PROGRAMMING IN JAVA

Assignment -- 0

TYPE OF QUESTION: MCQ

Number of questions: 15 **Total mark: 15×1 = 15**

QUESTION 1:

A one dimensional array A has indices 0....54.Each element is a string and takes up four memory words. The array is stored starting at location 1000 in decimal form. The starting address of 41th element of A is

a. 1000
b. 1160
c. 1040
d. 1055

Correct Answer: b

Detailed Solution:

One element takes four memory words so memory location 1000 , 1001 , 1002,1003 stores first element. 41 th element will be stored at location 1160 , (1000+(40*4)).

QUESTION 2:

What is the output of the following program in C?

```
#include <stdio.h>
int main()
{
     enum Nptel { July = 0, Aug, Dec};
     enum Nptel course = Dec;
     if( course == 0)
            printf("Course is in July");
     else if(course == 1)
         printf("Course is in Aug");
     if(course == 2)
       printf("Course is in Dec");
}
```

a. Course is in July
b. Course is in Aug
c. Course is in Dec
d. Compilation error

Correct Answer: c

Detailed Solution:
Enumeration data type consists of named integer constants as a list. It start with 0 (zero) by default and value is incremented by 1 for the sequential identifiers in the list.

QUESTION 3:

Consider the statement

int X[2] [4] = { 1, 2, 3, 4, 5, 6, 7, 8} ;

5 will be the value of

a. X[0][1]
b. X[0][5]
c. X[1][5]
d. X[1][0]

Correct Answer: d

Detailed Solution:
Tha data are placed to the 2D array in row-major order, that is, int X[2][4] = {1,2,3,4,5,6,7,8}; will be seen as

X[2][4] = {

{1,2,3,4},

{5,6,7,8}

};

Thus, 5 is in X[1][0].

QUESTION 4:

What does the following declaration mean?

```
int (*ptr)[15];
```

a. ptr is array of pointers to 15 integers.
b. ptr is a pointer to an array of 15 integers.
c. ptr is an array of 15 integers.
d. ptr is a pointer to a two-dimensional array.

Correct Answer: b

Detailed Solution:
This is the declaration of pointer to an array of size 15.

QUESTION 5:

In C, if you pass the name of an array as an argument to a function, what actually gets passed?

a. Value of elements in array.
b. First element of the array.
c. Base address of the array.
d. Address of the last element of the array.

Correct Answer: c

Detailed Solution:
When we pass an array as a function argument, the base address of the array will be passed.

QUESTION 6:

Which of the following statements(s) is/are not true?

a. Assembler translates a program in high level programming language into a code in an assembly language.
b. Compiler translates a program in high level programming language into a code in a machine language.
c. Interpreter translates a program in high level programming language into a code in a machine language.
d. A program in C is compiled whereas a program in C++ is interpreted.

Correct Answer: a, d

Detailed Solution:
All high level language (in general) is compiled to code in machine level language. The same strategy is followed in both C and C++ programming languages.

QUESTION 7:

Consider the function

```
myfunc( int x, int y)
  {
 return (( x < y ) ? 0 : ( x - y ));
  }
```

Let a, b be two non-negative integers.
The call myfunc(a, myfunc(a, b)) can find the

a. Maximum of a, b.
b. Positive difference of a, b.
c. Sum of a, b.
d. Minimum of a, b.

Correct Answer: d

Detailed Solution:

This function `myfunc(a,myfunc(a, b))` is to find the minimum of a,b.

QUESTION 8:

What will be the value retuned by the following program, when it is called with a value 10?

```
#include<stdio.h>
int recur(int);
void main()
{
    int d;
    d=recur(10);
```

```
    printf("%d",d);
}
int recur(int num)
{
    if((num/2)!=0)
      return(recur(num/2 ) * 10+num%2 );
     else
       return 1;

}
```

a. 1010
b. 1011
c. 10
d. 0110

Select the correct option(s) in the above.

Correct Answer: a

Detailed Solution:

QUESTION 9:

Which of the following is not based on object-oriented paradigms?

a. C
b. C++
c. Java
d. Python

Correct Answer: a

Detailed Solution:
C is not based on 'object-oriented' paradigm rather than function-oriented paradigm. It has no concept of classes, objects, polymorphism, and inheritance.

QUESTION 10:

What is the output of the following code segment?

```
int  i = 1;
printf("%d + %d",i++,++i);
```

a. 1 + 2
b. 2 + 3
c. 2 + 2
d. 1 + 1

Correct Answer: b

Detailed Solution:
It is the concept of pre-increment and post-increment. The evaluation is done from right-to-left and printf will print the value from left-to-right.

QUESTION 11:

Which of the following operators takes only integer operands in C programming language?

a. + (plus)
b. *(multiplication).
c. /(division).
d. %(modulo)

Correct Answer: d

Detailed Solution:

modulus operator ('**%**'), computes the remainder by performing integer division.

QUESTION 12:

Which of the following statement(s) to declare an array, say 'a' is(are) **not** valid in C programming language?

```
a. int a[m];     //m is an integer variable
b. int *a;
c. int a[ ]={1,2,3,4,5};
d. int a[10000];
```

Correct Answer: a, b

Detailed Solution:
C language needs the size of the array at the time of compilation. Since the value of m is not known at the time of compilation, C compiler will give compilation error. Further, note that *a is a pointer to an integer variable, it can point to an array but not actually an array.

QUESTION 13:

Which of the following is not a valid keyword in C?

a. auto
b. void
c. switch
d. malloc

Correct Answer: d

Detailed Solution:

1. Keywords are those words whose meaning is already defined by Compiler
2. Cannot be used as **Variable Name**
3. There are **32 Keywords** in C
4. C Keywords are also called as **Reserved words**

The name "malloc" stands for memory allocation. It is not a keyword, rather is a library fiction defined in <stdlib.h> in C programming language. The malloc() function reserves a block of memory of the specified number of bytes. And, it returns a pointer of type void which can be casted into pointer of any form.

QUESTION 14:

What result the following macro will produce in a statement in C programming language?

```
#include <stdio.h>
#define square(x)  x*x
int main() {
        int  x=36/square(6);
        printf("%d",x);
        return 0;
}
```

a. 36
b. 1

c. 0
d. 6

Correct Answer: a

Detailed Solution:
The calculation is 36/6*6. This will be equivalently (36/6)*6 because both / and * are of same precedence and with left-to-right associativity.

QUESTION 15:

Consider the following piece of code in the C programming language.

```
#include<stdio.h>
main()
{
  inc(); inc(); inc();
}
inc()
{
      static int x;
      printf("%d", ++x);
}
```

If you run this program, then which of the following statement(s) is (are) true?

a. The program will print the value 123
b. The program will print the value 111.
c. The program will print the value 012.
d. The program will print the value 000.

Correct Answer: a

Detailed Solution:

By default x will be initialized to 0. Since its storage class is static, it presents its exit value (and forbids reinitialization on re-entry). So, 123 will be printed.
