Tutorial 1,2,3,4,5

PS Gunathilake

30138

Tutorial 01

Answer

1.

- We use programming language to write programs.
- Programming languages is a tool for writing code, solving problems and creating software.

---

2 .

(a). Source Code vs. Machine Code

Source Code	Machine Code
• close to the user	• closer to the machine
• needs a language translator	• doesn't need a language translator
• uses HLL Languages	• uses machine language

(b). High Level Language vs. Low Level Language

High Level Language	Low Level Language
• uses English language	• use binary code (machine language)
• machine independent	• machine dependent
• portable	• not usually portable

(c). Compiler vs. Interpreter

Compiler	Interpreter
• translates the whole program at once	• Whole program line by line in thee enter order

(d). **Structured Language vs. Object Oriented Language**

- Those two have different principles of programming languages and they have their own way to organizing and manipulating code.

(e**). C vs C++**

- C is a procedural programming language.
- C++ supports both procedural and object- oriented programming paradigms.

(f**). C++ v. Java**

C++	Java
• supports both procedural and object-oriented programming paradigms	• primarily an object-oriented programming language

(g**). Syntax error vs Logical error**

Syntax error	Logical error
• error that occurs in grammar of the language	• Saw up when we use the parenthesis in the incorrect way and when the program runs it gives incorrect results.

***

Tutorial 2

Answer

1. How do you write comments in a c program? What is the purpose of comments in a program?

- We write comments in a c program by using // at the beginning of the comment.
- The Purpose of Comments in a Program Is to Give an Idea to the Programmer of what the Program Says/Does

2. Which is the function that is essential in a C program?

- The Main Function of the Essential in a C program

3. What is the purpose of 'scanf'?

- Purpose Of 'scanf' Is to Get User Inputs

4. Is 'standard c' a case sensitive language?

- Yes,
- 'Standard c' is a case sensitive language.

5.

- Valid :- record1, $tax, name, name_and_address.

- Invalid :- 1record, file-3, return, name and address, name-and-address.
    - **1record** - Because c identifiers can't start with a number
    - **file-3**- Because in c identifiers hyphens or minus signs are not allowed

- **return**- Because return is reserved keyword in c name and address is invalid because c doesn't allow spaces in identifiers
- **name-and-address**- Because in c identifiers hyphens or minus signs are not allowed 123-45-6789 is invalid because in c identifiers hyphens or minus signs are not allowed

---

6 .

a) Function 'printf' always begins printing at the beginning of a new line. – **False**
b) Comments cause the computer to print the text enclosed between /* and */ on the screen when the program is executed. – **False**
c) The escape sequence \n when used in a printf format control string causes the cursor to position to the beginning of the next line on the screen. – **True**
d) All variables must be defined before they're used. – **True**
e) All variables must be given a type when they're defined. – **True**
f) C considers the variables, number and NuMbEr to be identical. – **False**
g) A program that prints three lines of output must contain three printf statements. - **False**

7.

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

```
*
**
***
****
*****
```

8

a) **scanf("d", value);**
   - there is no % before d and & before value
   - scanf("%d", value);

b) **printf("The product of %d and %d is %d" \n, x, y);**
   - There are 3 %d symbols but only 2 variables and the \n is written outside the "symbol
   - printf("The product of %d and %d \n", x, y);

c) **Scanf("%d", aninteger);**
   - there is no & before aninteger
   - Scanf("%d",& aninteger);

d) **printf("Remainder of %d divided by %d is\n", x, y, x % y);**
   - There are 3 variables but there are only 2 %d symbols
   - printf("Remainder of %d divided by %d is%d \n", x, y, x % y);

e) **printf("The sum is %d\n," x + y);**
   - the comma is inside the ""
   - printf("The sum is %d\n" ,x + y);

f) **Printf("The value you entered is: %d\n, &value);**
   - there is no ending symbol of " "symbol and there is an extra & Infront of value
   - Printf("The value you entered is: %d\n", &value);

9.

