

## 1. 다음에서 서술 내용이 맞으면 0, 틀리면 x 하시오.

- 1) 컴퓨터와 스마트폰 등의 정보기기에서 사용되는 프로그램을 만드는 사람을 프로그래머라 한다. ( o )
- 2) 소프트웨어는 크게 응용 소프트웨어와 시스템 소프트웨어로 나눈다. ( o )
- 3) 사람이 프로그래밍 언어로 컴퓨터에게 명령을 내리기 위해서는 프로그래밍 언어를 기계어로 변환하는 통역사인 운영체제가 필요하다. ( x )
- 4) 컴파일러는 고급언어로 작성된 프로그램을 기계어 또는 목적코드(object code)로 바꾸어 주는 프로그램이다. ( o )
- 5) 아스키코드는 전 세계의 모든 문자를 컴퓨터에서 일관되게 표현하고 다룰 수 있도록 설계된 산업 표준이다. ( x )
- 6) 주소는 변수이름과 같이 저장장소를 참조하는 하나의 방법이다. ( o )
- 7) \*가 피연산자인 변수의 메모리 주소를 반환하는 주소 연산자이다. ( x )
- 8) 어느 변수의 주소 값을 저장하려면 반드시 그 변수의 자료유형과 동일한 포인터 변수에 저장해야 한다. ( o )
- 9) int형 포인터에 20이라는 주소 값을 저장하려면 연산식 (int \*) 20을 사용해 저장한다. ( x )
- 10) int형 포인터 pi에 저장된 주소 값이 100이라면 (pi+1)은 101이다. ( x )
- 11) 포인터 변수는 자료 유형에 따라 그 크기가 다르다. ( o )
- 12) 연산식 ++\*p는 ++(\*p)으로 포인터 p가 가리키는 값을 1 증가시킨 후 참조한다. ( x )
- 13) 변수의 주소 값을 형식제어문자 %u 또는 %d로 직접 출력할 수 있다. ( o )
- 14) \*포인터변수는 l-value와 r-value로 모두 사용이 가능하나, 주소 값인 '&변수'는 r-value로만 사용이 가능하다. ( o )
- 15) 간접연산자 \*는 후위 증감연산자 a++보다 우선순위가 높다. ( o )

## 2. 다음 각각의 문제에서 가장 적절한 것을 하나 선택하십시오.

- ① 다음 중 정보 처리 단위가 작은 것부터 순서로 나열된 것은?            나)
  - 가) 바이트, 니블, 비트, 워드
  - 나) 비트, 니블, 워드, 바이트
  - 다) 비트, 니블, 바이트, 워드
  - 라) 워드, 비트, 니블, 바이트
- ② 다음 중에서 C 언어에 대한 설명이 아닌 것은?            다)
  - 가) 이식성이 좋음
  - 나) 메모리의 할당과 해제
  - 다) 객체지향 언어
  - 라) 다양한 분야에 사용
- ③ 다음 중에서 C 언어에 배워야 하는 이유로 가장 관련성이 적은 것은?            라)
  - 가) 자바나 C#, Objective-C 등이 그 뿌리는 C이므로 그 이후의 언어들은 그 습득이 매우 쉬워진다.
  - 나) C 언어는 범용적인 프로그래밍 언어로 다양한 분야에 활용된다.
  - 다) C 언어의 학습을 통해 단순히 C 언어의 문법 학습뿐 아니라 일반적인 프로그래밍 기초 지식을 학습한다.
  - 라) 자바와 Objective-C와 같이 객체와 클래스의 활용에도 선행 학습이 된다.
- ④ 다음 중에서 2진수의 특징으로 잘못된 것은?            라)
  - 가) 2진수는 수의 자릿수에 사용할 수 있는 숫자가 0과 1, 2개 이므로 이진수라 한다.
  - 나) 디지털 신호에서 전기가 흐를 경우 '참'을 의미하는 '1', 흐르지 않을 경우 '거짓'의 '0'으로 표현된다.
  - 다) 컴퓨터는 논리의 조합이 간단하고 내부에 사용되는 소자의 특성상 편리하기 때문에 이진법을 사용하는 것이 가장 합리적이고 효율적인 방식이다.
  - 라) 2진수에서 수의 길이는 상대적으로 8진수보다 짧다.

