Homework #6

01286121 Computer Programming

Software Engineering Program,

Department of Computer Engineering,

School of Engineering, KMITL

By

68011278   Ananda   Stallard

1.

Code:

```python
valid = False
hr24, min24 = 0, 0

while valid == False:
    hr24, min24 = input("Please enter a 24 hr format time: ").split(":")
    hr24, min24 = int(hr24), int(min24)
    if 0 > hr24 or hr24 > 24 or 0 > min24 or min24 > 60:
        print("Invalid time.")
    else:
        valid = True

def time_24_to_12_hr(hr, min):
    if hr < 12:
        ap = "AM"
    else:
        ap = "PM"

    if hr == 0:
        hr12 = 12
    elif hr == 12:
        hr12 = 12
    elif 24 > hr > 12:
        hr12 = hr - 12
    elif hr == 24:
        hr12 = 0
    else:
        hr12 = hr

    print(f"{hr12}:{min:02d} {ap}")

time_24_to_12_hr(hr24, min24)
```

Result:

```
Please enter a 24 hr format time: 23:24
11:24 PM
```

2.

Code:

```python
import turtle

month = int(input("Please input a month (Month number): "))
year = 2025

month_days = {
    1: 31, 2: 28, 3: 31, 4: 30, 5: 31, 6: 30,
    7: 31, 8: 31, 9: 30, 10: 31, 11: 30, 12: 31
}

month_name = {
    1: "January", 2: "February", 3: "March", 4: "April", 5: "May", 6: "June",
    7: "July", 8: "August", 9: "September", 10: "October", 11: "November", 12: "December"
}

name = month_name[month]
days = month_days[month]

offset = 2
target = month
total_days = offset

for m, d in month_days.items():
    if m == target:
        break
    total_days += d

month_offset = total_days % 7
```

```python
rows = ((days + month_offset - 1) // 7) + 1
cell_w = 60
cell_h = 50
width = 7 * cell_w
height = (rows + 2) * cell_h


def draw_calendar():
    turtle.pd()
    for _ in range(2):
        turtle.forward(width)
        turtle.right(90)
        turtle.forward(height)
        turtle.right(90)

    for i in range(1, rows + 2):
        turtle.penup()
        turtle.goto(-width//2, height//2 - i*cell_h)
        turtle.pendown()
        turtle.forward(width)

    for i in range(1, 7):
        turtle.penup()
        turtle.goto(-width//2 + i*cell_w, height//2 - cell_h)
        turtle.pendown()
        turtle.goto(-width//2 + i*cell_w, -height//2)

def add_info():
    turtle.penup()
    turtle.goto(0, height//2 - cell_h + 10)
    turtle.write(f"{name} {year}", align="center", font=("Arial", 24, "bold"))

    days_name = ["Mo", "Tu", "We", "Th", "Fr", "Sa", "Su"]
    for i, d in enumerate(days_name):
        cx = -width//2 + i*cell_w + cell_w/2
        cy = height//2 - 2*cell_h + 10
        turtle.penup()
        turtle.goto(cx, cy)
        turtle.write(d, align="center", font=("Arial", 14, "normal"))
```

```python
        day = 1
        for r in range(rows):
            for c in range(7):
                if r == 0 and c < month_offset:
                    continue
                if day > days:
                    return
                x = -width//2 + c*cell_w + 6
                y = height//2 - (r+3)*cell_h + 12
                turtle.penup()
                turtle.goto(x, y)
                turtle.write(str(day), font=("Arial", 14, "normal"))
                day += 1


turtle.speed(0)
turtle.hideturtle()
turtle.penup()
turtle.goto(-width//2, height//2)
draw_calendar()
add_info()
turtle.done()
```

Result:

August 2025

Mo	Tu	We	Th	Fr	Sa	Su
				1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

3.

Code:

```python
number = input("Enter a number: ")

units = ["", "one", "two", "three", "four", "five", "six", "seven", "eight", "nine"]
teens = ["ten", "eleven", "twelve", "thirteen", "fourteen", "fifteen",
         "sixteen", "seventeen", "eighteen", "nineteen"]
tens = ["", "", "twenty", "thirty", "forty", "fifty",
        "sixty", "seventy", "eighty", "ninety"]

def read(num):
    inum = int(num)
    if inum == 0:
        return "zero"
    words = []

    if inum >= 100:
        words.append(units[inum // 100] + " hundred and")
        inum %= 100
        if inum > 0:
            words.append("")

    if 10 <= inum <= 19:
        words.append(teens[inum - 10])
    else:
        if inum >= 20:
            tens_part = tens[inum // 10]
            unit_part = units[inum % 10]
            if unit_part:
                words.append(f"{tens_part}-{unit_part}")
            else:
                words.append(tens_part)
        elif inum > 0:
            words.append(units[inum])

    to_print = []
    for w in words:
        if w:
            to_print.append(w)
```

```python
    return " ".join(to_print)


num = int(number)
if 0 <= num <= 999:
    print(read(number))
else:
    print("I don't know")
```

Result:

4.

Code:

```python
money = int(input("Input your amount of money: "))

thou_note = 0
five_hund_note = 0
one_hund_note = 0
fifty_note = 0
twenty_note = 0
ten_coin = 0
five_coin = 0
two_coin = 0
one_coin = 0

while money > 0:
    if money >= 1000:
        money -= 1000
        thou_note += 1
    elif money >= 500:
        money -= 500
        five_hund_note += 1
    elif money >= 100:
        money -= 100
        one_hund_note += 1
    elif money >= 50:
        money -= 50
        fifty_note += 1
    elif money >= 20:
```

```python
        money -= 20
        twenty_note += 1
    elif money >= 10:
        money -= 10
        ten_coin += 1
    elif money >= 5:
        money -= 5
        five_coin += 1
    elif money >= 2:
        money -= 2
        two_coin += 1
    else:
        money -= 1
        one_coin += 1

if thou_note >= 1:
    print(f"1000-baht notes: {thou_note}")
if five_hund_note >= 1:
    print(f"500-baht notes: {five_hund_note}")
if one_hund_note >= 1:
    print(f"100-baht notes: {one_hund_note}")
if fifty_note >= 1:
    print(f"50-baht notes: {fifty_note}")
if twenty_note >= 1:
    print(f"20-baht notes: {twenty_note}")
if ten_coin >= 1:
    print(f"10-baht coins: {ten_coin}")
if five_coin >= 1:
    print(f"5-baht coins: {five_coin}")
if two_coin >= 1:
    print(f"2-baht coins: {two_coin}")
if one_coin >= 1:
    print(f"1-baht coins: {one_coin}")
```

Result:

```
Input your amount of money: 1693
1000-baht notes: 1
500-baht notes: 1
100-baht notes: 1
50-baht notes: 1
20-baht notes: 2
2-baht coins: 1
1-baht coins: 1
```

5.

Code:

```python
def reverse(number):
    num = list(str(number))
    rev = ""
    for i in num[::-1]:
        rev += i
    return rev

print(reverse(3456))
```

Result:

yolkai@Anandas-MacBook-Pro python % /usr/
6543