a). printf("%d",x);	2
b). printf("%d",x+x);	4
c). printf("x=");	x=
d). printf("x=%d",x);	x=2
e). printf("%d = %d",x+y, y+x);	5=5
f). z=x+y;	Nothing
g). scanf("%d %d", &x, &y);	Nothing
h). /* printf("x + y = %d",x + y);	*/x+y=5
i). printf("\n");	print the line break

10.

a) **C operators are evaluated from left to right.**
   - **False.**
   - It Depends on The Precedence of The Operators

b) **The following are all valid variable names: under_bar_m928134, 15, 17, her _sales, his _account _total, a, b, c, z, z2.**
   - **True**

c) **The statement printf("a = 5;"); is a typical example of an assignment statement.**
   - **False.**
   - Because 5 is only the typical example of assignment statement not printf("a=5");

d) **A valid arithmetic expression containing no parentheses is evaluated from left to right.**
   - **False.**
   - It Depends on The Precedence of The Operators

e) **he following are all invalid variable names: 3g, 87, 67h2, h22, 2h**
   - **False.**
   - Because h22 is a valid variable name

***

Tutorial 03

Answer

1.

- **Statement 1** int x=0; x=x+1;
- **Statement 2** int x=0; x= -1;
- **Statement 3** int x=0; x++;
- **Statement 4** int x:x=0+1;

2.

a) Assign the sum of x and y to z and increment the value of x by 1 after the calculation.

- z=x+y; x++;

b) Multiply the variable product by 2 using the operator.

- product*=2;

c) Multiply the variable product by 2 using the and operators.

- Product=product* 2

d) Test if the value of the variable count is greater than 10. If it is, print "Count is greater than 10."

- if(count>10) printf("Count is greater than 10.");

e) Decrement the variable x by 1, then subtract it from the variable total

- x++; y=total-x;

f) Add the variable x to the variable total, then decrement x by 1.

- Y=x total; X--;

g) Calculate the remainder after q is divided by divisor and assign the result to q. Write this statement two different ways.

Divisor - d

- qq%d;
- p-q%d;q=p;

h) Print the value 123.4567 with 2 digits of precision. What value is printed?

- 123.45

i) Print the floating-point value 3.14159 with three digits to the right of the decimal. What value is printed?

- 3.141

3.

a) Input integer variable x with scanf

- Scanf("%d",&x);

b) Input integer variable y with scanf.

- Scanf("%d",&y);

c) Initialize integer variable i to 1

- int i=1;

d) Initialize integer variable power to 1.

- Int power= 1;

e) Multiply variable power by x and assign the result to power.

- power=power*x;

f) Increment variable i by 1.

- i=i+1;

g) Test i to see if it's less than or equal to y in the condition of a while statement.

- while(i<=y)

h) Output integer variable power with printf.

- printf("%d",power);

***

Tutorial 04

Answer

1) .

Errors –

- There is no () used to indicate the condition of the if statement,
- The statements inside if and else is not indented,
- The condition 2 is already inside the condition 1

---

2)

printf("No, I'm here! \n");

printf("No, actually, I'm here! \n");

---

3)

- If ‘ doesSignificantWork ’ is true and ‘ makeBreakthrough ’ is true, ‘ nobelPrizeCandidate ’ is also true.
- If ‘ doesSignificantWork ’ is true and ‘ makeBreakthrough ‘ is false, ‘nobelPrizeCandidate ‘ is become false.
- If ‘ doesSignificantWork ‘ is false, ‘ nobelPrizeCandidate ‘ is always become false.

---

4)

- If character variable taxCode is 'T', increase price by adding the taxRate percentage of price to it.

- If(taxcode='T')
  {price=price+taxRate;}

-If integer variable opCode has the value 1, read in double values for X and Y and calculate and print their sum.

- If(opCode=1){
  printf("The Sum is %f",%x+y);}

- If integer variable currentNumber is odd, change its value so that it is now 3 times currentNumber plus 1, otherwise change its value so that it is now half of currentNumber (rounded down when currentNumber is odd).

- if(currentNumber%2=1)(currentNumber 3 currwntNumber + 1;)
  else(currentNumber-currentNumber/2;)

-Assign true to the boolean variable leap Year if the integer variable year is a leap year. (A leap year is a multiple of 4, and if it is a multiple of 100, it must also be a multiple of 400.)

