



برنامه‌ای بنویسید که از کاربر یک عدد صحیح مثبت دریافت کند و بزرگ‌ترین رقم فرد آن را پیدا کند. اگر عدد هیچ رقم فردی نداشت، پیام مناسب نمایش دهد. برنامه باید تا زمانی که کاربر یک عدد معتبر وارد نکرده است، از او بخواهد که عدد صحیح مثبت وارد کند.

Input 1:

Enter a positive integer: 3489

Output 1:

Largest odd digit: 9



Input 2:

Enter a positive integer: -123

Enter a positive integer: 0

Enter a positive integer: 2864

Output 2:

No odd digit found.

برنامه‌ای بنویسید که از کاربر یک عدد صحیح مثبت دریافت کند و بزرگ‌ترین رقم فرد آن را پیدا کند. اگر عدد هیچ رقم فردی نداشت، پیام مناسب نمایش دهد. برنامه باید تا زمانی که کاربر یک عدد معتبر وارد نکرده است، از او بخواهد که عدد صحیح مثبت وارد کند.

1 برنامه باید تا زمانی که کاربر یک عدد معتبر وارد نکرده است، از او بخواهد که عدد صحیح مثبت وارد کند.

2 بزرگ‌ترین رقم فرد آن را پیدا کند. اگر عدد هیچ رقم فردی نداشت، پیام مناسب نمایش دهد

1

برنامه باید تا زمانی که کاربر یک عدد معتبر وارد نکرده است، از او بخواهد که عدد صحیح مثبت وارد کند.

```
n = int(input("Enter a positive integer: "))
```

1 2 :

3

1

برنامه باید تا زمانی که کاربر یک عدد معتبر وارد نکرده است، از او بخواهد که عدد صحیح مثبت وارد کند.

```
n = int(input("Enter a positive integer: "))
```

```
while 2:
```

3

برنامه باید تا زمانی که کاربر یک عدد معتبر وارد نکرده است، از او بخواهد که عدد صحیح مثبت وارد کند.

```
n = int(input("Enter a positive integer: "))
```

```
while n <= 0:
```

3

برنامه باید تا زمانی که کاربر یک عدد معتبر وارد نکرده است، از او بخواهد که عدد صحیح مثبت وارد کند.

```
n = int(input("Enter a positive integer: "))

while n <= 0:
    print("Invalid input! Please enter a positive integer.")
    n = int(input("Enter a positive integer: "))
```



بزرگ‌ترین رقم فرد آن را پیدا کند. اگر عدد هیچ رقم فردی نداشت، پیام مناسب نمایش دهد

```
n = int(input("Enter a positive integer: "))

while n <= 0:
    print("Invalid input! Please enter a positive integer.")
    n = int(input("Enter a positive integer: "))
```

پیش از این ↑

```
temp = n
```

```
while 1:
```

```
    2
```

```
        3
```

```
        2
```


2

بزرگ‌ترین رقم فرد آن را پیدا کند. اگر عدد هیچ رقم فردی نداشت، پیام مناسب نمایش دهد

```
temp = n
```

```
while temp > 0:
```

2

3

2

2

بزرگ‌ترین رقم فرد آن را پیدا کند. اگر عدد هیچ رقم فردی نداشت، پیام مناسب نمایش دهد

```
temp = n
```

```
while temp > 0:
```

```
    digit = temp % 10
```

3

```
    temp //= 10
```

بزرگ‌ترین رقم فرد آن را پیدا کند. اگر عدد هیچ رقم فردی نداشت، پیام مناسب نمایش دهد

```
2
temp = n

while temp > 0:
    digit = temp % 10
    if 1:
        if 2:
            3
    temp //= 10
```

بزرگ‌ترین رقم فرد آن را پیدا کند. اگر عدد هیچ رقم فردی نداشت، پیام مناسب نمایش دهد

```
2
temp = n

while temp > 0:
    digit = temp % 10
    if digit % 2 == 1:
        if 2:
        3
    temp //= 10
```

بزرگ‌ترین رقم فرد آن را پیدا کند. اگر عدد هیچ رقم فردی نداشت، پیام مناسب نمایش دهد

```
max_odd = -1
temp = n

while temp > 0:
    digit = temp % 10
    if digit % 2 == 1:
        if digit > max_odd:
            max_odd = digit
    temp //= 10
```

بزرگ‌ترین رقم فرد آن را پیدا کند. اگر عدد هیچ رقم فردی نداشت، پیام مناسب نمایش دهد

```
max_odd = -1
temp = n

while temp > 0:
    digit = temp % 10
    if digit % 2 == 1:
        if digit > max_odd:
            max_odd = digit
    temp //= 10
```

?

