

- [Reading Comprehension](#)
 - [Reading Comprehension answers](#)
 - [Lab exercises](#)
 - [Lab exercises answers](#)

Reading Comprehension

C Programming Journey

Here are some questions to deepen your understanding of control statements in C:

1. If Statements:

- What is the purpose of an `if` statement in C?
- How does an `if` statement evaluate conditions?
- Can you provide an example of a simple `if` statement in C?
- What happens if the condition inside the `if` statement is true? What if it's false?

2. If-Else Statements:

- How does an `if-else` statement differ from an `if` statement?
- Can you explain the flow of control in an `if-else` statement?
- Provide an example where an `if-else` statement is used in C.
- What happens when the condition in the `if` part is true? What about the `else` part?

3. Nested If-Else Statements:

- What is a nested `if-else` statement?
- When would you use nested `if-else` statements?
- Provide an example of a nested `if-else` statement in C.
- How do you manage multiple levels of nesting in `if-else` statements?

4. If-Else-If Statements:

- What is the purpose of `if-else-if` statements in C?
- How does the flow of control work in `if-else-if` statements?
- Provide an example where `if-else-if` statements are used.
- What is the significance of the `else` part in `if-else-if` statements?

Reading Comprehension answers

1. If Statements:

- The purpose of an **if** statement in C is to execute a block of code if a specified condition is true.
- An **if** statement evaluates conditions by checking whether the given condition is true or false. If it's true, the code inside the **if** block executes; otherwise, it's skipped.
- Example:

```
int num = 10;
if (num > 0) {
    printf("The number is positive.\n");
}
```

- If the condition inside the **if** statement is true, the code inside the **if** block executes. If it's false, the program continues to the next statement after the **if** block.

2. If-Else Statements:

- An **if-else** statement differs from an **if** statement in that it provides an alternative code block to execute when the condition is false.
- In an **if-else** statement, if the condition is true, the code inside the **if** block executes; otherwise, the code inside the **else** block executes.
- Example:

```
int num = 10;
if (num % 2 == 0) {
    printf("The number is even.\n");
} else {
    printf("The number is odd.\n");
}
```

- If the condition in the **if** part is true, the code inside the **if** block executes. If it's false, the code inside the **else** block executes.

3. Nested If-Else Statements:

- A nested **if-else** statement is an **if-else** statement inside another **if** or **else** block.
- Nested **if-else** statements are used when multiple conditions need to be checked sequentially.
- Example:

```
int num = 10;
if (num > 0) {
    if (num % 2 == 0) {
        printf("The number is positive and even.\n");
    } else {
        printf("The number is positive and odd.\n");
    }
} else {
    printf("The number is non-positive.\n");
}
```

- Multiple levels of nesting can be managed by indenting the code properly and ensuring each **if** or **else** block corresponds to its parent block.

4. If-Else-If Statements:

- The purpose of **if-else-if** statements in C is to provide multiple conditions to be evaluated sequentially.
- The flow of control in **if-else-if** statements proceeds from top to bottom, and the code block associated with the first true condition is executed.
- Example:

```
int num = 10;
if (num > 0) {
    printf("The number is positive.\n");
} else if (num < 0) {
    printf("The number is negative.\n");
} else {
    printf("The number is zero.\n");
}
```

- The **else** part serves as a fallback option if none of the preceding conditions are true. It ensures that at least one code block executes.

Lab exercises

1. If Statements:

- Write a program in C to check if a given number is positive, negative, or zero using the **if** statement.
- Create a program in C to determine if a student has passed or failed an exam based on their score using the **if** statement.
- Implement a C program to find the maximum of two numbers using the **if** statement.

2. If-Else Statements:

- Develop a C program to determine whether a given year is a leap year or not using the **if-else** statement.
- Write a C program to check if a person is eligible to vote based on their age using the **if-else** statement.
- Create a program in C to determine the grade of a student based on their marks using the **if-else** statement.

3. Nested If-Else Statements:

- Implement a C program to find the largest of three numbers using nested **if-else** statements.
- Write a program in C to check if a given character is an alphabet, digit, or special character using nested **if-else** statements.
- Develop a C program to determine the quadrant of a given coordinate point using nested **if-else** statements.

4. If-Else-If Statements:

- Create a C program to classify a given angle as acute, obtuse, or right using **if-else-if** statements.
- Write a program in C to find the roots of a quadratic equation using **if-else-if** statements.

