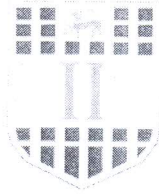


IT No.....

PC No.



Sri Lanka Institute of Information Technology

B.Sc. Degree
in
Information Technology

Final Examination
Year 1, Semester 1 (2017)
June Intake

Introduction to Programming (C / C++) (N102)

Duration: 2 Hours



Instructions to Candidates:

- ◆ This is an **online closed book** examination.
- ◆ This paper contains **1 question on 2 pages** without the cover page.
- ◆ Create a folder in the **Home** directory with your **IT number**.
- ◆ Save the program you write inside the folder.
- ◆ Read the question before answering.
- ◆ The total marks obtainable for this examination is 50.

Question One

(50 marks)

The "XYZ GOODS" Pvt. Ltd. manufactures spices for the export market. A sample production of machine A for Week 1 and Week 2 are shown in the following table 1 and 2 respectively. The production ranges between 50 and 300 per day.

Table 1-Machine A Production Week 1

Day	1 (MON)	2 (TUE)	3 (WED)	4 (THU)	5 (FRI)	6 (SAT)	7 (SUN)
Array index	0	1	2	3	4	5	6
Production	50	60	150	100	190	240	155

Table 1-Machine A Production Week 2

Day	1 (MON)	2 (TUE)	3 (WED)	4 (THU)	5 (FRI)	6 (SAT)	7 (SUN)
Array index	0	1	2	3	4	5	6
Production	80	65	115	170	145	95	103

- (a) Write a function using C statements called *InputProduction()* which takes an integer *array* and the *size of the array* as parameters. The method should ask the user to insert production of each day (Monday to Sunday) and fill the array. The values are entered through the keyboard. (5 marks)
- (b) Write a function using C statements called *MinimumDay()* which takes an integer *array* and the *size of the array* as parameters. The method should find and return the day of the lowest production (1 to 7) in the array. (5 marks)
- (c) Write a function using C statements called *MaximumDay()* which takes an integer *array* and the *size of the array* as parameters. The method should find and return the day of the highest production (1 to 7) in the array. (5 marks)
- (d) Write a function using C statements called *TotalProduction()* which takes an integer *array* and the *size of the array* as parameters. The method should find and return the total production of all the elements in the array. (5 marks)
- (e) Write a function using C statements called *AverageProduction()* which takes an integer *array* and the *size of the array* as parameters. The method should find and return the average production of all the elements in the array. (5 marks)

- (f) Write a function using C statements called *CompareProduction()* which will take (10 marks)
three integer arrays (*array1*, *array2* and *array3*) and the size of the array as
parameters. The method should find the highest production of each day by
comparing *array1* and *array2* production and store in the *array3* for 7 days.
- (g) Write a function using C statements called *DisplayProduction()* to print the (5 marks)
production of a particular day. The function should take the integer *array*, the size
of the array and the day as parameters. The function validates the input and prints
the production of that day.
- (h) Implement the main method of a C program to do the followings:
- i. Create integer arrays with the names *week1_production*, *week2_production*
and *production_highest*. The arrays are of size 7. (1 mark)
 - ii. Insert machine A week 1 production details to the *week1_production* array
using the function *InputProduction()*. (1 mark)
 - iii. Input machine A week 2 production details to the *week2_production* array
using the function *InputProduction()*. (1 mark)
 - iv. Find and print the day number which has the highest production of week 1
using the function *MaximumDay()* and *DisplayProduction()*. (1 mark)
 - v. Find and print the day number which has the highest production of week 2
using the function *MaximumDay()* and *DisplayProduction()*. (1 mark)
 - vi. Find and print the day number which has the lowest production of week 1
using the function *MinimumDay()* and *DisplayProduction()*. (1 mark)
 - vii. Find and print the day number which has the lowest production of week 2
using the function *MinimumDay()* and *DisplayProduction()*. (1 mark)
 - viii. Find and print the average production of both week 1 and week 2 using the
function *AverageProduction()*. (1 mark)
 - ix. Find and print the total production of both week 1 and week 2 using the
function *TotalProduction()*. (1 mark)
 - x. Find and print the highest production of each day using the function (1 mark)
CompareProduction().

-----End of the question paper-----