

11-1.c

The image shows a Visual Studio Code editor window with a C program named `11-1.c` open. The program defines a macro `INIT_ARRAY` and a `main` function that reads an integer `value` and prints an array of 10 elements initialized with `value`.

```
1  #define _CRT_SECURE_NO_WARNINGS
2  #include <stdio.h>
3  #define SIZE 10
4  #define INIT_ARRAY(array,size,value) for (int i=0;i<size;i++) array[i] = value+i; // value로 배열을 초기화하는 매크로 설정
5
6  int main(void) {
7      int a[SIZE];
8      int value;
9      printf("배열 초기화를 시작할 정수를 입력하십시오: ");
10     scanf("%d", &value);
11     INIT_ARRAY(a, SIZE, value); // 입력받은 value로 배열을 초기화
12     printf("초기화된 배열 a = [%d", a[0]);
13     for (int i = 1; i < SIZE; i++) printf(", %d", a[i]); // 배열 출력
14     printf("]");
15 }
```

The right sidebar shows the 'Solution Explorer' with the project structure:

- 과제11
 - 참조
 - 외부 종속성
 - 리소스 파일
 - 소스 파일
 - 11-1.c
 - 11-2.c
 - 헤더 파일

The bottom panel shows the 'Output' window with the build log:

```
출력 보기 선택(S): 빌드
1>----- 빌드 시작: 프로젝트: 과제11, 구성: Debug x64 -----
1>11-1.c
1>과제11.vcxproj -> C:\Users\chany\OneDrive - Ajou University\Mathematics\2023-2-2\컴퓨터프로그래밍및실습 (F024-1)\Visual Studio\11주차 실습\과제11\x64\Debug\과제11.exe
===== 빌드: 1개 성공, 0개 실패, 0개 최신 상태, 0개 건너뛴 =====
===== 빌드가(가) 오후 12:54에 완료되었으며, 00.952 초이(가) 걸림 =====
```

The status bar at the bottom indicates 'Build Succeeded' (빌드 성공).

11-1.c

The image shows a screenshot of the Microsoft Visual Studio IDE. The main editor window displays a C program named `11-1.c` with the following code:

```
1  #define _CRT_SECURE_NO_WARNINGS
2  #include <stdio.h>
3  #define SIZE 10
4  #define INIT_ARRAY(array,size,value) for (int i=0;i<size;i++) array[i] = value+i; // value로 배열을 초기화하는 매크로 설정
5
6  int main(void) {
7      int a[SIZE];
8      int value;
9      printf("배열 초기화를 시작할 정수를 입력하십시오: ");
10     scanf("%d", &value);
11     INIT_ARRAY(a, SIZE, value); // 입력받은 value로 배열을 초기화
12     printf("초기화된 배열 a = [%d", a[0]);
13     for (int i = 1; i < SIZE; i++) printf(", %d", a[i]); // 배열 출력
14     printf("]");
15 }
```

The right sidebar shows the Solution Explorer with the project structure:

- 과제11
 - 참조
 - 외부 종속성
 - 리소스 파일
 - 소스 파일
 - 11-1.c
 - 11-2.c
 - 헤더 파일

The Output window at the bottom shows the execution results:

```
배열 초기화를 시작할 정수를 입력하십시오: 2
초기화된 배열 a = [2, 3, 4, 5, 6, 7, 8, 9, 10, 11]
C:\Users\chany\OneDrive - Ajou University\Mathematics\2023-2-2\컴퓨터프로그래밍및실습 (F024-1)\Visual Studio\11주차 실습\과제11\x64\Debug\과제11.exe(프로세스 3580개)이(가) 종료되었습니다(코드: 0개).
이 창을 닫으려면 아무 키나 누르세요...
```

The status bar at the bottom indicates that the build was successful (빌드 성공).

11-2.c

The image shows a Visual Studio Code editor window with a C program named `11-2.c`. The program defines macros for bit manipulation and implements a `main` function that sets, checks, and clears bits.

