

프로그래밍 기초 과제

7 주차 실습

학과	컴퓨터공학과
학번	2022111120
이름	김지민
담당교수님	한인 교수님
제출일자	2023.04.23

| 해당 주차 내용

- while 문: 반복 횟수가 정해져 있지 않고, 특정 조건이 만족될 때까지 반복한다.

형태 : 조건

→ (참이면) 문장 ⇒ 문장에는 조건을 중단시키는 문장이 있어야 한다.

→ 거짓이면 아웃 횟수가 정해진 반복은 while 로 쓰지 않는게 좋다.

무한루프 시 break 문을 이용하여 루프를 빠져 나오는데 사용한다.

break 는 자신을 감싸고 있는 딱 하나의 반복문만 벗어난다.

c 언어는 0 이 아닌 다른 수를 모두 true 로 본다.

while(1) 대신에 상수화하자 → #define TRUE 1 하고, while(TRUE)로 쓰는 걸 추천한다.

continue 는 다음 문장 체크 안하고, 다시 반복문의 조건을 체크하러 간다.

for 문에서는 continue 에서 증감식으로 갔다가, 조건식으로 간다.

| 실습문제 1

```
#include <stdio.h>
int main() {
    char cSelectmenu;
    int iInput1, iInput2;
    int iResult;

    while (1) {
        printf("*****\n");
        printf("A---- Add\n");
        printf("S---- Subtract\n");
        printf("M---- Multiply\n");
        printf("D---- Divide\n");
        printf("Q---- Quit\n");
        printf("*****\n");
        printf("연산을 선택하시오:");
        scanf(" %c", &cSelectmenu);

        if (cSelectmenu == 'A') {
            printf("두 수를 공백으로 분리하여 입력하시오:");
            scanf("%d %d", &iInput1, &iInput2);
            iResult = iInput1 + iInput2;
            printf("%d\n", iResult);
        }
        else if (cSelectmenu == 'S') {
            printf("두 수를 공백으로 분리하여 입력하시오:");
            scanf("%d %d", &iInput1, &iInput2);
            iResult = iInput1 - iInput2;
            printf("%d\n", iResult);
        }
        else if (cSelectmenu == 'M') {
            printf("두 수를 공백으로 분리하여 입력하시오:");
            scanf("%d %d", &iInput1, &iInput2);
            iResult = iInput1 * iInput2;
            printf("%d\n", iResult);
        }
    }
}
```

```

    }
    else if (cSelectmenu == 'D') {
        printf("두 수를 공백으로 분리하여 입력하시오:");
        scanf("%d %d", &iInput1, &iInput2);
        if (iInput2 == 0) {
            printf("0 으로 나눌 수 없습니다.\n");
        }
        else {
            iResult = iInput1 / iInput2;
            printf("%d\n", iResult);
        }
    }
    else if (cSelectmenu == 'Q') {
        break;
    }
    else {
        printf("연산을 선택하시오\n");
    }
}

return 0;
}

```

| 코드 설명 :

- 1) if-else 와 while 문을 통해 A,S,M,D,Q 가 아닌 다른 문자 및 숫자가 입력될 경우, 연산을 선택하시오라는 문구와 함께 다시 연산을 받는 문장으로 돌아갑니다.
- 2) if-else 를 이용하여 나누기의 경우 나누는 수가 0 이 될 경우 나눌 수 없다고 출력하며 다시 처음 화면으로 돌아갑니다.
- 3) 연산을 성공적으로 마치면 break 를 통해 반복문을 빠져나옵니다.

