Week6_programs

January 16, 2025

[7]:

```
'''1. Write a function that accepts a positive integer as a parameter and then
returns a representation of that number in binary (base 2).
Hint: This is in many ways a trick question. Think!'''

def binary(n):
    if n <= 0:
        return "Input must be a positive integer"

    binary_rep = ""
    while n > 0:
        binary_rep = str(n % 2) + binary_rep
        n = n // 2

    return binary_rep


user_input = int(input("Enter a positive integer: "))

print(f"The binary representation of {user_input} is: {binary(user_input)}")
```

```
Enter a positive integer:  98
```

```
The binary representation of 98 is: 1100010
```

[6]:

```
'''2. Write and test a function that takes an integer as its parameter and␣
 ↪returns
the factors of that integer. (A factor is an integer which can be multiplied by
another to yield the original).'''

def factors(n):
    print(f"The factors of {n} are:")
    for i in range(1, n + 1):
        if n % i == 0:
            print(i)

user_input=int(input("Enter an integer:"))
factors(user_input)
```

```
Enter an integer: 30
```

```
The factors of 30 are:
1
2
3
5
6
10
15
30
```

[9]:

```python
'''3. Write and test a function that determines if a given integer is a prime
number. A prime number is an integer greater than 1 that cannot be produced by
multiplying two other integers.'''

def prime(n):
    if n<=0:
        print(n, "is not a prime number.")
        return

    for i in range(2,n):
        if n % i == 0:
            print(n, "is not a prime number.")
            return
    print(n, "is a prime number.")

user_input = int(input("Enter an integer:"))

prime(user_input)
```

```
Enter an integer: 7
```

```
7 is a prime number.
```

[10]:

```python
'''4. Computers are commonly used in encryption. A very simple form of␣
 ↪encryption
(more accurately "obfuscation") would be to remove the spaces from a message␣
 ↪and
reverse the resulting string. Write, and test, a function that takes a string
containing a message and "encrypts" it in this way.'''

def encrypt_message(message):
    no_spaces = message.replace(" ", "")

    reversed_message = ""
    for char in no_spaces:
        reversed_message = char + reversed_message

    return reversed_message
```

```python

user_input = input("Enter a message to encrypt: ")

print("Encrypted message:", encrypt_message(user_input))
```

```
Enter a message to encrypt:  This is week 6 program.

Encrypted message: .margorp6keewsisihT
```

[7]:

```python
'''5. Another way to hide a message is to include the letters that make it up
within seemingly random text. The letters of the message might be every fifth
character, for example. Write and test a function that does such encryption. It
should randomly generate an interval (between 2 and 20), space the message out
accordingly, and should fill the gaps with random letters. The function should
return the encrypted message and the interval used.
For example, if the message is "send cheese", the random interval is 2, and for
clarity the random letters are not random:
send cheese
sendcheese
sxyexynxydxy cxyhxyexyexysxye'''

import random
import string

def encrypt_message(message):
    message = message.replace(" ", "")
    interval = random.randint(2, 20)

    encrypted_message = ""

    for letter in message:
        encrypted_message += letter

        for i in range(interval - 1):
            encrypted_message += random.choice(string.ascii_lowercase)

    return encrypted_message, interval

user_message = input("Enter your message to encrypt: ")

encrypted_message, interval = encrypt_message(user_message)

print("Encrypted message: ", encrypted_message)
print("Interval used: ", interval)
```

```
Enter your message to encrypt:  send cheese

Encrypted message:  svejnbdycfhzeieasoez
Interval used:  2
```

[11]:

```python
'''6. Write a program that decrypts messages encoded as above.'''

import random
import string

def decrypt_message(encrypted_message, interval):
    decrypted_message = ""

    for i in range(0, len(encrypted_message), interval):
        decrypted_message += encrypted_message[i]

    return decrypted_message

encrypted_message = input("Enter the encrypted message: ")
interval = int(input("Enter the interval used for encryption: "))

original_message = decrypt_message(encrypted_message, interval)

print(f"The decrypted message is {original_message}.")
```

```
Enter the encrypted message:  svejnbdycfhzeieasoez
Enter the interval used for encryption:  2

The decrypted message is sendcheese.
```