Assignment 1

Exercise 00: vc_print_alphabet

Turn-in files	**vc_print_alphabet.c**
Allowed functions	putchar

- Create a function that displays the alphabet in lowercase, on a single line, by ascending order, starting from a letter 'a'
- Function prototype: `void vc_print_alphabet(void);`

Exercise 01: vc_print_reverse_alphabet

Turn-in files	**vc_print_reverse_alphabet.c**
Allowed functions	putchar

- Create a function that displays the alphabet in lowercase, on a single line, by descending order, starting from the letter ’z’.
- Function prototype: `void vc_print_reverse_alphabet(void);`

Exercise 02: vc_print_numbers

Turn-in files	**vc_print_numbers.c**
Allowed functions	putchar

- Create a function that displays all digits, on a single line, by ascending order.
- Function prototype: `void vc_print_numbers(void);`

Exercise 03: vc_is_negative

Turn-in files	**vc_is_negative.c**
Allowed functions	putchar

- Create a function that displays ’N’ or ’P’ depending on the integer’s sign entered as a parameter. If n is negative, display ’N’. If n is positive or null, display ’P’.
- Function prototype: `void vc_is_negative(int n);`

Exercise 04: vc_print_comb

Turn-in files	**vc_print_comb.c**

Turn-in files	vc_print_comb.c
Allowed functions	putchar

- Create a function that displays all different combinations of three different digits in ascending order, listed by ascending order - yes, repetition is voluntary.
- Here's the intended output:

```
$ ./your_program | cat -e
012, 013, 014, 015, 016, 017, ..., 789$
```

- 987 isn't there because 789 already is.
- 999 isn't there because the digit 9 is present more than once.
- reference: `man cat`
- Function prototype: `void vc_vc_print_comb(void);`

Exercise 05: vc_print_comb2

Turn-in files	vc_print_comb2.c
Allowed functions	putchar

- Create a function that displays all different combinations of two digits between 00 and 99, listed by ascending order.
- Here's the intended output:

```
$ ./your_program | cat -e
00 01, 00 02, 00 03, 00 04, 00 55, ..., 00 99, 01 02, ..., 97 99, 98 99$
```

- Function prototype: `void vc_vc_print_comb2(void);`

Exercise 06: vc_putnbr

Turn-in files	vc_putnbr.c
Allowed functions	putchar

- Create a function that displays the number entered as a parameter. The function has to be able to display all possible values within an **int** type variable.
- Function prototype: `void vc_putnbr(int nb);`
- For example: `vc_putnbr(88) displays "88"`

Exercise 07: vc_print_combn

Turn-in files	**vc_print_combn.c**
Allowed functions	putchar

- Create a function that displays all different combinations of **n** numbers by ascending order.
- n will be so that : 0 < n < 10
- If n = 2, here's the expected output:

```
$ ./your_program | cat -c
01, 02, 03, ..., 09, 12, ..., 79, 89$
```

- Function prototype: `void vc_print_combn(int n);`