

1. C 언어에서 정적(static) 변수에 대한 설명으로 옳은 것을 고르시오. [5pts]

- ① 정적 변수는 함수 호출 시마다 메모리가 새로 할당된다.
- ② 정적 변수는 전역 변수처럼 프로그램 전체에서 접근 가능하다.
- ③ 정적 변수는 초기화하지 않으면 자동으로 0으로 초기화된다..
- ④ 정적 변수는 힙(heap) 영역에 저장된다.

2. C 언어의 메모리 구조에서 힙(Heap) 영역에 대한 설명으로 옳은 것을 고르시오. [5pts]

- ① 힙 영역은 정적 메모리 할당 방식이다.
- ② 메모리 할당이 주소가 높은 쪽에서 낮은 쪽으로 이루어진다.
- ③ 힙 영역의 메모리 사용이 끝난 경우 자동으로 해제된다.
- ④ 사용자가 직접 메모리를 할당하고 해제해야 한다.

3. C 언어에 대한 설명으로 옳은 것을 모두 고르시오. [5pts]

- ① return 문은 함수의 실행을 종료하고 값을 반환한다.
- ② C 언어는 함수 포인터를 지원하지 않는다.
- ③ extern 키워드는 변수의 선언을 여러 파일에서 공유할 수 있게 한다.
- ④ typedef 키워드는 새로운 데이터 타입을 정의하는 데 사용된다.

4. C 언어에서 동적 메모리 할당과 관련 함수에 대한 설명으로 옳은 것을 모두 고르시오. [5pts]

- ① 동적으로 할당된 메모리는 명시적으로 해제하지 않으면 프로그램이 종료될 때 자동으로 해제된다.
- ② malloc() 함수는 메모리 할당에 실패하면 NULL을 반환한다.
- ③ calloc() 함수는 메모리 블록을 할당하고 0으로 초기화한다.
- ④ free() 함수는 스택(stack) 영역에 할당된 메모리를 해제한다.

5. 다음 C 프로그램을 실행했을 때 출력 결과를 작성하시오. [10pts]

```
#include <stdio.h>

int main()
{
    int arr[3] = {1, 3, 5};
    int i, sum = 0;
    for (i = 0; i < 3; i++)
    {
        sum += (arr[i] & i + 1) ^ (arr[i] | i * 2);
    }
    printf("%d\n", sum);
}
```

6. 다음 C 프로그램을 실행했을 때 출력 결과를 작성하시오. [10pts]

```
#include <stdio.h>

int f(int n)
{
    if (n == 0)
        return 0;
    if (n % 2 == 0)
        return n % 10 + 2 * f(n / 10);
    else
        return n % 10 + 3 * f(n / 10);
}

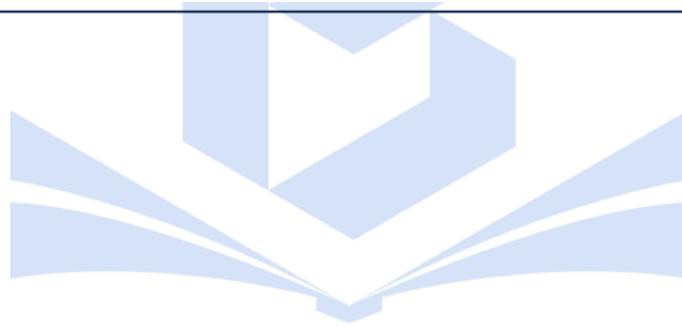
int main()
{
    int result = f(5234);
    printf("%d\n", result);
    return 0;
}
```

7. 다음 C 프로그램을 실행했을 때 출력 결과를 작성하시오. [15pts]

```
#include <stdio.h>

void f(int *n)
{
    static int x = 2;
    x += *n;
    *n -= x;
    printf("%d ", x);
}

int main()
{
    int a = 3, b = 5;
    for (int i = 0; i < 4; i++)
    {
        if (i % 2 == 0)
            f(&a);
        else
            f(&b);
    }
    return 0;
}
```



8. 다음 C 프로그램을 실행했을 때 출력 결과를 작성하시오. [15pts]

```
#include <stdio.h>

struct Data
{
    int value;
    int *ptr;
};

void update(struct Data arr[], int size)
{
    for (int i = 0; i < size; i++)
    {
        *(arr[i].ptr) += arr[i].value;
    }
}

int main()
{
    int values[] = {10, 20, 30};
    struct Data arr[3];

    for (int i = 0; i < 3; i++)
    {
        arr[i].value = i + 1;
        arr[i].ptr = &values[2 - i];
    }

    update(arr, 3);

    for (int i = 0; i < 3; i++)
    {
        printf("%d ", values[i]);
    }

    return 0;
}
```

9. 다음 C 프로그램의 실행 결과가 “5 12 11 6”이 될 수 있도록 (A)에 들어갈 알맞은 코드를 작성하시오. [15pts]

```
#include <stdio.h>

int add(int a, int b)
{
    return a + b;
}

int multiply(int a, int b)
{
    return a * b;
}
```

(A)

```
int main()
{
    execute(add, 2, 3);
    execute(multiply, 3, 4);
    execute(add, 5, 6);
    execute(multiply, 2, 3);
    return 0;
}
```

10. 다음 C 프로그램은 2차원 3*3 배열을 입력받아 시계 방향으로 90도 회전하여 출력하는 `rotateArray` 함수가 구현되어 있다. 프로그램이 올바르게 동작할 수 있도록 (A)에 들어갈 알맞은 코드를 작성하시오, [15pts]

ex.

1	2	3	→	7	4	1
4	5	6		8	5	2
7	8	9		9	6	3

```
#include <stdio.h>
#define LEN 3

void rotateArray(int arr[][3])
{
    (A)
}

int main()
{
    int arr[LEN][LEN];
    for (int i = 0; i < LEN; i++)
    {
        for (int j = 0; j < LEN; j++)
        {
            scanf("%d", &arr[i][j]);
        }
    }
    rotateArray(arr);
    return 0;
}
```

본 자료는 BLIS 편입에서 수험생들을 위해 만든 자료로써, 무단배포 및 상업적 이용을 금합니다.

<https://blis.co.kr/>