Programming Examples

Example-4:

Black Jack
Level : Easy

Program Problem Statement

- Blackjack is a very well known gambling card game played against a dealer in a casino. In this card game, each player is trying to beat the dealer, by obtaining a sum of card values not more than 21 which is greater than the sum of dealer card values. Player is initially given 2 cards, but he could choose to **hit** (ask for 3rd card) or **stand** (no more cards). If he chooses to hit for 3rd card and total score crosses 21 he get busted (loses irrespective of the total score of dealer cards). Face cards (**Jacks**, **Queens** and **Kings**) are worth 10 points. **Aces** are worth 1 or 11, whichever is preferable. Other cards are represented by their number.

  Here, you have to implement a conservative player strategy of playing blackjack. This conservative player does not want to get busted and hit only when its safe to do so. He follows following strategy:

  hit if (score<=11) or (an ACE is held)
  stand otherwise.

  Write a program to implement the above strategy given the initial 2 cards of the player.

- The Prototype of the Function is :
- **public int hitOrStand(char firstcard, char secondcard)**
  Where the function hitOrStand takes 2 character input firstcard, secondcard and return interger output.

- Constraints
  1) Color of the card does not affects the score of the card.
  2) If the value of card is not valid then return -1.
  3) Don't consider the card with value 10.
  4) Input for Face cards must be capital letters('A','Q','J','K') else return -1

- Example 1
  - Input : firstcard = 'J' secondcard = '5'
  - Output The function returns 0.
  - **Explanation:** Here the value of J (JACK) is 10. so the sum of card values is 15, so he chooses to stand.
- Example 2
  - Input : firstcard = 4 secondcard = 3
  - Output The function returns 1.
  - **Explanation:** Here the sum of values of card 4 and 3 is 7. It is less than 11, so it makes sense for him to draw another card.

- Example 3
  - Input :   firstcard = 'A'   secondcard = 'F'.
  - Output The function returns -1
  - **Explanation:** Here F is an invalid card name.

- For Java solutions
  - Package Name:test.blackjack
  - File Name:BlackJack.java
  - Class Name:BlackJack
  - Function Name:public int hitOrStand(char firstcard, char secondcard)

- **General Instructions** The package names, class names, method signatures to be used are mentioned in the problem statement. Do not use your own names or change the method signatures and fields. You can add any number of additional methods.

Pseudo Code

- Check whether two cards are valid or not. If not valid, return -1.
- Check the value of the two cards given as input.
- Check whether the sum of the values of the two cards is greater than equal to 11 or if one of the card is ace.
- If the above condition satisfies return 1 else return 0.

Program Solution

```
•  package test.blackjack;


•  public class BlackJack {

•     public int hitOrStand(char firstcard, char secondcard)
•                 {
•                           int val1=valueOf(firstcard);
•                           int val2=valueOf(secondcard);
•                           if(val1==-1 ||val2==-1) return -1;
•           int total=val1+val2;
•                           if (total <= 11 || firstcard=='A' ||
   secondcard=='A') return 1;
•                           else return 0;
•                 }
•                 int valueOf(char card)
•                 {
•
if(card>=49 && card<=57)
                        return card-'0';
                    else if(card=='A')
                        return 11;
                    else if(card=='K'|| card=='Q'|| card=='J')
                        return 10;
                    else
                        return -1;

                }
                public static void main(String[] args)
                {
                    BlackJack obj=new BlackJack();
                    System.out.println(obj.hitOrStand('3','0'));
                }
}
```