

GEBZE TECHNICAL UNIVERSITY
DEPARTMENT OF COMPUTER ENGINEERING

CSE102 HW03

In this homework, you are going to implement a program which contains several sub-programs.

Function: void menu()

This is a menu function. User enters a choice to execute corresponding sub-program. When 0 is chosen, the program stops. Whenever a sub-program is finished, your program goes back to menu so that user can execute another sub-program. You are going to use do-while loop and switch case to implement this function.

The user will have four different choices, i.e., {1, 2, 3 or 0}. Any other input will be invalid, and program must display menu again when a wrong input is entered.

Sample output if this function is:

1. Horse Racing Game 2. Occurrence Counter 3. Triangle of Sequences 0. Exit
--

Function: int getInt(int mini, int maxi)

It gets an integer input value from user. It checks if the entry is within the operands [mini,maxi] (including the borders). If the entry is not within the range, function requests another input from the user. This continues until the right input within range is entered.

Function: int numberGeneratorBetweenRange(int min, int max)

This program generates a number between min and max values, including them. The number generator should sample uniformly within this range.

Function: void horseRacingGame()

In this game (sub-program), you are going to implement a virtual horse race. You must generate number of horses randomly between 3 and 5. Each horse runs the course within a randomly generated time between 10 and 20 units. The fastest horse (least race time) wins the race. There are no case of tie. If your horse is the only fastest horse or one of the fastest horse, you win. Otherwise, you lose (even there are more than one fastest horse except yours).

At the end of the race, you should output the total number of horse, your horse number, race time of each horse and the result of the race. Sample outputs are provided below.

Example Outputs:

Number of Horse: 5 Horse Number: 3 Racing starts... Horse 1: ----- Horse 2: ----- Horse 3: ----- Horse 4: ----- Horse 5: ----- You win! Winner is Horse 3.	Number of Horse: 4 Horse Number: 2 Racing starts... Horse 1: ----- Horse 2: ----- Horse 3: ----- Horse 4: ----- You lose! Winner is Horse 1 (or 4. It does not matter.).	Number of Horse: 3 Horse Number: 3 Racing starts... Horse 1: ----- Horse 2: ----- Horse 3: ----- You win! Winner is Horse 3.
--	--	--

GEBZE TECHNICAL UNIVERSITY
DEPARTMENT OF COMPUTER ENGINEERING

Function: void countOccurrence()

This sub-program asks for two positive integer values from the user. These two values are bigNumber and searchNumber. This sub-program computes number of occurrences of searchNumber in bigNumber.

Sample Outputs:

Big Number: 111 Search Number: 11 Occurrence: 2	Big Number: 209480948 Search Number: 7 Occurrence: 0	Big Number: 1231231 Search Number: 1 Occurrence: 3
---	--	--

Function: void triangleOfSequences()

This sub-program generates a random integer value between 2 and 10. It then generates a triangle of sequences as illustrated below.

Output (for 6):

1
2 4
3 6 9
4 8 12 16
5 10 15 20 25
6 12 18 24 30 36

Due Date: 25.03.2018 09:00 AM

Submission Rule: student_no.zip

Grading Policy:

1. If you do not apply submission rule, you will get –20 points.
2. In case of cheating, each participant gets –100.
3. A code which does not compile or run (shows nothing) will be graded as 0 points.

Contact: mcelenlioglu@gtu.edu.tr