| 출력 결과

<pre> ***** A---- Add S---- Subtract M---- Multiply D---- Divide Q---- Quit ***** 연산을 선택하시오 : A 두 수를 공백으로 분리하여 입력하시오 :10 20 30 </pre>	<pre> A---- Add S---- Subtract M---- Multiply D---- Divide Q---- Quit ***** 연산을 선택하시오 :D 두 수를 공백으로 분리하여 입력하시오 :10 0 0으로 나눌 수 없습니다 . ***** </pre>	<pre> ***** A---- Add S---- Subtract M---- Multiply D---- Divide Q---- Quit ***** 연산을 선택하시오 :1 연산을 선택하시오 </pre>
---	--	---

| 실습문제 2

```
#define TRUE 1
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

int main() {
    int RandomNum;
    srand(time(0));
    int iInput;
    int istory, iend;
    int cnt;
    cnt=1;

    istory = 1;
    iend = 100;
    RandomNum = (rand() % iend) + istory;

    printf("컴퓨터가 당신이 생각하는 숫자를 알아맞히는 게임입니다.\n");
    printf("하나의 숫자를 생각하세요.\n");
    printf("컴퓨터가 제시한 숫자보다 정답이 높으면 1, 낮으면 -1 라고 하세요.\n");
    printf("컴퓨터가 숫자를 맞히면 0 이라고 하세요.\n");

    while (TRUE) {
        printf("숫자가 %d 인가요?", RandomNum);
        scanf("%d", &iInput);
        if (iInput == 0) {
            printf("당신이 생각한 숫자는 %d 입니다! \n", RandomNum);
            printf("컴퓨터가 %d 번만에 맞췄습니다.\n", cnt);
            break;
        }
        else if (iInput == -1) {
            iend = RandomNum - 1;
            RandomNum = (rand() % (iend-istory+1)) + istory;
            cnt=cnt+1;
        }
        else if (iInput == 1) {
            istory = RandomNum + 1;
            RandomNum = (rand()%(iend-istory+1))+istory;
            cnt=cnt+1;
        }
        else {
            printf("-1, 1, 0 사이의 숫자로 입력해주세요\n");
        }
    }
    return 0;
}
```

| 코드 설명 :

- 변수 설명

RandomNum	컴퓨터가 만들 랜덤숫자를 받는 변수
ilInput	사용자가 -1,1,0 을 입력할 변수
istart	난수 형성 시 범위의 하한선을 위해 만든 변수
iend	난수 형성 시 범위의 상한선을 위해 만든 변수
cnt	몇 번만에 맞췄는지 카운트하기 위해 만든 변수

1) while 문을 통한 반복과 if-else if 문을 통한 조건 확인

사용자가 생각한 숫자를 컴퓨터가 맞추기 전까지 반복되게 while 문을 사용합니다.

2) 원하는 범위 내의 난수 만들기

처음 시도를 할 때, 생각한 숫자가 50 이고 컴퓨터가 30 을 출력하면 다음에 컴퓨터가 생성해야 할 난수는 31 부터 100 입니다. 그리고 만약 60 을 출력했다면 그 다음 시도에 출력 해야 할 난수의 범위는 31 부터 59 입니다.

따라서 원하는 범위 내의 난수를 만들기 위해 (iend-istart+1)로 나눈 나머지에 + istart 를 더했습니다. 만약 입력 값이 -1 이라면 iend 값을 RandomNum-1 을 빼서 변화시켜주고, 1 이라면 istart 값을 RandomNum+1 을 통해 바꿔줍니다.

| 출력 결과

```
컴퓨터가 당신이 생각하는 숫자를 알아맞히는 게임입니다
하나의 숫자를 생각하세요.
컴퓨터가 제시한 숫자보다 정답이 높으면 1, 낮으면 -1라고 하세요.
컴퓨터가 숫자를 맞히면 0이라고 하세요.
숫자가 84 인가요?-1
숫자가 75 인가요?-1
숫자가 49 인가요?1
숫자가 74 인가요?-1
숫자가 66 인가요?-1
숫자가 60 인가요?-1
숫자가 59 인가요?-1
숫자가 57 인가요?-1
숫자가 50 인가요?0
당신이 생각한 숫자는 50 입니다!
컴퓨터가 9번만에 맞췄습니다.
gimjimin-ui-MacBook:김지민_2022111120_실습 7주차 kimjimin$
```

