Assignment 8 Solution

1. A function prototype is used for
   a) Declaring the function logic
   b) Calling the function from the main body
   c) Telling the compiler, the kind of arguments used in the function
   d) Telling the user for proper use of syntax while calling the function

Solution: (c) A function prototype tells the compiler what kind of arguments a function is looking to receive and what kind of return value a function is going to give back. This approach helps the compiler ensure that calls to a function are made correctly and that no erroneous type conversions are taking place.

2. What is the default return type if it is not specified in function definition?
   a) void
   b) integer
   c) double
   d) float

Solution: (b) Integer is the default data type if not specified in the function.

3. What will be the output?

```
#include <stdio.h>
int main()
{
      {
         int a = 70;
      }
{
         printf("%d", a);
    }
  return 0;
}
```

   a) 70
   b) Garbage value
   c) Compilation error
   d) None

Solution: (c) Compilation error.

A Block is a set of statements enclosed within left and right braces ({and} respectively). A variable declared in a block is accessible in the block and all inner blocks of that block but not accessible outside the block. Thus the program produces a compiler error.

4. How many times will ‘Hello world’ be printed?

```
#include<stdio.h>
int main()
{
   printf("Hello world\n");
   main();
   return 0;
}
```

a) Infinite times
b) 32767
c) 65535
d) Till stack overflows

Solution: (d) Till stack overflows

5. How many times ‘Hi’ will be printed in the program given below

```
#include<stdio.h>
int i;
int fun();

int main()
{
   while(i)
   {
      fun();
      main();
   }
   printf("Hello\n");
   return 0;
}
int fun()
{
   printf("Hi");
}
```

a) Only once
b) Zero times
c) Infinite times
d) Compilation error

Solution: (b) The default value of i is ‘0’. Thus, the while loop will never be executed and the control will not come into the function. Thus, ‘Hi’ will never be printed.

6. How many times the function factorial will be executed?

```
#include<stdio.h>
int factorial(int);
int main()
{
int n=5;
long int f;
f = factorial(n);
printf("%d! = %ld\n", n, f);
return 0;
}
int factorial(int n)
{
if (n == 0)
return 1;
else
return(n * factorial(n-1));
}
```

Short Answer Type: Answer: 6

As n =5, so the function factorial() calls itself 5 times. In addition, factorial() function is called once from the main(). So, total 6 times the function factorial() will be executed.

7. What will be the output?

```
#include<stdio.h>
void func(int n, int sum)
{
int k = 0, j = 0;
if (n == 0) return;
        k = n % 10;
j = n / 10;
sum = sum + k;
func (j, sum);
printf ("%d,", k);
}

int main ()
{
int a = 2048, sum = 0;
func (a, sum);
printf ("%d ", sum);
}
```

a) 8 ,4, 0, 2, 14
b) 8, 4, 0, 2, 0
c) 2, 0, 4, 8, 14
d) 2, 0, 4, 8, 0

Solution: (d) 2, 0, 4, 8, 0
sum has no use in func(), it is there just to confuse. Function func() just prints all digits of a number. In main, there is one more printf statement after func(), so one more 0 is printed after all digits of n.

8. What is the output of the following C program?

```
#include <stdio.h>
int fun(int n)
{
   int i, j, sum = 0;
   for(i = 1; i<=n; i++)
      for(j=i; j<=i; j++)
         sum = sum + j;
   return(sum);
}
int main()
{
   printf("%d", fun(10));
   return 0;
}
```

a) 55
b) 45
c) 66
d) 10

Solution: (a) The program finds the sum of the integer numbers till 10. Thus the output is 55.

9. Consider the function

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

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

a) Maximum of a, b
b) Positive difference between a and b
c) Sum of a and b
d) Minimum of a and b

Solution: (d) Minimum of a and b

The function returns

0 if x<y

x-y if x>y

Assume in the first case a>b so find(a,b) will return (a-b). So find(a,find(a,b)) will be find(a,a-b) which will return (a-(a-b))= b which is the minimum number.

Now if a<b then find(a,b)=0 so find(a,find(a,b))=find(a,0) which will return (a-0)=a which is the minimum number. So, the code actually returns a minimum of two numbers.

10. What is the output of the C code given below

```
#include <stdio.h>
float func(float age[ ]);

int main()
{
   float result, age[] = { 23.4, 55, 22.6, 3, 40.5, 18 };
   result = func(age);
   printf("%0.2f", result);
return 0;
}

float func(float age[ ])
{
   int i;
   float result, sum = 0.0;
   for (i = 0; i < 6; ++i) {
      sum += age[i];
   }
   result = (sum / 6);
   return result;
}
```

Ans: 27.08