





code.sprint^{MT}



TASKS BOOKLET
- Qualifiers Round -
Post-Secondary Category
2019



DIRECTORATE FOR LEARNING &
ASSESSMENT PROGRAMMES



Qualifiers Round Schedule

Time Allotted	Analysis	Task	Submit	Break
15:00 – 15:05	Task 1			
15:05 – 16:05		Task 1		
16:05 – 16:10			Task 1	
30 minutes				Break
16:40 – 16:45	Task 2			
16:45 – 17:45		Task 2		
17:45 – 17:50			Task 2	
30 minutes				Break
FINALISTS ARE ANNOUNCED				

Task 1 – Trente et un Card Game (60 minutes)

Trente et un or Thirty-one is a card game played between various players, where they attempt to assemble a hand which totals 31. This goal has propelled the implementation of various games such as Cribbage, Trentuno, and Wit & Reason since the 15th century.



Create a variation of this card game according to the rules below:

Program Rules:

- The game uses the standard 52-card pack; *list of cards shown in figure 1.*
- The game is played in one player mode against the computer (Player vs Computer)
- The goal is to obtain 31 points or as near as possible.
- The player is always the first to play.
- Players take turns drawing one card at a time by pressing 'D'.
- Points are collected depending on the card drawn:
 - An ACE (A) is worth 11 points
 - A KING, QUEEN or JACK is worth 10 points
 - All other cards are worth their own value.
- Computer's cards are kept hidden throughout the gameplay.
- Players may choose to stop drawing cards at any time by pressing 'S'.
- Players will automatically lose the chance to draw more cards if points exceed 31.
- The game finishes when both players stop drawing cards.
- At the end of the game, the following should be displayed:
 - The player's cards
 - The computer's cards
 - Winner is announced unless it is not a DRAW.

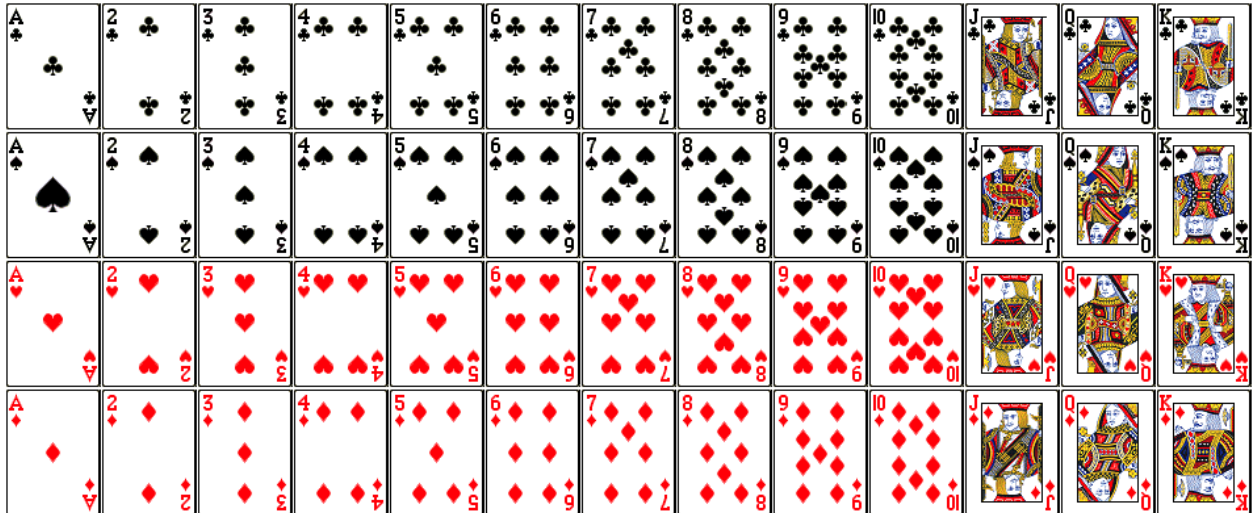


Figure 1: List of Cards (Standard Card Deck)

Name the class containing the main method **RunApp1**
 Submit your program in a folder called **TASK1_INDEXNUM**
 e.g. TASK1_0025 or TASK1_0004

Assessment Rubric (Task 1)

Program Functionality	User-Friendly Interface	Code Efficiency	Proper use of In-line Text (Comments)	Use of Proper Conventions (Camel Case, meaningful variable names etc.)	Name of Folder & Class/es	User Input
Suitable Prompts / Messages displayed	User's input Validation (Draw or Stop)	Proper Randomisation (Card Deck)	Cards Validation (No duplicate cards are drawn)	Computer "Intelligence" (Algorithm used for the computer to play the game)	Use of proper Data Structures	Other Features (Not listed in the task)

0 – Not Satisfactorily | 1- Partly Satisfactorily | 2- Entirely Satisfactorily

Maximum Score: 26 + 2 for every extra feature

Task 2 – Vending Machine (60min)

A vending machine is an automated machine that provides items, such as snacks, beverages, over the counter medicine and lottery tickets, to consumers after money or a credit card is inserted into the machine.



The earliest known reference to a vending machine is in the work of Hero of Alexandria, an engineer and mathematician in first-century Roman Egypt. His mechanical machine accepted a coin and then dispensed holy water. The first modern vending machine was invented by Percival Everitt in 1883 and introduced in London in the early 1880s, dispensing postcards. Nowadays, technology advancement made it possible to have a variety of vending machines.

Write a program for a snacks/beverage vending machine which works according to the rules below:

- The vending machine can dispense several items according to the list in table 1.
- The vending machine accepts 5c, 10c, 20c, 50c, €1 euro or €2 coins only.
- Upon execution, the vending machine displays the details of available items; i.e. the item number, description and price.
- To dispense an item, the user must:
 1. enter a valid item number;
 2. enter enough coins to pay for the chosen item. To identify coins, the user must enter:
 - number 5 to indicate a 5c coin
 - number 10 to indicate a 10c coin
 - number 20 to indicate a 20c coin
 - number 50 to indicate a 50c coin
 - number 1 to indicate a €1 coin
 - number 2 to indicate a €2 coin

- When the correct change is inserted, the vending machine:
 - displays “Item is being dispensed”, and
 - if applicable, issues the correct change in number of coins; for example: x3 20c coins & x1 5c coin

Num.	Description	Price	Num.	Description	Price
001	Twistees	0.60	009	Water	0.60
002	Peanut M&M's	0.75	010	Energy Drink	2.25
003	Twix	0.65	011	Gatorade	1.20
004	Milky Way	0.65	012	Coke	1.50
005	Doritos	1.00	013	Sprite	1.50
006	Oreos	0.85	014	Kinnie	1.50
007	Nuts Trail Mix	1.15	015	Diet Kinnie	1.50
008	Granola Bars	1.25	016	Coconut Water	2.10

Table 1: List of items

Name the class containing the main method **RunApp2**
 Submit your program in a folder called **TASK2_INDEXNUM**
 e.g. TASK2_0025 or TASK2_0004

Assessment Rubric (Task 2)

Program Functionality	User Friendly Interface	Code Efficiency	Proper use of In-line Text (Comments)	Use of Proper Conventions (Camel Case, meaningful variable names etc.)
Name of Folder & Class/es	User Input	Suitable Prompts / Messages displayed	Arithmetic Processes (correct change in coins)	Validation Processes (item num & coins entered)
Proper Use of Data Structures	Other Features (not listed in the task)	Maximum Score: 22 + 2 for every extra feature		
0 – Not Satisfactorily 1- Partly Satisfactorily 2- Entirely Satisfactorily				

