Revision. * Refer operators lesson

26/02/2019

C - Sample Paper (18.2)

p = 10 ; q = 3

(A) a.) r = p++ + q++
= 10 + 3
= 13

p++ (post increment) doesn't change

++p (pre increment changes

b) r = p-- % q
= 10 % 3
= 1

% modulus

/ divide (quotient)

c) r = --p / --q
= 9 / 2
= 4 (it cannot be 4.5 because datatype is int

d) r = ++p * q++
= 11 * 3
= 33

(B)
```
#include <stdio.h>
int main() {
    int x, y, c, power = 1;
    printf("Enter value for x and y ");
    scanf("%d %d", &x, &y);

    for (c=1; c<=y; c++)
    {
        power = power * x;
    }
    printf("Answer is %d \n", power);
```

y = exponent
x = base

* Refer loops.

(C)

a.) What is the general format of for loop, on which occasions you use a 'for loop'?

→
```
for ( expression 1 ; expression2 ; expression3 )
{

        // body

}
```

* For is used to represent an iteration or repetition within a program.

b.) Write a single printf statetment to display the values of an integer X and float variable Y.

→
```
printf ( " %d  %f " , x , y );
```

OR

```
printf ( " x = %d  y = %f " , x , y );
```

c.) Declare and assign values for four different variables with four different data types.

→
```
int a = 10 ;
char b = 'c' ;
float c = 12.5 ;
double d = 58752.62
```

* use single qoute for singe character 'c'

d) What are the assignment operators and when we use them?

→ + =

- =

* =

/ =

Assignment operators are used to write an arithmetic expression is a short form

Ex : x = x + 10 as x + = 10

(1 mark) (1 mark)

e.) Explain the general formula and the use of a switch conditional structure.

*Study a program using Switch.

```
→ switch (<variable>)
  {
      case <option 1> : //statement ; break ;
      case <option 2> : // statement ; break ;
      case <option 3> : // statement ; break ;
      default : statement ;
  }
```

(2)

1.) Use adding 2 numbers as an example and write four different functions to explain the behavior and use four different function types.

→ No return type, no parameters

```
# include  <stdio.h>
void  sum()
{
        int x, y, sum;
        printf ("Enter two numbers : ");
        scanf ("%d  %d", &x, &y);
        Sum = x + y;
        printf (" The sum is %d", sum);
}
int main ()
{
          sum();
}
```

→ No return type, with parameters.

```
# include <stdio.h>
void  sum ( int a, int b);
{
        int z;
        z = a + b;
        printf ("The Sum is %d", z);
}
int main()
{
          int x, y;
          printf ("Enter two numbers : ");
          scanf ("%d  %d", &x , &y);
          sum('x','y');
}
```

* value of x goes to a

* value of y goes to b

return z;

→ With return type , no parameters

```
# include <stdio.h>
int sum()
{
        int x , y , z ;
        printf (" Enter two numbers " );
        scanf ( " %d %d ", &x , &y );
        z = x+y ;
return z ;
}

int main()
{
        int c = sum();
        printf (" The sum is %d ", c );
}
```

→ With return type , with parameters .

```
#include <stdio.h >
int sum ( int x , int y )
{
        int z ;
        z = x +y ;
return z;
}
int main ( )
{
        int a , b , c ;
        printf (" Enter two numbers " );
        scanf (" %d %d ", &a , &b ) ;
        c = sum (a , b ) ;
        printf ( " The sum is %d \n ", c );
}
```

B) Create a function to provide three integers as parameters to a function and find and return the highest number. Input 3 numbers in the main function, call the function and display the highest number.

```
→ int   findmax ( int a, int b, int c)
  {
        int  max ;
        if (a > b)
        max = a
        else if ( b > c)
        max = b..
        else    max = c
  return  max ;
  }
int main ()
{
        int x, y , z ;
        printf (" Enter three numbers: ");
        scanf (" %d  %d  %d ",&x &y &z);
        int  ans = findmax (x,y,z);
        printf (" The highest is %d \n", ans);
```

(3.)

(A) I.) Input 10 float values and store them in an array

```
→ int main() {
  float arr [10];
  int i;
  for (i = 0; i < 10; i++)
  {
        printf ( "Enter a value in the element %d", i+1);
        Scanf ("%f", arr [i] );
   }
 }
```

II.) Display the values of the above array.

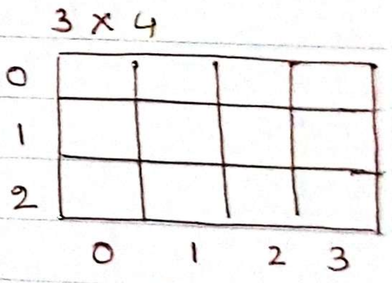
```
 int main ()
 {
   float arr [10];
   int i;
   for (i = 0; i<10; i++)
   {
        printf (" Enter a value in the element %d ", i+1 );
        Scanf ( "%f ", arr[i]);
    }
→ for ( i=0; i<10; i++)
  {
      printf ("%.2f", arr [i]);
  }
 }
```

III.) Find and display the minimum value.

```
→    float = arr [0] ;
     for (i = 1 ; i < 10 ; i++)
     {
          if (arr [i] < min )
          min = arr [i] ;
     }
     printf (" The lowest  is  %.2f ", min ) ;
```

(B) ~~[illegible]~~ Declare a multi - dimensional array with the size of 3 x 4. Input values to the array and display the average value

```
for (r = 0 ; r < 3 ; r ++ )
{
        for (c = 0 ; c < 4 ; c++ )
        {
              printf ("Enter a value ") ;
              scanf (" %.d , & v[r][c]) ;
              Sum = sum + v [r] [c] ;
        }
}
avg = (float) sum / 12 ;
Printf (" The Average is  %.2f \n ", avg ) ;
```



* row by row.