بزرگ‌ترین رقم فرد آن را پیدا کند. اگر عدد هیچ رقم فردی نداشت، پیام مناسب نمایش دهد

```
max_odd = -1
temp = n

while temp > 0:
    digit = temp % 10
    if digit % 2 == 1:
        if digit > max_odd:
            max_odd = digit
    temp //= 10

if max_odd == -1:
    print("No odd digit found.")
else:
    print("Largest odd digit:", max_odd)
```



برنامه‌ای بنویسید که از کاربر یک عدد صحیح مثبت دریافت کند و بزرگ‌ترین رقم فرد آن را پیدا کند. اگر عدد هیچ رقم فردی نداشت، پیام مناسب نمایش دهد. برنامه باید تا زمانی که کاربر یک عدد معتبر وارد نکرده است، از او بخواهد که عدد صحیح مثبت وارد کند.

```
n = int(input("Enter a positive integer: "))

while n <= 0:
    print("Invalid input! Please enter a positive integer.")
    n = int(input("Enter a positive integer: "))

max_odd = -1
temp = n

while temp > 0:
    digit = temp % 10
    if digit % 2 == 1:
        if digit > max_odd:
            max_odd = digit
    temp //= 10

if max_odd == -1:
    print("No odd digit found.")
else:
    print("Largest odd digit:", max_odd)
```





ضرب المثل آمریکایی:

فیل را یک لقمه به یک لقمه بخور



الگوریتم‌های تقسیم و غلبه

Divide and Conquer

حدس عدد با تعداد تلاش محدود

برنامه‌ای بنویسید که یک عدد تصادفی بین ۱ تا ۱۰۰ انتخاب کند. سپس از کاربر بخواهد که آن را حدس بزند.

- اگر حدس کاربر کمتر از عدد باشد، پیام **"Too low! Try again"** نمایش دهد.
- اگر حدس کاربر بیشتر از عدد باشد، پیام **"Too high! Try again"** نمایش دهد.
- اگر حدس درست باشد، پیام **"Congratulations! You guessed the number"** نمایش دهد و حلقه متوقف شود.
- کاربر حداکثر ۷ بار فرصت دارد تا عدد را حدس بزند. اگر بعد از ۷ تلاش موفق نشد، پیام **"Game over! The number was X"** نمایش دهد و حلقه متوقف شود.



حدس عدد با تعداد تلاش محدود

```
Guess the number (1-100): 50
Too high! Try again.
Guess the number (1-100): 25
Too low! Try again.
Guess the number (1-100): 37
Too low! Try again.
Guess the number (1-100): 42
Congratulations! You guessed the number.
```

```
Guess the number (1-100): 10
Too low! Try again.
Guess the number (1-100): 80
Too high! Try again.
Guess the number (1-100): 50
Too low! Try again.
Guess the number (1-100): 60
Too low! Try again.
Guess the number (1-100): 70
Too high! Try again.
Guess the number (1-100): 65
Too high! Try again.
Guess the number (1-100): 63
Game over! The number was 62.
```

حدس عدد با تعداد تلاش محدود

```
import random

secret_number = random.randint(1, 100)
attempts = 0
max_attempts = 7
```

دستور `import` را در آینده یاد می‌گیریم

برنامه به طور خودکار یک عدد تصادفی بین ۱ تا ۱۰۰ داخل متغیر `secret_number` می‌گذارد

حدس عدد با تعداد تلاش محدود

```
import random
```

```
secret_number = random.randint(1, 100)
```

```
attempts = 0
```

```
max_attempts = 7
```

```
while 1:
```

```
2
```

```
if 3:
```

```
3
```

```
elif 3:
```

```
3
```

```
else:
```

```
4
```

```
5
```

حدس عدد با تعداد تلاش محدود

```
import random
```

```
secret_number = random.randint(1, 100)
```

```
attempts = 0
```

```
max_attempts = 7
```

```
while attempts < max_attempts:
```

2

```
if [redacted]:
```

3

```
elif [redacted]:
```

3

```
else:
```

4

5

حدس عدد با تعداد تلاش محدود

```
import random

secret_number = random.randint(1, 100)
attempts = 0
max_attempts = 7

while attempts < max_attempts:
```

```
    guess = int(input("Guess the number (1-100): "))
    attempts += 1

    if [REDACTED]:
```

```
        [REDACTED]
```

```
    elif [REDACTED]:
```

```
        [REDACTED]
```

```
    else:
```

```
        [REDACTED]
```

```
    [REDACTED]
```

