

Question 1. [10 MARKS]

Complete the following functions.

Part (a) [4 MARKS]

```
def read_player_location_data(f):
```

```
    '''f is an open text file containing lines of the form:  
    player_num:team_name:x:y
```

Return a dictionary that contains (player_num, team_name) tuples as keys
and (x, y) tuples as values.

team_name is a str that may contain letters, numbers, and spaces.

player_num, x and y are ints.

, , ,

```
players = {}
```

```
for line in f:
```

```
    data = line.rstrip().split(':')
```

```
    players[(int(data[0]), data[1])] = (int(data[2]), int(data[3]))
```

```
return players
```

Part (b) [6 MARKS] You may assume that colour values are in the range 0-255, and the x and y coordinates are in the range 0-99.

```
def display_player_locations(teams, players):
    '''Return a 100x100 picture with a black background with the (x, y) location
    for each player from the dictionary players coloured according to her team's
    colour in the dictionary teams.

    Dictionary teams contains team_names as keys, and
    (red_value, blue_value, green_value) tuples as values.

    Dictionary players contains (player_num, team_name) tuples as keys,
    and (x, y) tuples as values.

    red_value, blue_value, green_value, player_num, x and y are ints.
    team_name is a str that may contain letters, numbers, and spaces.
    '''

    field = make_empty_picture(100, 100)
    for player in players.keys():
        coords = players[player]
        pixel = get_pixel(field, coords[0], coords[1])

        team = player[1]
        colour = teams[team]
        set_red(pixel, colour[0])
        set_blue(pixel, colour[1])
        set_green(pixel, colour[2])
    return field
```

Question 2. [6 MARKS]

Recall that you wrote the following function in Assignment 4:

```
def increment_count(d, key):
    '''Increment the integer associated with the str key in dict d if the key
    is already in the dictionary; add the key to the dictionary with value
    1 if key is not in d.
    '''
```

Complete the following functions. You may use `increment_count` in either function.

Part (a) [2 MARKS]

```
def count_items(L):
    '''Return a dictionary where the keys are the items of list L and the
    values are the number of times that each item occurs in L.
    '''

    d = {}
    for item in L:
        increment_count(d, item)
    return d
```

Part (b) [4 MARKS] For function `mode`, you may assume `len(L) >= 1`. You may also use `count_items`.

```
def mode(L):
    '''Return the mode of list L. (The mode is the item that occurs most
    frequently.) If there is more than one mode, return any one of them.
    '''

    counts = count_items(L)
    max = -1
    for key in counts.keys():
        if counts[key] > max:
            max = counts[key]
            max_key = key
    return max_key
```