- if(year%4=0)
  {leapYear=true;}

-Assign a value to double variable cost depending on the value of integer variable distance as follows:

- if(distance<=100) {cost=5.00;}
  else if(distance<=500) {cost=8.00;}
  else if(distance<1000) {cost=10.00;}
  else{cost=12.00;}

Tutorial 5

switch

```
#include<stdio.h>
int main () {
   int choice;
   float no1, no2;

   printf("enter two numbers");
   scanf("%f %f, &no1 , &no2");

   printf("1. +\n");
   printf("2. -\n");
   printf("3, *\n");
   printf("4. /\n");
   printf("please enter your choice");
   scanf("%f, &choice");
```

```
switch (choice) {
      case 1:
        printf("result %.2f\n", no1 + no2);
        break;
      case 2:
        printf("result %.2f\n", no1 - no2);
        break;
```

```
    case 3:
       printf("result %.2f\n ", no1 * no2);
       break;
    case 4:
     if (no2 != 0) {
       printf("result %.2f\n", no1 / no2);
     } else {
       printf("error: division by zero is not allowed here\n");
     }
     break;
    default:
      printf("invalid choice\n");
      break;
  }
return 0;}
```

While loop

1.

```
#include<stdio.h>
int main () {
      int i = 1, evenc = 0, oddc = 0, no;

      printf("Enter 10 numbers:\n");

       while ( i <= 10) {
              printf("Enter number %d: ", i);
       scanf("%d", &no);

 if (no % 2 == 0) {
     evenc++;
  } else {
               oddc++;
              }

               i++;
                }
                printf("Total count of even numbers: %d\n", evenc);
                printf("Total count of odd numbers: %d\n", oddc);
return 0;
        }
```

2.

```
#include<stdio.h>
int main () {
     int no , evenc = 0 , oddc = 0;
     printf("Enter numbers (terminate with -99):\n");;

    while(1) {
        printf("Enter a number: ");
       scanf("%d", &no);
     if (no == -99) {
           break; // terminate the loop when -99 is entered
          }
    if  (no % 2 == 0) {
         evenc++;
      } else {
        oddc++;
      }
    }
         printf("Total count of even numbers: %d\n", evenc);
         printf("Total count of odd numbers: %d\n", oddc);


  return 0;

}
```

Do while loop

1.
```
#include<stdio.h>
int main () {
   int i = 1,evenc = 0, oddc = 0, no;

   printf("Enter 10 numbers:\n");

   do {
      printf("enter number %d: ", i);
      scanf("%d", &no);

      if (no % 2 == 0) {
         evenc++;
      }else {
         oddc++;
      }
      i++;
   } while (i <=10 );
   printf("total count of even numbers: %d\n", evenc);
   printf("total count of odd numbers: %d\n", oddc );

   return 0;
}
```

2.

```
#include<stdio.h>
int main () {
   int no, evenc = 0,oddc = 0;

   printf("enter numbers (terminate with -99):\n");

   do {
      printf("enter a number:");
      scanf("%d", &no);

      if (no== -99){
          break;//terminate the loop when -99 is entered
      }
       if (no % 2 ==-99) {
         evenc++;
       } else {
          oddc++;
       }
      } while (1);

       printf("total count of even numbers: %d\n", evenc);
       printf("total count of odd numbers: %d\n", oddc);
 return 0;}
```

For loop

1.

```
#include<stdio.h>
        int main () {
           int total = 0 , count = 0 ,num;
           for( int i = 0; i < 10; i++ ) {
                    printf("Enter number: ");
                         scanf("%d", &num);


                   total += num;
                   count++;
             }
          float average = (float) total / count;


          printf("The average is: %.2f\n", average);
      return 0;
     }
```

2.

```
#include<stdio.h>
int main () {
    for (int i = 1; i <= 5; i++) {
 for (int j = 1; j <= i; j++) {
    printf(“*”);
 }
   Printf(“\n”);
}
return 0;
}
```

***