3. 반지름이 7.58인 원의 면적과 둘레와 길이를 구하는 프로그램을 작성하시오.

- 상수로 원주율 PI를 3.14로 정의
- 면적공식: 반지름 \* 반지름 \* 3.14(원주율)
- 둘레공식: 2 \* 3.14(원주율) \* 반지름

```
#include <stdio.h>
int main(){
    const double PI =3.14;

    double area =0;
    double perimeter =0;
    double radius =7.58;
    area =radius *radius *PI;
    perimeter =2 *PI *radius;
    printf("area: %lf\nperimeter: %lf", area, perimeter);
}
```

•

4. 아파트 면적인 18평, 25평, 32평, 44평, 52평을 제곱미터(m<sup>2</sup>)의 면적으로 출력하는 프로그램을 작성하시오.

- 1 평은 3.305785 제곱미터(m<sup>2</sup>)

```

#include <stdio.h>
int main(){
    int apt[]={18, 25, 32, 44, 52};
    const double ratio =3.305785;
    for(int i =0; i <sizeof(apt)/sizeof(int); i++){
        printf("%dp = %.2lfm2\n", apt[i], apt[i] *ratio);
    }
}

```

5. **scanf로 숫자를 입력 받아** 아래와 같이 소수가 출력 되도록 프로그램을 작성 한다.  
scanf로 입력 받은 숫자가 999이면 프로그램의 실행을 종료 한다.

=====

2이상의 숫자를 입력 하세요(종료:999)> 100

2    3    5    7    11

13   17   19   23   29

31   37   41   43   47

53   59   61   67   71

73   79   83   89   97

2이상의 숫자를 입력 하세요(종료:999)>

=====

```

#include <stdio.h>
int main(){
    int input =0;
    int count =0;
    int count2 =0;
    while(1){
        printf("2이상의 숫자를 입력하세요(종료:999)> ");
        scanf("%d", &input);
    }
}

```

```
switch(input){
    case 999:
        return 0;

    default:
        break;
}

for(int i =2; i <=input; i++){
    for(int n =1; n <=i; n++){
        if(i %n ==0) count++;
    }

    if(count ==2){
        printf("%2d", i);
        if(++count2 %5 ==0) printf("\n");
    }

    count =0;
}
}
}
```

6. 아래와 같이 출력 되도록 아래의 source code를 완성 하시오

```

ABCDEFGHIJKLMNOPQRSTUVWXYZ
BCDEFGHIJKLMNOPQRSTUVWXYZA
CDEFGHIJKLMNOPQRSTUVWXYZAB
DEFGHIJKLMNOPQRSTUVWXYZABC
EFGHIJKLMNOPQRSTUVWXYZABCD
FGHIJKLMNOPQRSTUVWXYZABCDE
GHIJKLMNOPQRSTUVWXYZABCDEF
HIJKLMNOPQRSTUVWXYZABCDEFG
IJKLMNOPQRSTUVWXYZABCDEFGH
JKLMNOPQRSTUVWXYZABCDEFGHI
KLMNOPQRSTUVWXYZABCDEFGHIJ
LMNOPQRSTUVWXYZABCDEFGHIJK
MNOPQRSTUVWXYZABCDEFGHIJKL
NOPQRSTUVWXYZABCDEFGHIJKLM
OPQRSTUVWXYZABCDEFGHIJKLMN
PQRSTUVWXYZABCDEFGHIJKLMNO
QRSTUVWXYZABCDEFGHIJKLMNOP
RSTUVWXYZABCDEFGHIJKLMNOPQ
STUVWXYZABCDEFGHIJKLMNOPQR
TUVWXYZABCDEFGHIJKLMNOPQRS
UVWXYZABCDEFGHIJKLMNOPQRST
VWXYZABCDEFGHIJKLMNOPQRSTU
WXYZABCDEFGHIJKLMNOPQRSTUV
XYZABCDEFGHIJKLMNOPQRSTUVW
YZABCDEFGHIJKLMNOPQRSTUVWX
ZABCDEFGHIJKLMNOPQRSTUVWXY
ABCDEFGHIJKLMNOPQRSTUVWXYZ
```

```
int main(void)
{
    char alpha[] = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
    shift_alpha(alpha);
    return 0;
}
```

```
void shift_alpha(char *alpha){
    char saved;
    for(int i =0; i <=26; i++){
```

```

        saved =alpha[0];
        for(int n =0; n <26; n++){
            printf("%c", alpha[n]);
            alpha[n] =alpha[n +1];
        }

        printf("\n");
        alpha[25] =saved;
    }
}

