In [1]:

```
n = int(input())
m = int(input())
for i in range(n,m+1):
    print(i,end=" ")
```

```
20
25
20 21 22 23 24 25
```

In [3]:

```
n = int(input())
m = int(input())
s = int(input())
for i in range(n,m+1,s):
    print(i,end=" ")
```

```
200
350
20
200 220 240 260 280 300 320 340
```

In [21]:

```
ran = int(input())
f,f1,f2=0,1,0
# print(f,f1,end=" ")
for r in range(ran):
#     f2=f+f1
    f,f1=f1,f2
#     print(f2,end=" ")
    f2=f+f1
    print(f2,end=" ")
```

```
20
1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987 1597
2584 4181 6765
```

Input: 2

11

Output:

12AK1A0502

|

|

12AK1A0509

12AK1A0510

In [29]:

```
start = int(input())
end = int(input())
for h in range(start,end+1):
#     if h<10:
#         print("12AK1A050{}".format(h))
#     else:
#         print("12AK1A05{}".format(h))
    print("12AK1A05{:02}".format(h))
```

```
2
10
12AK1A0502
12AK1A0503
12AK1A0504
12AK1A0505
12AK1A0506
12AK1A0507
12AK1A0508
12AK1A0509
12AK1A0510
```

In [32]:

```
st = int(input())
ed = int(input())
while st!=ed+1:
    print(st,end=" ")
    st+=1
```

```
200
210
200 201 202 203 204 205 206 207 208 209 210
```

Input: 2346287348253871825663489126834 7

Output: Given number is 2346287348253871825663489126834 7
and its length is: 32

In [36]:

```
n = int(input())
co=0
m=n
while n!=0:
    n=n//10
    co+=1
print("Given number is: {} and its length is: {}".format(m,co)
```

```
29387409261397468712364987612039847213 4
29387409261397468712364987612039847213 4 39
```

Input: 287423

Output: Given number is 287423
Even numbers are: 2 8 4 2

In [8]:

```
nu = int(input())
rv = ec = od = 0
print("Given number is: {}".format(nu),end=" ")
while nu!=0:
    rv = rv*10+nu%10
    nu = nu//10
# print(rv)
print("\nEven numbers are: ",end=" ")
odr = rv
while rv!=0:
    k = rv%10
    if k%2==0:
        print(k,end=" ")
        ec+=1
    rv = rv//10
print("\nOdd numbers are: ",end=" ")
while odr!=0:
    k = odr%10
    if k%2!=0:
        print(k,end=" ")
        od+=1
    odr = odr//10
print("\nEven digit count is: {}".format(ec),end=" ")
print("\nOdd digit count is: {}".format(od))
```

```
287423
Given number is: 287423
Even numbers are:  2 8 4 2
Odd numbers are:  7 3
Even digit count is: 4
Odd digit count is: 2
```

Jumping Statements:

- continue
- break
- pass
- return

In [14]:

```
n = int(input())
m = int(input())
d = int(input())
for j in range(n,m+1):
    if j==d:
#         break
        continue
    else:
        print(j,end=" ")
```

```
4
10
5
4 6 7 8 9 10
```

Inner Loops:

- Loop within a loop

In [18]:

```
s = int(input())
for r in range(1,s+1):
    for c in range(1,s+1):
        print(c,end=" ")
    print(end="\n")
```

```
5
1 2 3 4 5
1 2 3 4 5
1 2 3 4 5
1 2 3 4 5
1 2 3 4 5
```

Input: 3

Output:

```
        1 2 3
```

```
4 5 6
7 8 9
```

In [26]:

```python
s = int(input())
g = 0
for r in range(1,s+1):
    for c in range(1,s+1):
        g+=1
        print("{:2}".format(g),end=" ")
    print(end="\n")
```

```
5
 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
```