- Implement a C program to print the name of a month based on its number using `if-else-if` statements.

5. Switch Case:

- Develop a C program to print the name of a day of the week based on its number using the `switch` statement.
- Write a program in C to perform arithmetic operations (addition, subtraction, multiplication, division) based on user input using the `switch` statement.
- Implement a C program to convert a given number into words (e.g., 1 as "One", 2 as "Two") using the `switch` statement.

Lab exercises answers

1. If Statements:

- Program to check if a number is positive, negative, or zero:

```
#include <stdio.h>

int main() {
    int num;
    printf("Enter a number: ");
    scanf("%d", &num);

    if (num > 0)
        printf("%d is positive.\n", num);
    else if (num < 0)
        printf("%d is negative.\n", num);
    else
        printf("The number is zero.\n");

    return 0;
}
```

- Program to determine if a student has passed or failed an exam:

```
#include <stdio.h>

int main() {
    int score;
    printf("Enter the student's score: ");
    scanf("%d", &score);
```

```

if (score >= 50)
    printf("Student has passed.\n");
else
    printf("Student has failed.\n");

return 0;
}

```

- Program to find the maximum of two numbers:

```

#include <stdio.h>

int main() {
    int num1, num2;
    printf("Enter two numbers: ");
    scanf("%d %d", &num1, &num2);

    if (num1 > num2)
        printf("Maximum: %d\n", num1);
    else
        printf("Maximum: %d\n", num2);

    return 0;
}

```

2. If-Else Statements:

- Program to determine whether a given year is a leap year or not:

```

#include <stdio.h>

int main() {
    int year;
    printf("Enter a year: ");
    scanf("%d", &year);

    if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0))
        printf("%d is a leap year.\n", year);
    else
        printf("%d is not a leap year.\n", year);

    return 0;
}

```

- Program to check if a person is eligible to vote based on their age:

```
#include <stdio.h>

int main() {
    int age;
    printf("Enter your age: ");
    scanf("%d", &age);

    if (age >= 18)
        printf("You are eligible to vote.\n");
    else
        printf("You are not eligible to vote.\n");

    return 0;
}
```

- Program to determine the grade of a student based on their marks:

```
#include <stdio.h>

int main() {
    int marks;
    printf("Enter the student's marks: ");
    scanf("%d", &marks);

    if (marks >= 90)
        printf("Grade: A\n");
    else if (marks >= 80)
        printf("Grade: B\n");
    else if (marks >= 70)
        printf("Grade: C\n");
    else if (marks >= 60)
        printf("Grade: D\n");
    else
        printf("Grade: F\n");

    return 0;
}
```

3. Nested If-Else Statements:

- Program to find the largest of three numbers:

```
#include <stdio.h>

int main() {
    int num1, num2, num3, max;
    printf("Enter three numbers: ");
    scanf("%d %d %d", &num1, &num2, &num3);
```

```

if (num1 >= num2) {
    if (num1 >= num3)
        max = num1;
    else
        max = num3;
} else {
    if (num2 >= num3)
        max = num2;
    else
        max = num3;
}

printf("Largest: %d\n", max);

return 0;
}

```

- Program to check if a character is an alphabet, digit, or special character:

```

#include <stdio.h>

int main() {
    char ch;
    printf("Enter a character: ");
    scanf(" %c", &ch);

    if ((ch >= 'a' && ch <= 'z') || (ch >= 'A' && ch <= 'Z'))
        printf("%c is an alphabet.\n", ch);
    else if (ch >= '0' && ch <= '9')
        printf("%c is a digit.\n", ch);
    else
        printf("%c is a special character.\n", ch);

    return 0;
}

```

- Program to determine the quadrant of a given coordinate point:

```

#include <stdio.h>

int main() {

    int x, y;
    printf("Enter the coordinates (x, y): ");
    scanf("%d %d", &x, &y);

    if (x > 0) {
        if (y > 0)
            printf("Quadrant I\n");
        else if (y < 0)
            printf("Quadrant IV\n");
    }
}

```



```

        else
            printf("On the positive x-axis\n");
    } else if (x < 0) {
        if (y > 0)
            printf("Quadrant II\n");
        else if (y < 0)
            printf("Quadrant III\n");
        else
            printf("On the negative x-axis\n");
    } else {
        if (y != 0)
            printf("On the positive y-axis\n");
        else
            printf("Origin\n");
    }

    return 0;
}