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#define GET_BIT(n,pos) (((n) >> (pos)) & 0x01) // n의 pos번째 비트를 반환하는 매크로
#define SET_BIT(n,pos) ((n) |= (0x01 << (pos))) // n의 pos번째 비트를 1로 바꾸는 매크로
#define CLR_BIT(n,pos) ((n) &= ~(0x01 << (pos))) // n의 pos번째 비트를 0으로 바꾸는 매크로

int main(void) {
    int n = 0x00000e05; // n의 초기값 설정
    printf("n = %#010X\n", n); // n 출력
    int pos;
    printf("n에서 확인할 비트의 위치를 입력하시오(0 - 31에서 입력): ");
    scanf("%d", &pos);
    printf("GET_BIT(n,%d)의 결과: %#010X\n", pos, GET_BIT(n, pos)); // n의 pos번째 비트 출력
    printf("n에서 1로 설정할 비트의 위치를 입력하시오(0 - 31에서 입력): ");
    scanf("%d", &pos);
    SET_BIT(n, pos); // n의 pos번째 비트를 1로 변환
    printf("SET_BIT(n,%d)의 결과: %#010X\n", pos, n); // n 출력
    printf("n에서 0로 설정할 비트의 위치를 입력하시오(0 - 31에서 입력): ");
    scanf("%d", &pos);
    CLR_BIT(n, pos); // n의 pos번째 비트를 0으로 변환
    printf("CLR_BIT(n,%d)의 결과: %#010X\n", pos, n); // n 출력
    return 0;
}
```

The output window shows the successful execution of the program:

```
출력 보기 선택(S): 빌드
1>----- 빌드 시작: 프로젝트: 과제11, 구성: Debug x64 -----
1>11-2.c
1>과제11.vcxproj -> C:\Users\chany\OneDrive - Ajou University\Mathematics\2023-2-2\컴퓨터프로그래밍및실습 (F024-1)\Visual Studio\11주차_실습\과제11\11-2.c\Debug\과제11.exe
===== 빌드: 1개 성공, 0개 실패, 0개 최신 상태, 0개 건너뛴 =====
===== 빌드가(가) 오후 12:52에 완료되었으며, 00.986 초이(가) 걸림 =====
```

The status bar at the bottom indicates "빌드 성공" (Build Success).

11-2.c

The image shows a screenshot of the Microsoft Visual Studio IDE. The main window displays a C program named `11-2.c` with the following code:

```
1 #define _CRT_SECURE_NO_WARNINGS
2 #include <stdio.h>
3 #define GET_BIT(n,pos) (((n) >> (pos)) & 0x01) // n의 pos번째 비트를 반환하는 매크로
4 #define SET_BIT(n,pos) ((n) |= (0x01 << (pos))) // n의 pos번째 비트를 1로 바꾸는 매크로
5 #define CLR_BIT(n,pos) ((n) &= ~(0x01 << (pos))) // n의 pos번째 비트를 0으로 바꾸는 매크로
6
7 int main(void) {
8     int n = 0x00000e05; // n의 초기값 설정
9     printf("n = %#010X\n", n); // n 출력
10    int pos;
11    printf("n에서 확인할 비트의 위치를 입력하시오(0 - 31에서 입력): ");
12    scanf("%d", &pos);
13    printf("GET_BIT(n,%d)의 결과: %d\n", pos, GET_BIT(n, pos)); // n의 pos번째 비트 출력
14    printf("n에서 1로 설정할 비트의 위치를 입력하시오(0 - 31에서 입력): ");
15    scanf("%d", &pos);
16    SET_BIT(n, pos); // n의 pos번째 비트를 1로 변환
17    printf("SET_BIT(n,%d)의 결과: %#010X\n", pos, n); // n 출력
18    printf("n에서 0로 설정할 비트의 위치를 입력하시오(0 - 31에서 입력): ");
19    scanf("%d", &pos);
20    CLR_BIT(n, pos); // n의 pos번째 비트를 0으로 변환
21    printf("CLR_BIT(n,%d)의 결과: %#010X", pos, n); // n 출력
22    return 0;
23 }
```

The right sidebar shows the Solution Explorer with the project structure:

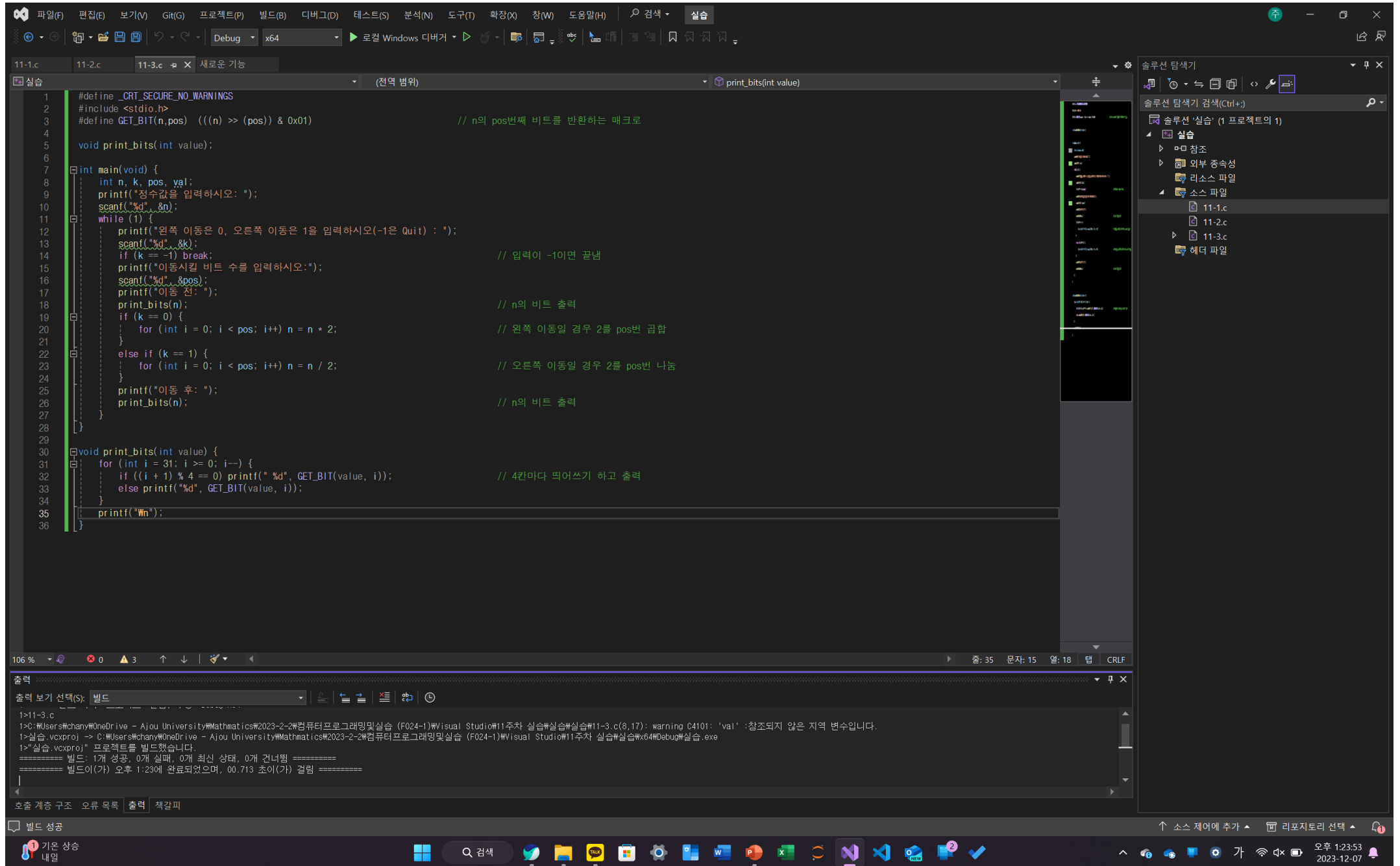
- 과제11
 - 마디 참조
 - 외부 종속성
 - 리소스 파일
 - 소스 파일
 - 11-1.c
 - 11-2.c
 - 헤더 파일

The bottom status bar shows the output window with the following text:

```
n = 0X00000E05
n에서 확인할 비트의 위치를 입력하시오(0 - 31에서 입력): 10
GET_BIT(n,10)의 결과: 1
n에서 1로 설정할 비트의 위치를 입력하시오(0 - 31에서 입력): 6
SET_BIT(n,6)의 결과: 0X00000E45
n에서 0로 설정할 비트의 위치를 입력하시오(0 - 31에서 입력): 9
CLR_BIT(n,9)의 결과: 0X00000C45
C:\Users\chany\OneDrive - Ajou University\Mathematics\2023-2-2\컴퓨터프로그래밍및실습 (F024-1)\Visual Studio\11주차 실습\
과제11\x64\Debug\과제11.exe(프로세스 22152개)이(가) 종료되었습니다(코드: 0개).
이 창을 닫으려면 아무 키나 누르세요...
```

The bottom status bar also shows the output window with the text "출력 보기 선택(S): 빌드" and "출력" button.

11-3.c



11-3.c

The image shows a Visual Studio IDE with a C program named 11-3.c. The program implements a bit shift operation and a bit printing function. The output window shows the execution results for two test cases.

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#define GET_BIT(n,pos) (((n) >> (pos)) & 0x01) // n의 pos번째 비트를 반환하는 매크로

void print_bits(int value);

int main(void) {
    int n, k, pos, val;
    printf("정수값을 입력하시오: ");
    scanf("%d", &n);
    while (1) {
        printf("왼쪽 이동은 0, 오른쪽 이동은 1을 입력하시오(-1은 Quit) : ");
        scanf("%d", &k);
        if (k == -1) break;
        printf("이동시킬 비트 수를 입력하시오:");
        scanf("%d", &pos);
        printf("이동 전: ");
        print_bits(n);
        if (k == 0) {
            for (int i = 0; i < pos; i++) n = n * 2;
            // 왼쪽 이동일 경우 2를 pos번 곱함
        }
        else if (k == 1) {
            for (int i = 0; i < pos; i++) n = n / 2;
            // 오른쪽 이동일 경우 2를 pos번 나눔
        }
        printf("이동 후: ");
        print_bits(n);
        // n의 비트 출력
    }
}

void print_bits(int value) {
    for (int i = 31; i >= 0; i--) {
        if ((i + 1) % 4 == 0) printf(" %d", GET_BIT(value, i));
        else printf("%d", GET_BIT(value, i));
        // 4칸마다 띄어쓰기 하고 출력
    }
    printf("\n");
}
```

Execution Output:

```
정수값을 입력하시오: 369
왼쪽 이동은 0, 오른쪽 이동은 1을 입력하시오(-1은 Quit) : 0
이동시킬 비트 수를 입력하시오: 8
이동 전: 0000 0000 0000 0000 0000 0001 0111 0001
이동 후: 0000 0000 0000 0001 0111 0001 0000 0000
왼쪽 이동은 0, 오른쪽 이동은 1을 입력하시오(-1은 Quit) : 1
이동시킬 비트 수를 입력하시오: 4
이동 전: 0000 0000 0000 0001 0111 0001 0000 0000
이동 후: 0000 0000 0000 0000 0001 0111 0001 0000
왼쪽 이동은 0, 오른쪽 이동은 1을 입력하시오(-1은 Quit) : -1
```

File Path: C:\Users\chany\OneDrive - Ajou University\Mathematics\2023-2-2\컴퓨터프로그래밍및실습 (F024-1)\Visual Studio\Debug\실습.exe(프로세스 23792개)이(가) 종료되었습니다(코드: 0개).

이 창을 닫으려면 아무 키나 누르세요...