In [34]:

```python
s = int(input())
for r in range(1,s+1):
    for c in range(1,s+1):
        if r==1 or c==1 or r==s or c==s:
            print("*",end=" ")
        else:
            print(" ",end=" ")
    print(end="\n")
```

```
5
* * * * *
*       *
*       *
*       *
* * * * *
```

```
 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

 1 **  3 **  5 **  7
**  9 ** 11 ** 13 **
15 ** 17 ** 19 ** 21
** 23 ** 25 ** 27 **
29 ** 31

 1 **  3 **  5 **  7
**  9 ** 11 ** 13 **
15 ** 17 ** 19 ** 21
## ## ## ## ## ## ##
## ## ##

() ** :: ** () || ()
** :: ** [] ** || **
:: ** [] ** [] || ::
## ## ## ## ## || ##
## ## ##
```

Strings and String Functions

In [47]:

```
1 s = "Anits College"
```

In [74]:

```
print(s)
print(len(s))
print(s[:1]) # to print the starting value of string
print(s[1:]) # to print the string except starting value
print(s[::-1]) # to print the reverse of a string
print(s[2:8:2]) #to print from start (2),end(8),with step(2)
print(s[:5]) # to print first 5 chars
print(s[5:]) #to print chars after 5th char
print(s[-7:]) # to print last 7 chars
print(s[::2])# to print odd position
print(s[1::2]) # to print  even position
print(s[4::-1]) # to print the first 5 chars in reverse
print(s[::-2])# to print odd positions in reverse order
print(s[-2::-2]) # to print even positions in reverse order
```

...

In [88]:

```
print(type(s))
print(s.split(" "))
print(s.isupper())
print(s.islower())
print(s.lower())
print(s.upper())
print(s.count('e'))
```

```
<class 'str'>
['Anits', 'College']
False
False
anits college
ANITS COLLEGE
2
```

In [94]:

```
a = "this is python workshop"
print(a.capitalize())# first letter of first word will be capi
print(a.title())# first letter of every word will be capitaliz
```

```
This is python workshop
This Is Python Workshop
```

In [95]:

```
print(dir(str))
```

```
['__add__', '__class__', '__contains__', '__delattr_
_', '__dir__', '__doc__', '__eq__', '__format__', '_
_ge__', '__getattribute__', '__getitem__', '__getnew
args__', '__gt__', '__hash__', '__init__', '__init_s
ubclass__', '__iter__', '__le__', '__len__', '__lt_
_', '__mod__', '__mul__', '__ne__', '__new__', '__re
duce__', '__reduce_ex__', '__repr__', '__rmod__', '_
_rmul__', '__setattr__', '__sizeof__', '__str__', '_
_subclasshook__', 'capitalize', 'casefold', 'cente
r', 'count', 'encode', 'endswith', 'expandtabs', 'fi
nd', 'format', 'format_map', 'index', 'isalnum', 'is
alpha', 'isascii', 'isdecimal', 'isdigit', 'isidenti
fier', 'islower', 'isnumeric', 'isprintable', 'isspa
ce', 'istitle', 'isupper', 'join', 'ljust', 'lower',
'lstrip', 'maketrans', 'partition', 'replace', 'rfin
d', 'rindex', 'rjust', 'rpartition', 'rsplit', 'rstr
ip', 'split', 'splitlines', 'startswith', 'strip',
'swapcase', 'title', 'translate', 'upper', 'zfill']
```

In [96]:

```
1 print(help(str))
```

```
Help on class str in module builtins:

class str(object)
 |  str(object='') -> str
 |  str(bytes_or_buffer[, encoding[, errors]]) ->
str
 |
 |  Create a new string object from the given obje
ct. If encoding or
 |  errors is specified, then the object must expo
se a data buffer
 |  that will be decoded using the given encoding
and error handler.
 |  Otherwise, returns the result of object.__str_
_() (if defined)
 |  or repr(object).
 |  encoding defaults to sys.getdefaultencoding().
 |  errors defaults to 'strict'.
 |
```

In [ ]:

```
1
```