حدس عدد با تعداد تلاش محدود

```
import random

secret_number = random.randint(1, 100)
attempts = 0
max_attempts = 7

while attempts < max_attempts:
    guess = int(input("Guess the number (1-100): "))
    attempts += 1

    if guess < secret_number:
        print("Too low! Try again.")
    elif guess > secret_number:
        print("Too high! Try again.")
    else:
```

4

5

حدس عدد با تعداد تلاش محدود

```
import random

secret_number = random.randint(1, 100)
attempts = 0
max_attempts = 7

while attempts < max_attempts:
    guess = int(input("Guess the number (1-100): "))
    attempts += 1

    if guess < secret_number:
        print("Too low! Try again.")
    elif guess > secret_number:
        print("Too high! Try again.")
    else:
        print("Congratulations! You guessed the number.")
        break
```

حدس عدد با تعداد تلاش محدود

```
import random

secret_number = random.randint(1, 100)
attempts = 0
max_attempts = 7

while attempts < max_attempts:
    guess = int(input("Guess the number (1-100): "))
    attempts += 1

    if guess < secret_number:
        print("Too low! Try again.")
    elif guess > secret_number:
        print("Too high! Try again.")
    else:
        print("Congratulations! You guessed the number.")
        break

if attempts == max_attempts and guess != secret_number:
    print("Game over! The number was", secret_number)
```



محاسبه مجموع اعداد مثبت و نادیده گرفتن اعداد منفی

برنامه‌ای بنویسید که از کاربر تعدادی عدد صحیح دریافت کند و مجموع اعداد مثبت را محاسبه کند.

- اگر کاربر عدد منفی وارد کند، برنامه باید بدون پردازش آن عدد، از او عدد جدیدی بخواهد.
- اگر کاربر مقدار ۰ را وارد کند، برنامه باید متوقف شود و مجموع اعداد مثبت را نمایش دهد.

محاسبه مجموع اعداد مثبت و نادیده گرفتن اعداد منفی

Input 1:

Enter a number: 5
Enter a number: -3
Enter a number: 10
Enter a number: -8
Enter a number: 7
Enter a number: 0

Output 1:

Sum of positive numbers: 22

Input 2:

Enter a number: -10
Enter a number: -5
Enter a number: -2
Enter a number: 0

Output 2:

Sum of positive numbers: 0

محاسبه مجموع اعداد مثبت و نادیده گرفتن اعداد منفی

```
total = 0

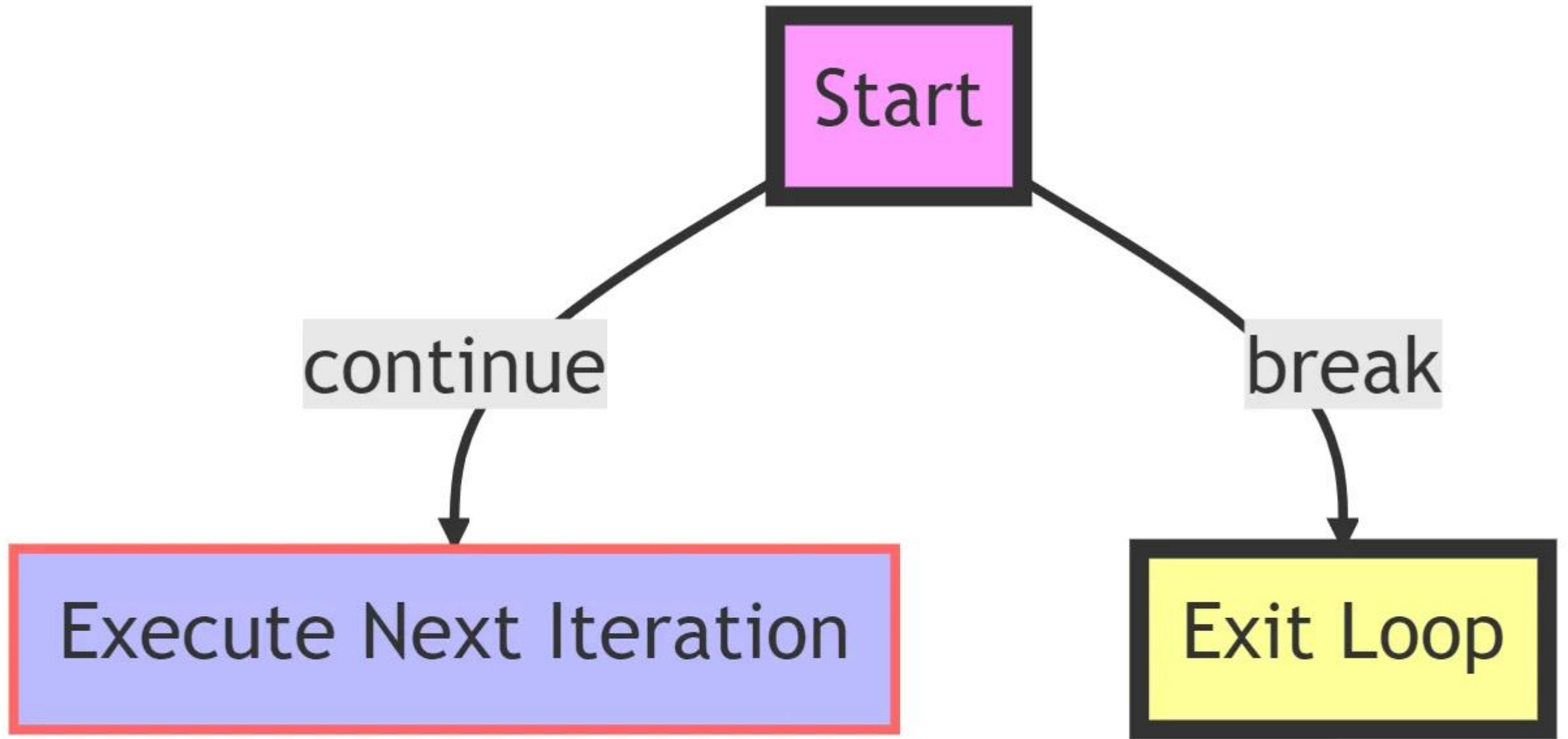
while True:
    num = int(input("Enter a number: "))

    if num == 0:
        break

    if num < 0:
        continue

    total += num

print("Sum of positive numbers:", total)
```



`while` expression:

`#inside while -loop`

`if condition:`

Guru99.com

`break`

`#inside while-loop`

`#while loop-exit`

`for var in range(len(my_list)):`

`#inside for -loop`

`if condition:`

Guru99.com

`continue`

`#inside for-loop`

`#exit for-loop`

```
while True:  
    print("Hello")  
    break
```




```
x = 5  
while x > 0:  
    continue  
    x -= 1
```



```
print(1)
```

```
print(2)
```

```
print(3)
```

```
print(4)
```

```
print(5)
```

```
print(1)
```

```
print(2)
```

```
print(3)
```

```
print(4)
```

```
print(5)
```

1

2

3

4

5

```
print(1, end=" ")  
print(2, end=" ")  
print(3, end=" ")  
print(4, end=" ")  
print(5)
```

```
print(1, end=" ")  
print(2, end=" ")  
print(3, end=" ")  
print(4, end=" ")  
print(5)
```

1 2 3 4 5

```
print(10, end="-")  
print(20, end="-")  
print(30, end="-")  
print(40, end="-")  
print(50)
```

10-20-30-40-50

```
print("Apple", end=" | ")  
print("Banana", end=" | ")  
print("Cherry", end=" | ")  
print("Date")
```

Apple | Banana | Cherry | Date

```
num = 5
while num > 0:
    print(num, end="... ")
    num -= 1
print("Blast off!")
```

5... 4... 3... 2... 1... Blast off!

ساخت جدول ضرب

برنامه‌ای بنویسید که یک عدد n از کاربر دریافت کند و جدول ضرب $n \times n$ را نمایش دهد.



ساخت جدول ضرب

Input 1:

Enter a number: 3

Output 1:

1	2	3
2	4	6
3	6	9

Input 2:

Enter a number: 5

Output 2:

1	2	3	4	5
2	4	6	8	10
3	6	9	12	15
4	8	12	16	20
5	10	15	20	25

ساخت جدول ضرب

```
n = int(input("Enter a number: "))
```

```
row = 1
```

```
while row <= n:
```

```
    col = 1
```

```
    while col <= n:
```

```
        print(row * col, end="  ")
```

```
        col += 1
```

```
    print()
```

```
    row += 1
```

حلقه‌ی بیرونی

حلقه‌ی درونی