MIDDLE EAST TECHNICAL UNIVERSITY, NORTHERN CYPRUS CAMPUS
CNG140 C Programming – Programming Assignment 2

Date handed out: **28 March 2017, Tuesday**
Date submission due: **11 April 2017, Tuesday**

Programming Assignment 2: Yacht

Purpose:

The main purpose of this programming assignment is to revise the topics that we have covered in the first six weeks including fundamentals of C programming, conditional statements, repetitive statements, and functions.

Description:

You will write a program for playing Yacht (dice) game between two players. Yacth is an old dice game which dates back to at least 1938. This will give you practice with all three control constructs (sequence, selection and repetition). We are including some design constraints in the "programming task" section, so you will also use functions. This will give you the experience of decomposing a problem into parts, and then implementing each part in one highly cohesive, loosely coupled function.

Don't try to compile your entire program in one "big bang"! Compile it piece by piece. Test each piece that you have compiled to make sure it works correctly before you add the next piece.

Yacht Rules:

Equipment: 5 dices and a scoresheet
Number of Players: 2 players
How to play: Each player will have a turn to roll 5 dice. The object of the game is to score points by rolling five dice to make certain combinations. The dice can be rolled **up to three times** in a turn to try to make these combinations. A game consists of **twelve rounds**. After each round the player chooses which scoring category is to be used for that round. Once a category has been used in the game, **it cannot be used again**. The scoring categories have varying point values, some of which are fixed values and others where the score depends on the value of the dice. A **Yacht** is five-of-a-kind and scores 50 points; the highest of any category. The winner is the player who scores most points at the end of the 12 rounds.
Scoresheet: A Scoresheet looks as follows:

Player 1	**Player 2**
20	50
50	63
80	63

As you can see scores are **accumulated** from the previous round.
Score calculation: Table 1 shows the scoring of each case.

How to Play Yacht?

You will write the program that will allow a player play the Yacht game against the computer. Your program will first roll a dice for both computer and the player, and whoever gets the higher score the game will start with that player. Your program will allow the players play the game for 12 rounds and the end of each round display their scores. At the end of 12 rounds, whoever has the highest score will win the game. In each round, the computer will only play once, but the other player needs to be able to re-roll the dice at most three times. If they do not like their point, they can actually ask to roll them again but they can do this at most three times.

Table 1 Scoring Rules

Category	Description	Score	Example
Ones	Any combination	The sum of dice with the number 1	⚀⚀⚀⚂⚃ scores 3
Twos	Any combination	The sum of dice with the number 2	⚁⚁⚁⚄⚅ scores 6
Threes	Any combination	The sum of dice with the number 3	⚂⚂⚂⚂⚃ scores 12
Fours	Any combination	The sum of dice with the number 4	⚃⚃⚄⚄⚄ scores 8
Fives	Any combination	The sum of dice with the number 5	⚀⚀⚁⚁⚅ scores 0
Sixes	Any combination	The sum of dice with the number 6	⚁⚂⚅⚅⚅ scores 18
Full House	Three of one number and two of another	Sum of all dice	⚁⚁⚄⚄⚄ scores 19
Four-Of-A-Kind	At least four dice showing the same face	Sum of those four dice	⚃⚄⚄⚄⚄ scores 20
Little Straight	1-2-3-4-5	Varies Often 30	⚀⚁⚂⚃⚄ scores 30
Big Straight	2-3-4-5-6	Varies Often 30	⚁⚂⚃⚄⚅ scores 30
Choice	Any combination	Sum of all dice	⚀⚀⚂⚂⚄ scores 13
Yacht	All five dice showing the same face	50	⚀⚀⚀⚀⚀ scores 50

