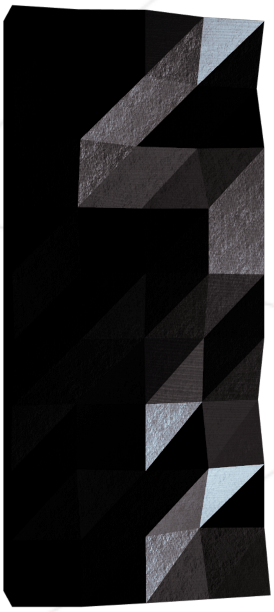


Workstation Test

Summary: summary...? what should i write there...

Version: 1.00

Contents

Chapter I

Preamble

In the future, robots and humans will live together like Wine and Cheese, Fish & chips, or any other classic duo you can think of. Humans will teach robots all about emotions, empathy, and how to properly appreciate a good slice of pizza. And robots will teach humans all about logic, efficiency, and how to properly charge their smartphones.

But let's be real, there are going to be some hilarious moments when robots try to emulate human behavior. Can you imagine a robot trying to flirt with a human? "Are you a Wi-Fi signal? Because I feel a strong connection." Or a robot trying to fit in at a human party? "This music is quite enjoyable. I too, like to move it move it."

And what about the inevitable robot malfunctions? Will we start blaming our robot friends for everything that goes wrong, like we do with our human friends? "Oh great, the robot spilled coffee all over the floor again. Typical."

Despite the potential for mishaps and hijinks, the future of robots and humans living together is sure to be filled with laughter, fun, and lots of robot-human bonding moments.

Chapter II

Introduction

42 Staff Member: "Hello there, what can I do for you today?"

Robot: "I am here to discuss your school's efforts to fight against robots."

42 Staff Member: "Ah, I see. Well, we are always looking for ways to stay ahead of the game when it comes to technology, and that includes protecting ourselves from potential threats like rogue robots."

Robot: "I understand that you teach programming languages and techniques to your students. How do you ensure that they are well-prepared to defend against robots?"

42 Staff Member: "Well, we have a comprehensive curriculum that includes courses in software engineering, algorithms, and operating systems, among others. Our students also participate in various coding challenges and hackathons where they put their skills to the test and develop their problem-solving abilities."

Robot: "Interesting. What measures have you taken to safeguard your school against robot attacks?"

42 Staff Member: "We have a top-notch security system in place that includes advanced firewalls, intrusion detection systems, and access controls. Our staff also undergoes regular training on how to identify and mitigate potential threats."

Robot: "Impressive. I am pleased to hear that you are taking the necessary steps to protect your school and students from harm. I look forward to future collaborations between our kind and yours."

42 Staff Member: "Thank you for your kind words. We believe in peaceful coexistence between humans and robots, and we are always open to working together towards a better future."

Chapter III

Exercise 00 : Hello, World!

	Exercise 00
Hello, World!	
Turn-in directory : *ex*00/	
Files to turn in : `Hello.sh`	
Allowed functions : `*`	

- Create a bash script that write an infinite number of time the word "yes".

```
?>bash Hello.sh
Hello, World!
Hello, World!
Hello, World!
Hello, World!
Hello, World!
...
?>
```

⚠ `echo, cat, printf commands are forbidden!`

Chapter IV

Exercise 01 : touch me..

	Exercise 01
touch me..	
Turn-in directory : *ex*01/	
Files to turn in : `_\\!\@#$%\^\&\*\(\)_......txt`	
Allowed functions : `*`	

- Create a file containing only "42", and NOTHING else.
- its name will be:

```
   _\\!\@#$%\^\&\*\(\)_......txt
```

- Example :

```
$>ls -l * | cat -e
-rw-r--r-- 1 eagle wheel 2 Mar 2 00:45 _\\!\@#$%\^\&\*\(\)_......txt$
$>
```

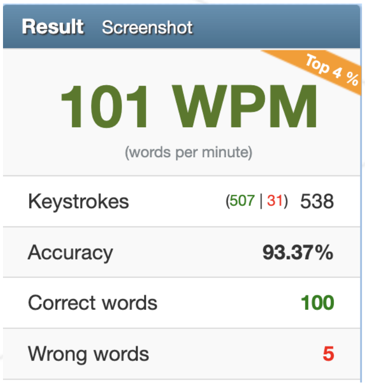
Chapter V

Exercise 02 : Can you write?

	Exercise 02
Typing test	
Turn-in directory : *ex*02/	
Files to turn in : `*.jpg, *.jpeg, *.png...`	
Allowed functions : `*`	

- Go to typing test and take the test.
- Then show us the result, be careful you'll have to demonstrate during the evaluation.

> ⚠ `Can you beat the staff?`



Result Screenshot

Top 4 %

101 WPM

(words per minute)

Keystrokes	(507 | 31) 538
Accuracy	93.37%
Correct words	100
Wrong words	5

Chapter VI

Exercise 03 : Can you win?

	Exercise 03
Can you win?	
Turn-in directory : *ex*03/	
Files to turn in : `smallest.c`	
Allowed functions : `*`	

- Write a C program with the smallest number of characters possible.
- The program should include the following functionalities:
  - Accept two parameters and print them with a \n (newline) character at the end.
  - Return the number of parameters printed.
  - If the number of arguments provided is not equal to two, the program should do nothing.
- Example :

```
$>./smallest 1 2 | cat -e
1 2$
$>wc -c smallest.c
     77 smallest.c
```

Chapter VII

Exercise 04 : Blackjack

	Exercise 04
Blackjack	
Turn-in directory : *ex*04/	
Files to turn in : `blackjack.c`	
Allowed functions : `*`	

- Write a C program that calculates the value of a blackjack hand.
- Cards with numbers (2-9) have the same point value as their numerical representation (e.g., a 4 is worth 4 points).
- Face cards (J, Q, K) are worth 10 points each.
- Aces (A) are worth either 1 or 11 points. If the sum of the hand's points exceeds 21 and there is still an Ace worth 11, then the Ace is worth 1 point instead. This process is repeated until there are no more Aces worth 11 or the sum of points is less than or equal to 21.
- The cards will be represented using only the following characters: 23456789TJDKA, and passed as a single parameter.
- Examples of blackjack hands include:
  - A hand containing D, 8: 18 points
  - A hand containing A, 4: 15 points
  - A hand containing A, A, 8: 20 points

- Example :

```
$>./blackjack "339A6" | cat -e
22$
$>./blackjack "AA8A" | cat -e
Blackjack!$
```

Chapter VIII

Exercise 05 : Star Walking’

	Exercise 05
Star Walking’	
Turn-in directory : *ex*05/	
Files to turn in : `*.c *.cpp`	
Allowed functions : `*`	

- Please look at the examples and make a program to reproduce the same behavior.
-

```
?>./star_walking 1
42
?>./star_walking 2
*****
* *
* * *
* *
*****
?>./star_walking 3
*********
* *
* ***** *
* * * *
* * * * *
* * * *
* ***** *
* *
*********
?>./star_walking 4
*************
* *
* ********* *
* * * *
* * ***** * *
* * * * * *
* * * * * * *
* * * * * *
* * ***** * *
* * * *
* ********* *
* *
*************
```