Unit 3 Programming Assignment

Array Algorithms

WARNING: Using built-in functions to perform the array algorithms, instead of writing them yourself, will result in a Level R to Level 1 grade.

# Option 1 Massage Room Receptionist

Create a web page that can store the contact information about massage clients.

Before the contact information can be inputted and searched, the user must log in. There should be 3 different username-password pairs that can all log in. Here are the profiles:

1. Username: guest, Password: g1

The guest can only traverse through the existing contact information. They cannot add or do any advanced searches.

1. Username: reception, Password: pass

The receptionist can add new contacts. They can search by first names or last names, and sort by increasing or decreasing LENGTH of first name. They do not traverse through the information.

1. Username: admin, Password: 4321

The admin is able to do all actions that the guest and receptionist can do. They can also do a partial match search for phone numbers. For example, search for “416”, and get all customers that have 416 in their phone number. They can also find all customers who have visited 0 times, and find the most frequent customer(s).

Information that can be stored for each client is

* First name
* Last name
* Phone number
* Times visited.

# Option 2 Battle DB

Create a web page that can store the information about monsters in a game. Think of something similar to a “Pokedex” without pictures.

Before the monster information can be inputted and searched, the user must log in. There should be 3 different username-password pairs that can all log in. Here are the profiles:

1. Username: guest, Password: monster

The guest can only traverse through the existing contact information. They cannot add or do any advanced searches.

1. Username: trainer, Password: pass

The trainer can add new monster information. They can search by monster name, and sort by increasing or decreasing health points..

1. Username: professor, Password: 4321

The professor is able to do all actions that the guest and receptionist can do. They can also do a partial match search for monster names. For example, search for “Ab”, and get all monsters that have “Ab” in their name. They can also find all monsters of a specific type/category, and the lowest health monster(s).

Each monster should have the following information:

* Monster name
* Health points
* Type/Category

Marking Rubric

# Rubric

| Communication – Documentation  Weight 3% /100 | Application – Coding  Weight 6% /100 | Thinking  Weight 2% /10 |
| --- | --- | --- |
| **Level 4: Clearly documents most to all of the following:**  Header,  Variable and constants,  Mathematical expressions,  Conditional statements,  Loops and algorithms.  Subprograms, parameters, and return values.  **Level 4: Properly names most to all**  Variables and constants,  Subprograms and parameters  **Level 4: Correctly indents**  The entire program | **Level 4: Very efficient and correct use of**  Variables and constants,  Conditional statements,  Procedures,  Mathematical expressions,  Arrays,  Array algorithms and loops,  Random numbers,  User Input,  Data type conversions.  Functions for array algorithms  Parameters for array algorithms | **Level 4:**  Demonstrates very effective individual problem solving. |
| **Level 3: Documents many of the following:**  Header,  Variable and constants,  Mathematical expressions,  Conditional statements,  Loops and algorithms,  Subprograms, parameters, and return values.  **Level 3: Properly names many**  Variables and constants,  Procedures  Subprograms and parameters  **Level 3: Correctly indents**  Almost all of the program | **Level 3: Mostly efficient and mostly correct use of, while missing one of:**  Variables and constants,  Conditional statements,  Procedures,  Mathematical expressions,  Arrays,  Array algorithms and loops,  Random numbers,  User Input,  Data type conversions.  Functions for array algorithms  Parameters for array algorithms | **Level 3:**  Demonstrates effective individual problem solving. |
| **Level 2: Documents numerous of the following:**  Header,  Variable and constants,  Mathematical expressions,  Conditional statements,  Loops and algorithms,  Subprograms, parameters, and return values.  **Level 2: Properly names numerous**  Variables and constants,  Procedures  Subprograms and parameters  **Level 2: Correctly indents**  Much of the program | **Level 2: Somewhat efficient and somewhat correct use of, or missing some of:**  Variables and constants,  Conditional statements,  Procedures,  Mathematical expressions,  Arrays,  Array algorithms and loops,  Random numbers,  User Input,  Data type conversions.  Functions for array algorithms  Parameters for array algorithms | **Level 2:**  Demonstrates ineffective individual problem solving while managing to solve some problems. |
| **Level 1: Documents some of the following:**  Header,  Variable and constants,  Mathematical expressions,  Conditional statements,  Loops and algorithms.  **Level 1: Properly names some**  Header,  Variable and constants,  Mathematical expressions,  Conditional statements,  Loops and algorithms,  Subprograms, parameters, and return values.  **Level 1: Correctly indents**  Some of the program | **Level 1: Mostly inefficient and mostly incorrect use of, or missing many of:**  Variables and constants,  Conditional statements,  Procedures,  Mathematical expressions,  Arrays,  Array algorithms and loops,  Random numbers,  User Input,  Data type conversions.  Functions for array algorithms  Parameters for array algorithms | **Level 1:**  Demonstrates highly ineffective individual problem solving but still manages to solve some problems. |
| **Level R: Documents few of:**  Header,  Variable and constants,  Mathematical expressions,  Conditional statements,  Loops and algorithms,  Subprograms, parameters, and return values.  **Level R: Properly names few of**  Variables and constant,  Procedures  Subprograms and parameters  **Level R: Correctly indents**  Little of the program | **Level R: Missing most of:**  Variables and constants,  Conditional statements,  Procedures,  Mathematical expressions,  Arrays,  Array algorithms and loops,  Random numbers,  User Input,  Data type conversions.  Functions for array algorithms  Parameters for array algorithms | **Level R:**  Demonstrates highly ineffective individual problem solving. |