| 추가문제

```
//컴퓨터공학과 2022111120 김지민
// 6주차 실습 수정
//제출일 0416
#define TRUE 1
#include <stdio.h>
int main() {
    int inputAccount, inputPrice;
    int account1, account2, account3;
    account1 = -1;
    account2 = -1;
    account3 = -1;
    int account1_price, account2_price, account3_price;
    int selectMenu, outputPrice;

    while(TRUE) { //계속 반복
        printf("---menu---\n");
        printf("1. 계좌개설\n");
        printf("2. 입금\n");
        printf("3. 출금\n");
        printf("4. 계좌정보 전체 출력\n");
        printf("5. 프로그램 종료\n");
        printf("선택 : ");
        scanf_s("%d", &selectMenu);

        //1번 메뉴 선택할 경우
        // 잘못 만들어졌을 때 account1에 생성되면 안된다!!

        if (selectMenu == 1) {
            while (TRUE) {
                printf("[계좌개설]\n");
                printf("계좌 ID : ");
                scanf_s("%d", &inputAccount);
                printf("입금액 : ");
                scanf_s("%d", &inputPrice);

                if (account1 != inputAccount && account1 == -1) { //account1==-1 넣은
                    //이유는 3개만 만들기 위해서 안그러면 계속 갱신되서 무한개 만드니까
                    account1 = inputAccount;

                    if (inputPrice >= 0) {
                        account1_price = inputPrice;
                        printf("***계좌 개설 성공***\n");
                        printf("계좌 ID: %d\n", account1);
                        printf("잔액: %d \n\n", account1_price);
                        break;
                    }
                }
                else {
                    account1 = -1;
                    printf("0원 미만은 입금할 수 없습니다.\n");
                }
            }
        }
    }
}
```

```

}
else if (account2 != inputAccount && account2 == -1) {
    if (inputAccount != account1) {
        account2 = inputAccount;
        account2_price = inputPrice;
        if (inputPrice >= 0) {
            account2_price = inputPrice;
            printf("***계좌 개설 성공***\n");
            printf("계좌 ID: %d\n", account2);
            printf("잔액: %d \n\n", account2_price);
            break;
        }
    }
    else {
        account2 = -1;
        printf("0원 미만은 입금할 수 없습니다.\n");
    }
}
else {
    account2 = -1;
    printf("**계좌 개설 실패**\n");
    printf("이미 존재하는 아이디입니다\n\n");
}
}
else if (account3 != inputAccount && account3 == -1) {
    if (inputAccount != account1 && inputAccount != account2) {
        account3 = inputAccount;
        account3_price = inputPrice;
        if (inputPrice >= 0) {
            account3_price = inputPrice;
            printf("***계좌 개설 성공***\n");
            printf("계좌 ID: %d\n", account3);
            printf("잔액: %d \n\n", account3_price);
            break;
        }
    }
    else {
        account3 = -1;
        printf("0원 미만은 입금할 수 없습니다.\n");
    }
}
else {
    account3 = -1;
    printf("**계좌 개설 실패**\n");
    printf("이미 존재하는 아이디입니다\n");
}
}
else if (inputAccount == -1) {
    printf("\n**계좌 개설 실패**\n");
    printf("-1이 입력될 수 없습니다.\n");
}
else {
    printf("\n**계좌 개설 실패**\n");
    printf("이미 3개의 계좌가 존재합니다.\n\n");
    break;
}