```

7. 표준입력(scanf)로 두 정수를 입력 받아 큰 수를 작은 수로 나눈 몫과 나머지를 각각 출력하는 프로그램을 작성하시오.

```

#include <stdio.h>
int main(){
    int num1 =0;
    int num2 =0;
    int temp =0;
    printf("Enter num1: ");
    scanf("%d", &num1);
    printf("Enter num2: ");
    scanf("%d", &num2);
    if(num1 <num2){
        temp =num1;
        num1 =num2;
        num2 =temp;
    }
}

```

```
printf("%d%d%d\n", num1, num2, num1 /num2);
printf("%d %%%d%d\n", num1, num2, num1 %num2);
}
```

8. 다음을 출력하는 프로그램을 for 문을 이용하여 작성하시오.

```
54321
5432
543
54
5
```

```
#include <stdio.h>
int main(){
    for(int i =0; i <5; i++){
        for(int n =5; n >i; n--){
            printf("%d", n);
        }

        printf("\n");
    }
}
```

9. 다음을 출력하는 프로그램을 중첩된 for 문을 이용하여 작성하시오.

```
0
101
21012
3210123
432101234
54321012345
6543210123456
765432101234567
```



```
#include <stdio.h>
int main(){
    for(int i =0; i <8; i++){
        for(int n =7; n >=0; n--){
            if(n <=i) printf("%d", n);
            else printf(" ");
        }

        for(int n =1; n <=i; n++){
            printf("%d", n);
        }

        printf("\n");
    }
}
```



10. 배열을 사용하여 98.56, 78.62, 78.69, 89.32, 95.29를 초기화하여 출력하고 배열의 총합과 평균을 구하여 출력하는 프로그램을 작성하시오.

```
int main(){
    double nums[]={98.56, 78.62, 89.32, 95.29};
    double sum =0;
    for(int i =0; i <sizeof(nums)/sizeof(int); i++){
        sum +=nums[i];
    }
    printf("sum      = %.2lf\naverage  = %.2lf\n",    sum,    sum
/(sizeof(nums)/sizeof(int)));
}
```

11. 0에서 9까지의 정수 중에서 20개의 수를 입력 받아 가장 많이 입력 받은 빈도 수는 무엇이고, 몇 번인지 출력하는 프로그램을 작성하시오.

```
#include <stdio.h>
int main(){
    int count[9] ={0,};
    int input =0;
    int most =0;
```

```
printf("Enter 20 numbers: ");

for(int i =0; i <20; i++){
    scanf("%d", &input);
    if(input <0 ||input >9){
        printf("Invalid number.\n");
        return 0;
    }

    count[input]++;
}

for(int i =0; i <sizeof(count)/sizeof(int); i++){
    if(count[most] <count[i]) most =i;
}

printf("Frequency: %d\nNumber: %d\n", count[most], most);
}
```

12. 1)아래#1는 구조체 배열을 이용 하여 작성한 은행계좌관리 프로그램이다. source program은 구조체배열로 작성한 코드이다.

