Program 2 : write a python program that defines a function to generate random passwords of a specified length the function takes an optional parameter length which is set to 8 by default , if no length is specified by the user , the password will have 8 characters.

Source code:

```
import random
import string

def generate_random_password(length=8):
    characters = string.ascii_letters + string.digits + string.punctuation
    password = ''.join(random.choice(characters) for _ in range(length))
    return password



a = input("Do you Want to Generate password of Specified length? Enter Yes or No:")

if a == "No":
    default_password = generate_random_password()
    print("Default Password:", default_password)
else:
    password = int(input("Enter the length:"))
    custom_length_password = generate_random_password(password)
    print("Custom Length Password:", custom_length_password)
```

Explanation of the `generate_random_password` function step by step:

1. `def generate_random_password(length=8):`:
   - This line defines a Python function named `generate_random_password`.
   - It takes one parameter, `length`, with a default value of 8. This means that if you don't specify a length when calling the function, it will default to 8 characters.

2. `characters = string.ascii_letters + string.digits + string.punctuation`:
   - This line creates a string called `characters` by concatenating three different strings together:
     - `string.ascii_letters`: Contains all the uppercase and lowercase letters (A-Z and a-z).
     - `string.digits`: Contains all the digits (0-9).
     - `string.punctuation`: Contains various punctuation characters like !, @, #, etc.
   - `characters` now contains all the characters that can be used to create the random password.

3. `password = ''.join(random.choice(characters) for _ in range(length))`:
   - This line generates the random password.
   - `random.choice(characters)` selects a random character from the `characters` string for each iteration of the loop.
   - `for _ in range(length)` repeats the process `length` times. The `_` is used as a convention to indicate that the loop variable is not actually used within the loop; it's just a throwaway variable.
   - `''.join(...)` takes all the randomly selected characters and joins them together into a single string. The `''` within the `join` function specifies that an empty string should be used as a separator, effectively concatenating the characters one after the other.
   - The resulting string, `password`, is the random password of the specified length.

4. `return password`:
   - This line returns the generated `password` as the output of the function.

So, the `generate_random_password` function takes an optional `length` parameter (defaulting to 8), creates a character set from letters, digits, and punctuation, and generates a random password by selecting characters from this set for the specified length. The function then returns the random password as a string.