Q1. What is the purpose of the main() function in a C program? Explain its significance.
The `main()` function in C is the starting point of every program. When the program is executed, control always begins at `main()`. It defines where the execution starts and where it ends. Without it, a C program cannot run. The return value of `main()` is also significant because returning `0` generally indicates that the program executed successfully, while other values may signal errors.

Q2. Explain the difference between a variable declaration and a variable initialization in C.
A variable declaration specifies the type and name of a variable but does not assign it a value. For example, `int x;` declares an integer variable named `x`. Initialization, however, assigns a value at the time of declaration, such as `int x = 10;`. Declaration reserves memory space, while initialization ensures that the variable starts with a specific value.

Q4. What are the different data types available in C? Provide examples of each data type.
C provides several data types for handling different kinds of data:

- **int**: Used to store integers (e.g., `int age = 20;`).
- **float**: Stores single-precision floating-point numbers (e.g., `float salary = 5000.75;`).
- **double**: Stores double-precision floating-point numbers (e.g., `double pi = 3.14159;`).
- **char**: Holds single characters (e.g., `char grade = 'A';`).
- **void**: Represents no value, commonly used as a return type for functions that do not return anything.

There are also variations such as `short`, `long`, `unsigned int`, and `long double` that allow different ranges and levels of precision.

Q5. Explain the concept of type conversions in C. Provide examples of implicit and explicit type conversions.
Type conversion in C means changing a variable from one data type to another. There are two kinds:

Implicit conversion (type promotion): Done automatically by the compiler when assigning one type to another with a larger capacity. For example:

```
int a = 5;
double b = a;  // implicit conversion from int to double
```

Explicit conversion (type casting): Performed manually by the programmer using a cast operator. For example:

```
double pi = 3.14;
int x = (int) pi;  // explicit conversion from double to int
```

Implicit conversion is safe and prevents data loss when possible, while explicit casting gives the programmer full control, even if information might be lost.

Q7. What is the role of the scanf() function in C? Provide an example of its usage.

The `scanf()` function is used to take input from the user. It reads data from the keyboard and stores it in variables according to the specified format. For example:

```
#include <stdio.h>

int main() {
    int age;
    printf("Enter your age: ");
    scanf("%d", &age);  // reads an integer from the user
    printf("You entered: %d", age);
    return 0;
}
```

Here, the `%d` format specifier tells the program it's an integer, and `&age` provides the address where the input should be stored.