```

4. If-Else-If Statements:

- Program to classify a given angle as acute, obtuse, or right:

```

#include <stdio.h>

int main() {
    int angle;
    printf("Enter the angle: ");
    scanf("%d", &angle);

    if (angle > 0 && angle < 90)
        printf("Acute angle\n");
    else if (angle == 90)

        printf("Right angle\n");
    else if (angle > 90 && angle < 180)
        printf("Obtuse angle\n");
    else
        printf("Invalid angle\n");

    return 0;
}

```

- Program to find the roots of a quadratic equation:

```

#include <stdio.h>
#include <math.h>

int main() {
    float a, b, c, discriminant, root1, root2;
    printf("Enter coefficients a, b, and c: ");
}

```

```

scanf("%f %f %f", &a, &b, &c);

discriminant = b * b - 4 * a * c;

if (discriminant > 0) {
    root1 = (-b + sqrt(discriminant)) / (2 * a);
    root2 = (-b - sqrt(discriminant)) / (2 * a);
    printf("Roots are real and different.\n");
    printf("Root1 = %.2f\n", root1);
    printf("Root2 = %.2f\n", root2);
} else if (discriminant == 0) {
    root1 = root2 = -b / (2 * a);
    printf("Roots are real and same.\n");
    printf("Root1 = Root2 = %.2f\n", root1);
} else {
    printf("Roots are complex.\n");
}

return 0;
}

```

- Program to print the name of a month based on its number:

```

#include <stdio.h>

int main() {
    int month;
    printf("Enter the month number (1-12): ");
    scanf("%d", &month);

    if (month == 1)
        printf("January\n");
    else if (month == 2)
        printf("February\n");
    else if (month == 3)
        printf("March\n");
    else if (month == 4)
        printf("April\n");
    else if (month == 5)
        printf("May\n");
    else if (month == 6)
        printf("June\n");
    else if (month == 7)
        printf("July\n");
    else if (month == 8)
        printf("August\n");
    else if (month == 9)
        printf("September\n");
    else if (month == 10)
        printf("October\n");
    else if (month == 11)
        printf("November\n");
    else if (month == 12)
        printf("December\n");
}

```

```

else
    printf("Invalid month number.\n");

return 0;
}

```

5. Switch Case:

- Program to print the name of a day of the week based on its number:

```

#include <stdio.h>

int main() {
    int day;
    printf("Enter the day number (1-7): ");
    scanf("%d", &day);

    switch (day) {
        case 1:
            printf("Monday\n");
            break;
        case 2:
            printf("Tuesday\n");
            break;
        case 3:
            printf("Wednesday\n");
            break;
        case 4:
            printf("Thursday\n");
            break;
        case 5:
            printf("Friday\n");
            break;
        case 6:
            printf("Saturday\n");
            break;
        case 7:
            printf("Sunday\n");
            break;
        default:
            printf("Invalid day number.\n");
    }

    return 0;
}

```

- Program to perform arithmetic operations based on user input:

```

#include <stdio.h>

int main() {

```

```

char operator;
float num1, num2;
printf("Enter an operator (+, -, *, /): ");
scanf(" %c", &operator);
printf("Enter two numbers: ");
scanf("%f %f", &num1, &num2);

switch (operator) {
    case '+':
        printf("Result: %.2f\n", num1 + num2);
        break;
    case '-':
        printf("Result: %.2f\n", num1 - num2);
        break;
    case '*':
        printf("Result: %.2f\n", num1 * num2);
        break;
    case '/':
        if (num2 != 0)
            printf("Result: %.2f\n", num1 / num2);
        else
            printf("Error! Division by zero.\n");
        break;
    default:
        printf("Invalid operator.\n");
}

return 0;
}

```

- Program to convert a given number into words:

```

#include <stdio.h>

int main() {
    int num;
    printf("Enter a number (1-5): ");
    scanf("%d", &num);

    switch (num) {
        case 1:
            printf("One\n");
            break;
        case 2:
            printf("Two\n");
            break;
        case 3:
            printf("Three\n");
            break;
        case 4:
            printf("Four\n");
            break;
        case 5:
            printf("Five\n");

```

```
        break;
    default:
        printf("Number out of range.\n");
    }

    return 0;
}
```

C Programming Journey