2) 아래#1의 코드를 그대로 실행을 해본후

3) 아래#2와 같이

전역변수 t\_account acc\_arr[100]; // Account 저장을 위한 배열  
int acc\_num=0; // 저장된 Account 수를 main함수 안으로 옮긴후  
구조체 배열을로 된 부분을 C언어의 구조체 포인터를 활용 하여  
프로그램을 작성 하시오

```
#include <stdio.h>
#include <string.h>
#define NAME_LEN 20

void show_menu(void); // 메뉴출력
void make_account(void); // 계좌개설을 위한 함수
void deposit_money(void); // 입 금
void with_draw_money(void); // 출 금
void show_all_acc_info(void); // 잔액조회

enum {MAKE=1, DEPOSIT, WITHDRAW, INQUIRE, EXIT=9};

typedef struct
{
    int acc_id; // 계좌번호
    int balance; // 잔 액
    char cus_name[NAME_LEN]; // 고객이름
} t_account;

t_account acc_arr[100]; // Account 저장을 위한 배열
int acc_num=0; // 저장된 Account 수

int main() // int main(argc, char *argv[])
{
    int choice;

    while(1)
    {
        show_menu();
```

```

        printf("선택: ");
        scanf("%d", &choice);
        printf("\n");

        switch(choice)
        {
        case MAKE:
            make_account();
            break;
        case DEPOSIT:
            deposit_money();
            break;
        case WITHDRAW:
            with_draw_money();
            break;
        case INQUIRE:
            show_all_acc_info();
            break;
        case EXIT:
            return 0;
        default:
            printf("Illegal selection..\n");
        }
    }
    return 0;
}

void show_menu(void)
{
    char *menu[] =
    {
        "-----Menu-----\n",
        "1. 계좌개설\n",
        "2. 입      금\n",
        "3. 출      금\n",
        "4. 계좌정보 전체 출력\n",
        "9. 종      료\n"
    };
    int i;

    for (i=0; i < 6; i++)
        printf("%s", *(menu+i));
}

void make_account(void)
{
    int id;
    char name[NAME_LEN];
    int balance;

    printf("[계좌개설]\n");
    printf("계좌ID: ");
    scanf("%d", &id);
    printf("이   름: ");
    scanf("%s", name);
    printf("입금액: ");
    scanf("%d", &balance);
    printf("\n");

    acc_arr[acc_num].acc_id=id;
    acc_arr[acc_num].balance=balance;
    strcpy(acc_arr[acc_num].cus_name,name);
    acc_num++;
}

void deposit_money(void)
{
    int money;
    int id, i;

```

```

    printf("[입금]\n");
    printf("계좌ID: ");
    scanf("%d", &id);
    printf("입금액: ");
    scanf("%d", &money);

    for (i=0; i<acc_num; i++)
    {
        if(acc_arr[i].acc_id==id)
        {
            acc_arr[i].balance += money;
            printf("입금완료\n\n");
            return;
        }
    }
    printf("유효하지 않은 ID 입니다.\n\n");
}

void with_draw_money(void)
{
    int money;
    int id, i;

    printf("[출금]\n");
    printf("계좌ID: ");
    scanf("%d", &id);
    printf("출금액: ");
    scanf("%d", &money);

    for(i=0; i<acc_num; i++)
    {
        if(acc_arr[i].acc_id==id)
        {
            if(acc_arr[i].balance < money)
            {
                printf("잔액부족\n\n");
                return;
            }

            acc_arr[i].balance -= money; // acc_arr[i].balance =
acc_arr[i].balance - money;
            printf("출금완료\n\n");
            return;
        }
    }
    printf("유효하지 않은 ID 입니다.\n\n");
}

void show_all_acc_info(void)
{
    int i;

    for(i=0; i<acc_num; i++)
    {
        printf("계좌ID: %d\n", acc_arr[i].acc_id);
        printf("이름: %s\n", acc_arr[i].cus_name);
        printf("잔액: %d\n\n", acc_arr[i].balance);
    }
}