```

```

    }
}

} //1번 선택 끝
//2번 선택 시작
if (selectMenu == 2) {
    while (TRUE) {
        printf("[입금]Wn");
        printf("계좌 ID :");
        scanf_s("%d", &inputAccount);
        printf("입금액 :");
        scanf_s("%d", &inputPrice);

        if (inputAccount == account1) {
            if (inputPrice > 0) {
                account1_price = account1_price + inputPrice;
                printf("**입금 성공!**Wn");
                printf("계좌 ID: %d Wn", inputAccount);
                printf("잔액: %dWn", account1_price);
                break;
            }
            else {
                printf("**입금 실패**Wn");
                printf("0이하의 금액은 입금할 수 없습니다.Wn");
            }
        }
        else if (inputAccount == account2) {
            if (inputPrice > 0) {
                account2_price = account2_price + inputPrice;
                printf("**입금 성공!**Wn");
                printf("계좌 ID: %d Wn", inputAccount);
                printf("잔액: %dWn", account2_price);
                break;
            }
            else {
                printf("**입금 실패**Wn");
                printf("0이하의 금액은 입금할 수 없습니다.Wn");
            }
        }
        else if (inputAccount == account3) {
            if (inputPrice > 0) {
                account3_price = account3_price + inputPrice;
                printf("**입금 성공!**Wn");
                printf("계좌 ID: %d Wn", inputAccount);
                printf("잔액: %dWn", account3_price);
                break;
            }
            else {
                printf("**입금 실패**Wn");
                printf("0이하의 금액은 입금할 수 없습니다.Wn");
            }
        }
    }
}

```



```

    }
    else {
        printf("**입금 실패**\n");
        printf("계좌ID가 존재하지 않습니다.\n");
        break;
    }
}
}
//3번 선택 시작
if (selectMenu == 3) {
    while (TRUE) {
        printf("[출금]\n");
        printf("계좌 ID :");
        scanf_s("%d", &inputAccount);
        printf("출금액 :");
        scanf_s("%d", &outputPrice);

        if (inputAccount == account1) {
            account1_price = account1_price - outputPrice;
            if (account1_price < 0) {
                printf("**출금 실패**\n");
                printf("잔액이 부족합니다.\n");
                account1_price = account1_price + outputPrice;
            }
            else {
                printf("**출금 성공!**\n");
                printf("계좌 ID: %d\n", account1);
                printf("잔액: %d\n", account1_price);
                break;
            }
        }
        else if (inputAccount == account2) {
            account2_price = account2_price - outputPrice;
            if (account2_price < 0) {
                printf("**출금 실패**\n");
                printf("잔액이 부족합니다.\n");

                account2_price = account2_price + outputPrice;
            }
            else {
                printf("**출금 성공!**\n");
                printf("계좌 ID: %d\n", account2);
                printf("잔액: %d\n", account2_price);
                break;
            }
        }
        else if (inputAccount == account3) {
            account3_price = account3_price - outputPrice;
            if (account3_price < 0) {
                printf("**출금 실패**\n");
                printf("잔액이 부족합니다.\n");

                account3_price = account3_price + outputPrice;
            }
        }
    }
}

```

```

    }
    else {
        printf("**출금 성공!**Wn");
        printf("계좌 ID: %d Wn", account3);
        printf("잔액: %d Wn", account3_price);
        break;
    }
}
else {
    printf("**출금 실패**Wn");
    printf("계좌아이디가 존재하지 않습니다.Wn");
    break;
}
}

//4번 선택 시작
if (selectMenu == 4) {
    if (account1 == -1) {
        printf("개설된 계좌가 없습니다Wn");
    }
    else {
        if (account1 != -1) {
            printf("계좌 ID: %dWn", account1);
            printf("잔액: %dWnWn", account1_price);
        } if (account2 != -1) {
            printf("계좌 ID: %dWn", account2);
            printf("잔액: %dWnWn", account2_price);
        } if (account3 != -1) {
            printf("계좌 ID: %dWn", account3);
            printf("잔액: %dWn", account3_price);
        }
    }
}

//5번 선택 시작
if (selectMenu == 5) {
    printf("프로그램을 종료합니다.");
    break;
}

return 0;
}

```

| 수정사항

저번에는 for 문을 이용하여 무한 반복을 실행했습니다. 하지만 이번에는 while 을 이용하여 무한 반복을 설정해줬습니다. #define TRUE 1 을 통해 TRUE 에 1 을 넣어주고, while 조건을 쓰는 공간에 TRUE 를 넣어줬습니다.