Basic Input

There is a function in C which allows the programmer to accept input from a

keyboard.

The Basic Syntax

scanf("format_specifier", &variable);

The following program illustrates the use of this function.

```
#include <stdio.h>

main()

int pin;

{

printf("Please type your PIN\n");

scanf("%d",&pin);

printf("Your access code is %d\n",pin);}
```

In this sequence, we are witnessing the fundamental way a program interacts with a user to store data.

Here is the step-by-step breakdown:

تعديل من خلال WPS Office

Initialization: We start by declaring an integer variable named pin.

At this stage, the computer sets aside a specific "mailbox" in memory for this value, though it doesn't have a meaningful number in it yet.

The User Prompt: The first printf acts as the interface.

It prints a string to the console, inviting the user to provide input.

The scanf Mechanism: This is the core of the operation.

The scanf function requires two specific components to work:

The Control String: Using the %d specifier (note: the prompt mentioned sd, but in standard C, we use %d for integers), the program tells the compiler exactly what kind of data to look for.

تعديل من خلال WPS Office

The Address List: This is where the & (address-of) operator comes in.

Instead of giving scanf the value of the pin, we give it the memory address (&pin).

Basic Integer Input

This is the most common use case: asking for a single whole number.

```
#include <stdio.h>

int main() {

    int age;

  printf("Enter your age: ");

    // %d is the format specifier for integers

    // &age provides the memory address

    scanf("%d", &age);

   printf("You are %d years old.\n", age);


    return 0;

}
```


تعديل من خلال WPS Office

Example 2

```
#include <stdio.h>
int main() {
    float length, width, area;
1 //    . Ask for the first input
    printf("Enter the length: ");
    scanf("%f", &length);
2 //    . Ask for the second input
    printf("Enter the width: ");
    scanf("%f", &width);
3 //    . Perform the calculation
    area = length * width;
4 //    . Display the result
    printf("The total area is: %f\n", area);
    return 0;
}
```


تعديل من خلال WPS Office

Handling Multiple Inputs

You can collect more than one piece of data in a single scanf line.

The order in the control string must match the order in the address list.

```
#include <stdio.h>

int main() {

    int day, month, year;

   printf("Enter your birth date (DD MM YYYY): ");

    // Scanning three integers at once

    scanf("%d %d %d", &day, &month, &year);

 printf("Your birthday is: %d/%d/%d\n", day, month, year);

  return 0;
```


تعديل من خلال WPS Office

}

Working with Different Data Types

scanf isn't just for integers, You can use %f for decimals (floats) and %c for single characters.

```
#include <stdio.h>

int main() {
    float grade;
    char letter;
  printf("Enter your numeric grade: ");
    scanf("%f", &grade);
  printf("Enter your grade letter: ");
    // Note: space before %c helps skip any leftover 'newline' characters
    scanf(" %c", &letter);
   printf("Score: % f | Grade: %c\n", grade, letter);
 return 0;
```


تعديل من خلال WPS Office

}

Important Reminder

Notice that for integers, floats, and characters, we always use the & symbol. This is because these variables are stored at specific memory addresses that scanf needs to access directly.

تعديل من خلال WPS Office

Decision Making Statements

Used to have a program execute different statements depending on certain

conditions.

In a sense,makes a program"smarter"by allowing different choices to be made.

In C,there are three decision making statements.

If execute a statement or not

if-else choose to execute one of two statements

switch choose to execute one of a number of statements

تعديل من خلال WPS Office

if Statement

The if statement allows branching(decision making) depending upon condition.

Program code is executed or skipped.

The basic syntax is

If (control expression)

program statement;

If the control expression is TRUE,the body of the if is executed.

If it is FALSE,the body of the if is skipped.

There is no “then”keyword in C!

Because of the way in which floating point types are stored,it makes it very

difficult to compare such types for equality.

Avoid trying to compare real

variables for equality,or you may encounter unpredictable results.

تعديل من خلال WPS Office

Example

Checking a Passing Grade

This program checks if a student has passed based on a minimum score.

```
#include <stdio.h>

int main() {

    int score;

 printf("Enter your score (0-100): ");

    scanf("%d", &score);

 Simple IF: only runs if score is 50 or //
higher

    if (score >= 50) {

        printf("Congratulations!\n");

        printf("You passed the exam.\n");

    }


    return 0;

}
```


تعديل من خلال WPS Office

Age Verification (Driving Eligibility)

A classic example

A classic example using a comparison operator to check if a user is old enough to drive.

```
#include <stdio.h>

int main() {

    int age;

    printf("Enter your age: ");

    scanf("%d", &age);

 // The condition is checked inside the parentheses

    if (age >= 18) {

        printf("You are eligible to apply for a driver's license.\n");

    }

 return 0;

}
```


تعديل من خلال WPS Office

Detecting Positive Numbers

This example shows how to perform a simple mathematical check.

```
#include <stdio.h>

int main() {

    int number;

  printf("Enter any number: ");

    scanf("%d", &number);

 // If the number is greater than zero

    if (number > 0) {

        printf("The number %d is a positive number.\n", number);

    }


    // Note: If the number is negative, nothing happens here!

    return 0;

}
```


تعديل من خلال WPS Office

Key Concepts to Remember:

Syntax: Always wrap your condition in ( ) parentheses.

Braces { }: While not strictly required for a single line of code, it is "best practice" to use curly braces to keep your code organized and prevent bugs.

تعديل من خلال WPS Office

if-else Statement

Used to decide between two courses of action.

The syntax of the if-else

statement is

If (expression)

statement 1;

else

statement 2;

If the expression is TRUE,statement 1 is executed;statement 2 is

skipped.

If the expression is FALSE,statement 2 is executed;statement 1 is skipped.

تعديل من خلال WPS Office

Even or Odd Checker

This is the most famous example of if-else. We use the modulo operator % to see if a number has a remainder when divided by 2.

```
#include <stdio.h>

int main() {

    int num;

printf("Enter an integer: ");

    scanf("%d", &num);

 If the remainder is 0, it's even

    if (num % 2 == 0) {

        printf("%d is an EVEN number.\n", num);

    }

    // Otherwise, it must be odd

    else {

        printf("%d is an ODD number.\n", num);

    }

return 0;
```


تعديل من خلال WPS Office

```
}
```

Example :

```
#include <stdio.h>
int main() {
    float grade;
printf("Enter your exam grade: ");
    scanf("%f", &grade);
   if (grade >= 60.0) {
        printf("Status: PASSED\n");
        printf("Great job!\n");
    }
    else {
        printf("Status: FAILED\n");
        printf("Please study harder for the next
one.\n");
    }


    return 0;
```


تعديل من خلال WPS Office

```
}
```

Example :

```
#include <stdio.h>
int main() {
    int a, b;
  printf("Enter two numbers (separated by
space): ");
    scanf("%d %d", &a, &b);
if (a > = b) {
        printf("%d is greater than %d\n", a, b);
    }
    else {
        printf("%d is greater than (or equal to)
 %d\n", b, a);
    }


    return 0;
}
```


تعديل من خلال WPS Office

Important Logic Tips:

The "Else" has no condition: You never write else (condition).

The else block automatically catches everything that the if missed.

Mutually Exclusive: Only one of the two blocks will ever run.

It's impossible for both the if and the else code to execute in the same run.

تعديل من خلال WPS Office