```

=== 아래#2 ===

```

#include <stdio.h>
#include <string.h>
#define NAME_LEN 20

```

```

void show_menu(void);           // 메뉴출력
void make_account(t_account *t, int *n); // 계좌개설을 위한 함수
void deposit_money(t_account *t, int *n); // 입      금
void with_draw_money(t_account *t, int *n); // 출      금
void show_all_acc_info(t_account *t, int *n); // 잔액조회

enum {MAKE=1, DEPOSIT, WITHDRAW, INQUIRE, EXIT=9};

typedef struct
{
    int acc_id; // 계좌번호
    int balance; // 잔      액
    char cus_name[NAME_LEN]; // 고객이름
} t_account;

int main() // int main(argc, char *argv[])
{
    int choice;
    t_account acc_arr[100]; // Account 저장을 위한 배열
    int acc_num=0; // 저장된 Account 수

    while(1)
    {
        show_menu();
        printf("선택: ");
        scanf("%d", &choice);
        printf("\n");

        switch(choice)
        {
            case MAKE:
                make_account(acc_arr, &acc_num);
                break;
            case DEPOSIT:
                deposit_money(acc_arr, &acc_num);
                break;
            case WITHDRAW:
                with_draw_money(acc_arr, &acc_num);
                break;
            case INQUIRE:
                show_all_acc_info(acc_arr, &acc_num);
                break;
            case EXIT:
                return 0;
            default:
                printf("Illegal selection..\n");
        }
    }
    return 0;
}

```

```

void show_menu(void)
{
    char *menu[]=
    {
        "-----Menu-----\n",
        "1. 계좌개설\n",
        "2. 입      금\n",
        "3. 출      금\n",
    }
}

```

```

        "4. 계좌정보 전체 출력\n",
        "9. 종    료\n"
    int i;

    for(i=0; i <6; i++)
        printf("%s", *(menu+i));
}

void make_account(t_account *t, int *n)
{
    int id;
    char name[NAME_LEN];
    int balance;

    printf("[계좌개설]\n");
    printf("계좌ID: ");
    scanf("%d", &id);
    printf("이   름: ");
    scanf("%s", name);
    printf("입금액: ");
    scanf("%d", &balance);
    printf("\n");
    t[*n].acc_id =id;
    strcpy(t[*n].cus_name, name);
    t[*n].balance =balance;
    (*n)++;
}

void deposit_money(t_account *t, int *n)
{
    int money;
    int id, i;

    printf("[입   금]\n");
    printf("계좌ID: ");
    scanf("%d", &id);
    printf("입금액: ");
    scanf("%d", &money);

    for(i=0; i<*n; i++)
{

```

```

        if(t[i].acc_id==id)
    {
        t[i].balance +=money;
        printf("입금완료\n\n");
        return;
    }
}

printf("유효하지 않은 ID 입니다.\n\n");
}

void with_draw_money(t_account *t, int *n)
{
    int money;
    int id, i;

    printf("[출금]\n");
    printf("계좌ID: ");
    scanf("%d", &id);
    printf("출금액: ");
    scanf("%d", &money);

    for(i=0; i<*n; i++)
    {
        if(t[i].acc_id==id)
        {
            if(t[i].balance < money)
            {
                printf("잔액부족\n\n");
                return;
            }

            t[i].balance -=money; // acc_arr[i].balance =
acc_arr[i].balance - money;
            printf("출금완료\n\n");
            return;
        }
    }

    printf("유효하지 않은 ID 입니다.\n\n");
}

void show_all_acc_info(t_account *t, int *n)

```



```
{  
    int i;  
  
    for(i=0; i<*n; i++)  
    {  
        printf("계좌ID: %d\n", t[i].acc_id);  
        printf("이름: %s\n", t[i].cus_name);  
        printf("잔액: %d\n\n", t[i].balance);  
    }  
}
```