UNIVERSITEIT•STELLENBOSCH•UNIVERSITY
jou kennisvennoot • your knowledge partner

Computer Programming 143

Practical 5- Memo

2016

Assignment 5A

```c
/* Filename: Assignment5A.c
* Date: 2016/01/01
* Name: Doe J.J.
* Student number: 12345678
* ------------------------------------------------------
* By submitting this file electronically, I declare that
* it is my own original work, and that I have not copied
* any part of it from another source.
* ------------------------------------------------------
* This program defines an array filled with random
* values and performs operations on this array.
* ------------------------------------------------------
*/
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

#define SIZE 10

int main (void)
{
        // define an array of characters of length SIZE
        int array [SIZE];
        int i;

        // change seed of random number generator
        srand(time(NULL));

        // populating values within array
        for(i=0;i<SIZE;i++)
        {
                // generate random value between 1 and 20
                // store this value within array
                array[i] = (rand()%20) + 1;
        }

        // displaying values within array
        for(i=0;i<SIZE;i++)
        {
                // print each value within array to screen
                printf("%d ", array[i]);
        }
        printf("\n");

        // maths on arrays
```

```
46          for(i=0;i<SIZE;i++)
47          {
48                  // multiply each element with 10
49                  array[i] = 100*array[i];
50          }
51
52          // displaying values within array in reverse
53          for(i=SIZE-1;i>=0;i--)
54          {
55                  // print each value within array to screen
56                  printf("%d ", array[i]);
57          }
58          printf("\n");
59
60          return 0;
61  }
```

Assignment 5B

```c
/* Filename: Assignment5A.c
* Date: 2016/01/01
* Name: Doe J.J.
* Student number: 12345678
* ------------------------------------------------------
* By submitting this file electronically, I declare that
* it is my own original work, and that I have not copied
* any part of it from another source.
* ------------------------------------------------------
* This program defines an array filled with characters
* and sorts the characters alphabetically.
* ------------------------------------------------------
*/
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

// define the size of array to be created
#define SIZE 100

// function prototypes
void assignArray(char array [], int sizeOfArray);
void displayArray(char array [], int sizeOfArray);
void sortArray(char array [], int sizeOfArray);

int main(void)
{
        // define an array of characters of length SIZE
        char array [SIZE];

        // change seed of random number generator
        srand(time(NULL));

        // function to assign random values to array
        assignArray(array, SIZE);

        // function to display content of array
        displayArray(array, SIZE);

        // function to sort the content of array in ascending order
        sortArray(array, SIZE);

        // function to display content of array
        displayArray(array, SIZE);

```

```
46          return 0;
47  }
48
49  // function definition
50  void assignArray(char array [], int sizeOfArray)
51  {
52          int i;
53          for(i=0;i<sizeOfArray;i++)
54          {
55                  // obtain a random number between 97 and 122
56                  // cast number to a character type
57                  array[i] = (char)((rand()%26)+97);
58          }
59  }
60
61  void displayArray(char array [], int sizeOfArray)
62  {
63          int i;
64          for(i=0;i<sizeOfArray;i++)
65          {
66                  // prints the character within array
67                  printf("%c ", array[i]);
68          }
69          printf("\n");
70  }
71
72  void sortArray(char array [], int sizeOfArray)
73  {
74          int i, j;
75          char temp;
76
77          // standard bubble sorting algorithm
78          for(i=0;i<sizeOfArray;i++)
79          {
80                  for(j=0;j<(sizeOfArray-1);j++)
81                  {
82                          if(array[j]>array[j+1])
83                          {
84                                  temp = array[j];
85                                  array[j] = array[j+1];
86                                  array[j+1] = temp;
87                          }
88                  }
89          }
90  }
```

Assignment 5C

```c
/* Filename: Assignment5A.c
* Date: 2016/01/01
* Name: Doe J.J.
* Student number: 12345678
* ------------------------------------------------------
* By submitting this file electronically, I declare that
* it is my own original work, and that I have not copied
* any part of it from another source.
* ------------------------------------------------------
* This program defines recursive functions which either
* prints increments a counter to a desired value or
* determine the greatest common divisor of two numbers
* ------------------------------------------------------
*/
#include <stdio.h>
#include <stdlib.h>

// function prototypes
void printCounter(int value);
int gcd(int x, int y);

int main (void)
{
        // function call to recursively print a counter
        printCounter(10);

        // multiple calls to recursive function which determines
        // greatest common divisor
        printf("\ngcd(%d, %d) = %d", 10, 20, gcd(10,20));
        printf("\ngcd(%d, %d) = %d", 20, 10, gcd(20,10));
        printf("\ngcd(%d, %d) = %d", 777, 99, gcd(777,99));
        printf("\ngcd(%d, %d) = %d", 345, 6000, gcd(345,6000));

        return 0;
}

// function definition
void printCounter (int value)
{
        // define a static variables
        static int x = 0;

        printf("\n%d", x);      // print value

        x++;    // increment counter
```

```
46
47          if(x<value)
48          {
49                  // if counter has not reached  desired value
50                  // call itself again
51                  printCounter(value);
52          }
53  }
54
55  // function definition
56  int gcd(int x, int y)
57  {
58          // determines whether y value is zero
59          // if true return 0, otherwise
60          // call itself again
61          if(y == 0)
62          {
63                  return x;
64          }
65          else
66          {
67                  return gcd(y, x%y);
68          }
69  }
```