MIDDLE EAST TECHNICAL UNIVERSITY, NORTHERN CYPRUS CAMPUS
CNG140 C Programming – Worksheet 10

Purpose:
In this work sheet, you familiarize yourself with functions, call by values parameters and return values.

Tasks:

a. Write a function `digit(n,k)` that returns the k^{th} digit (from the right) in n (positive integer). For example, `digit(829,1)` returns 9, `digit (829,2)` returns 2 and `digit (829,3)` returns 8. If k is greater than the number of digits in n, then the function returns 0.

Sample Run 1:

```
Enter n and k: 829 1
1st digit from the right is 9
```

Sample Run 2:

```
Enter n and k: 829 4
The number of digits in 829 is less than 4!
```

b. A palindromic *number* is a 'symmetrical' number - such as 16461 - that remains the same when its digits are reversed. Write a C function

```
int isPalindrome(int number)
{
  ...
}
```

that tests whether a given integer n is a palindromic number and use that function to write a program that lists all palindromic numbers with d digits. **Hint:** Consider the number 16461: strip the most and least significant digits from 16461, compare whether they are equal and, if they are, compute the new number 646 from 16461. Repeat the process until either two matching digits are unequal or there is nothing left to do.

Sample Run:

```
Enter number of digits: 2
Palindromic numbers with 2 digits are 11, 22, 33, 44, 55, 66, 77, 88, 99
```

c. Write a function called `isMultipleofGivenNumber` that takes two integers and returns true if the first given number is the multiple of the second given number, and returns false if it is not the case. For example, `isMultipleofGivenNumber(24,2)` returns 1 (true), and `isMultipleofGivenNumber(24,5)` returns 0 (false).

Sample Run:

```
Enter two integers: 6 3
6 is multiple of 3
```

d. Write a function called `IsSumDigitsEven` that takes an integer value and returns true if the sum of the digits is an even number and returns false if the sum of the digits is an odd number. For example, the function call `IsSumDigitsEven(3553)` should return true as the sum of digits of the given number is 16 which is an even number.

Sample Run:

```
Enter a number: 235
Sum of digits of number 235 is even.
```

e. Write a program to model a simple calculator. Each data line should consist of the next operation to be performed from the list below and the right operand. Assume that the left operand is the accumulator value (initial value of 0).

```
+ add
- subtract
* multiply
^ power
q quit

+ 5
result so far is 5
^ 2
result so far is 25
q 0
final result is 25
```

In this program, you need to write the following functions:

```
void menu (void) //This will display the menu
int add (int, int) //Takes two integers and returns their total
int sub (int, int) //Takes two integers and returns their total
int power (int, int) //Takes the power of a given number
int multiply (int, int) //Multiples two integers
```

f. At a pizza restaurant, a small pizza costs 3 euro with the first topping, and then an additional 85 cents for each additional topping and for a large pizza 5 euro with the first topping, and then an additional 95 cents for each additional topping. Write a program that takes the size of the pizza, the number of toppings, and also the amount of money paid, and calculates the cost of the pizza and also the change that needs to be returned. Your program needs to have two functions:

- `float cost_calculator(character, integer)`: This function takes the pizza type and also the number of toppings and returns the cost of the pizza.
- `float change_calculator(float, float)`: This function takes the total cost of a pizza, the amount of money paid, and returns the change.

Sample Run 1:

```
Enter the size of the pizza (s/l): s
Enter the number of toppings: 4
Enter the amount paid (in Euro): 6
Your Pizza costs 5.55 Euro and you will get 0.45 change back!
```

Sample Run 2:

```
Enter the size of the pizza (s/l): k
Not a valid Pizza size!
```

Sample Run 3:

```
Enter the size of the pizza (s/l): s
Enter the number of toppings: -2
Not a valid topping number!
```

Sample Run 4:

```
Enter the size of the pizza (s/l): s
Enter the number of toppings: 4
Enter the amount paid (in Euro): 3
Your Pizza costs 5.55 Euro and you need to pay 2.55 Euro more!
```

Recommended Reading: Chapter 6 (p. 271-326)
Recommended Exercises: All practical exercises in Chapter 6.