Sample run:

```
Welcome to the Yacht game.
Lets see who is lucky!
Player: 4 - Machine: 2
Player is the lucky one, lets get started!

Round: 1
=======================================
Rolled the dice for you:
Dice1: 1, Dice2: 1, Dice3:1, Dice4: 5, Dice5: 3
Do you want to roll or not (Y/N): N
Which scoring rule you would like to use (1: ones, 2: twos, 3: threes, 4:
fours, 5: fives, 6: sixes, 7: full house, 8: four_of_a_kind, 9: little
straight, 10: big straight, 11: choice, 12: yacht): 1
Your score is: 3, and your total is: 3

Rolled the dice for computer:
Dice1: 1, Dice2: 1, Dice3:1, Dice4: 5, Dice5: 3
Rule 11 (choice) is used!
Your score is: 11 and your total is: 11

Round: 2
=======================================
Rolled the dice for you:
Dice1: 1, Dice2: 1, Dice3:1, Dice4: 5, Dice5: 3
Do you want to roll or not (Y/N): N
Which scoring rule you would like to use (1: ones, 2: twos, 3: threes, 4:
fours, 5: fives, 6: sixes, 7: full house, 8: four_of_a_kind, 9: little
straight, 10: big straight, 11: choice, 12: yacht): 1
You cannot use this, already used it!
```

```
Which scoring rule you would like to use (1: ones, 2: twos, 3: threes, 4:
fours, 5: fives, 6: sixes, 7: full house, 8: four_of_a_kind, 9: little
straight, 10: big straight, 11: choice, 12: yacht): 11
Your score is: 11 and your total is: 14


Rolled the dice for computer:
Dice1: 1, Dice2: 1, Dice3:1, Dice4: 5, Dice5: 3
Rule 1 (ones) is used!
Your score is: 3 and your total is: 14



Round: 3
========================================

Rolled the dice for you:
Dice1: 1, Dice2: 2, Dice3:3, Dice4: 4, Dice5: 5
Do you want to roll or not (Y/N): Y
Dice1: 2, Dice2: 2, Dice3:5, Dice4: 5, Dice5: 5
Do you want to roll or not (Y/N): Y
Dice1: 6, Dice2: 6, Dice3:6, Dice4: 1, Dice5: 1
Which scoring rule you would like to use (1: ones, 2: twos, 3: threes, 4:
fours, 5: fives, 6: sixes, 7: full house, 8: four_of_a_kind, 9: little
straight, 10: big straight, 11: choice, 12: yacht): 6
Your score is: 18 and your total is: 32

Rolled the dice for computer:
Dice1: 1, Dice2: 1, Dice3:1, Dice4: 1, Dice5: 1
Rule 12 (yatch) is used!
Your score is: 50 and your total is: 64

….

Round: 12
========================================

……

Winner is Computer!
```

Programming Requirements:
In order to implement this game you will need to write at least the following functions, but if you need more functions you can add them.

roll-a-dice () – This function will roll a dice and return the result. The rolling action will give random values. Each result (1-6) should be equally likely. In otherwords having each outcome should have the same probability.

play_computer() – This function will mainly be responsible from making the computer play the game.

computer_strategy_decider() – This function will decide which scoring strategy will be used. The computer will have a simple greedy strategy. It will check the scoring of all the strategies and go for the one which will give the computer highest score for that turn. However, please note that a strategy can only be used **once**. Therefore, in each round the computer will have one less strategy to check.

play-human() – This function will mainly be used to get the player play a turn.

scoresheet() – This function will be used to display the scoresheet on the screen.

Grading Schema:

Your program will be graded as follows:

Grading Point	**Mark (100)**
Maintaining the number of rounds requested by the user and also maintaining the total cores	10
roll-a-dice() function	10
play_computer() function	20
computer_strategy_decider() function	20
play_user() function	20
scoresheet() function	10
Code quality (e.g., variable names, formulation of selection statements and loops, function prototypes, etc)	10

Rules:

Please make sure that you follow the restrictions for the assignment as follows.

- Strictly obey the input output format. Do not print extra things.
- **You are not allowed to use global variables**.
- **You are not allowed to use data structures such as arrays to store values** as we have not covered them in the class yet**.**
- **You are not also allowed to use external files.**
- Name your source file “Yacht.c”.
- Upload only source file. Do not compress it (zip, rar, …)