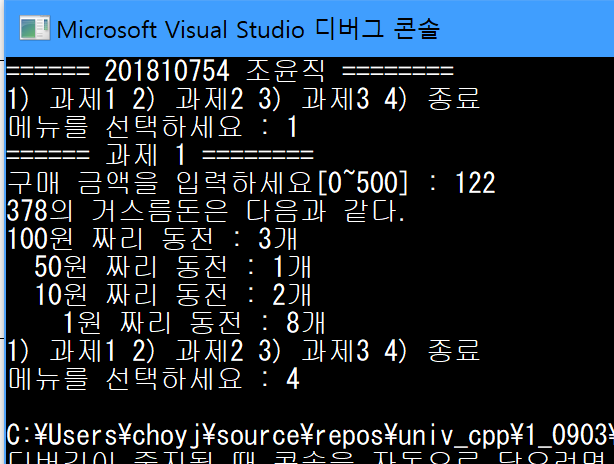
cout << "100원 짜리 동전 : " << cou\_100 << "개\n";

cout << " 50원 짜리 동전 : " << cou\_50 << "개\n";

cout << " 10원 짜리 동전 : " << cou\_10 << "개\n";

cout << " 1원 짜리 동전 : " << cou\_1 << "개\n";

}

-실행결과

-느낀점 및 토의사항 : int형 끼리 / 연산을 하면 몫만이 계산 되는 것, %나머지 연산,

문제 2: 자판기 쿠폰 사용하기

-문제정의: 자판기에서 초콜릿 바는 개당 1원 모든 초콜릿 바 안에는 쿠폰이 하나씩 들어있다. 또한 자판기에서는 쿠폰 7개의 쿠폰으로 1개의 초콜릿 바를 받을수 있다. 돈을 입력 받으면 그 돈으로 최대 몇 개의 초콜릿 바를 살수 있는 지, 남는 쿠폰은 몇 개 인지 출력

-문제 분석 및 해결책 : 돈으로 살 수 있는 최대 초콜릿을 사고나서, 가지고 있는 쿠폰으로 초콜릿을 다시 얻는 것이 한 루프라 하면 돈이 커질수록 한 두번이 아닌 많은 루프를 겨쳐야 쿠폰이 7미만으로 남는다. while문을 활용하여 쿠폰이 7개 미만으로 남을 때 까지 반복 실행.

-소스코드

void \_HW02() {

int money, choco = 0, coupon = 0;

cout << "====== 과제 2 ========" << endl;

do {

cout << "\* 돈을 넣어 주세요 : ";

money = get\_num();

} while (money < 0);

cout << "================== \n";

choco += money;

coupon += choco;

do {

int temp = coupon / 7;

choco += temp;

coupon = coupon % 7;

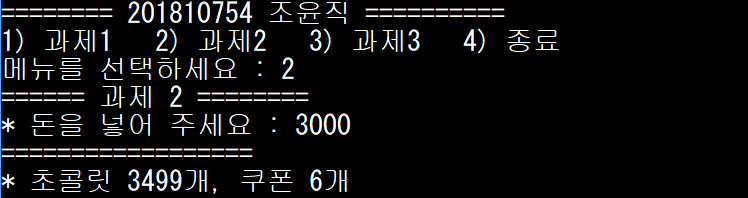
coupon += temp;

} while (coupon >= 7);

cout << "\* 초콜릿 " << choco << "개, 쿠폰 " << coupon << "개\n\n";

}

-실행결과



-느낀점 및 토의사항

처음에는 초콜릿 한번 사고, 나온 쿠폰으로 초콜릿 한번 더 사면 되겠지 생각 했지만 쿠폰이 7개 이상으로 남는 것을 보고 while문으로 계속 반복,

문제3 : 산수의 해법 찾기

-문제정의: 0~9 의 자연수를 알파벳 대문자로 표현하여 수식을 주고 수식을 만족하는 알파벳이 의미하는 값을 구하는 프로그램(단, 각 대문자가 의미하는 수는 서로 다르다. )

-문제 분석 및 해결책 : T,O,G,P에 다중 for문으로 0부터~9 까지 대입 하고 수식을 컴퓨터가 읽을 수 있는 식으로 표현하여 수식이 참일 때 그 경우의 TOGP값들을 출력한다.

-소스코드

void \_HW03() {

cout << "====== 과제 3 ========" << endl;

int count = 0;

int T, O, G, D;

cout << "TOO + TOO + TOO + TOO = GOOD\n";

cout << "식: (T \* 100 + O \* 11) \* 4 = G \* 1000 + O \* 110 + D\n";

for (T = 0; T < 10; T++) {

for (O = 0; O < 10; O++) {

if (T == O)

continue;

for (G = 0; G < 10; G++) {

if (T == G || G == O)

continue;

for (D = 0; D < 10; D += 2) {

if (D == T || D == O || T == G)

continue;

else if (T \* 100 \* 4 == (G \* 1000 + O \* 66 + D)) {

count++;

cout <<"경우 "<<count<< " : T = " << T << ", O = " << O << ", G = " << G << ", D = " << D << endl;

}

}

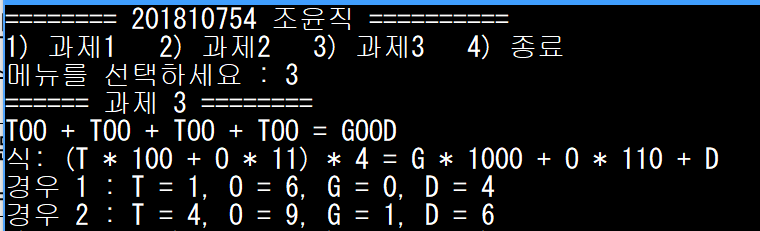
}

}

}

}

-실행결과



-느낀점 및 토의사항 : 다중 for문으로 처음에는 continue 문 없이 썼으나, 모든 알파벳의 값들이 중복 되는 값이 없으므로 각 for문이 시작할 때, 일치하는 값이 있을 경우 continue로 다음 반복으로 넘어 가게 해주었다.