

Қатысушының шешімдерін толтыруға арналған өріс / Поле для заполнения решений участника Парақ / Страница №

1 - тапсырма

$$c = \sqrt{a^2 + b^2}$$

есептеу бағдарламасы

```
import math
```

```
def hypotenuse (a, b):
```

```
    return math.sqrt(a**2 + b**2)
```

Тапсырмадан енгізу алу

```
a = float(input("a катетін енгізіңіз:"))
```

```
b = float(input("b катетін енгізіңіз:"))
```

Тапсырмадан есептеу және шығару

```
c = hypotenuse(a, b)
```

```
print(f"Гипотенуз: {c}")
```

2 - тапсырма

Автомобиль күн сайан n километр жер жүреді. Қандаған m километр жерде қиын жағдай күндері санан есептеу қиын жағдайға қалғанға болады:

$$\text{күндер саны} = \left\lfloor \frac{m}{n} \right\rfloor$$

Мұндаған $\{x\}$ - x санан жоғары дөңгелектеу

Деректерді енгізу:

```
n = int(input("Күн сайан жүрілетін қашықтақтама (n) енгізіңіз (1-ден 10000-ға дейін)"))
```

```
m = int(input("Жер қашықтақтама (m) енгізіңіз (1-ден 10000-ға дейін)"))
```

Енгізілетін мәндерден n мен m арақашықтақтама екенін тексеру.

```
if 1 <= n <= 10000 and 1 <= m <= 10000
```

Қажетті күндер санан есептеу

```
days = -(m // n) # m-ді n-ге бөліп жоғары дөңгелектеу.
```

```
print(f"Автомобиль {m} км жерде {days} күнде жүріп өтеді")
```

3-мәсәле

Өлшемдерді енгізу

```
N, M = map(int, input().split())
```

Матрицаны енгізу және K санын санау

```
matrix = []
```

```
for _ in range(M):
```

```
    row = list(map(int, input().split()))
```

```
    matrix.append(row)
```

Түзетін санды енгізу

```
K = int(input())
```

- 1 Бірінші жолдан матрицаның өлшемдерін (N және M) алады.
- 2 Кескі N жолдан матрицаны енгізеді.
- 3 Сөзін енгізген санды K - түзетін санды.
- 4 Барлық жолдар бойынша K санды қанша рет кездестіретіндігі есептеледі.
- 5 Нәтижені экранға шығарады.

Қатысушының шешімдерін толтыруға арналған өріс / Поле для заполнения решений участника Парақ / Страница №

1 тапсырма.

$$c = \sqrt{a^2 + b^2}$$

есептеу бағдарламасы

import math

def hypotenuse (a, b):

return math.sqrt (a**2 + b**2)

Пайдаланушыға екісін алу.

a = float (input ("a катетін екізгізіңіз:"))

b = float (input ("b катетін екізгізіңіз:"))

Түптеңізден есептеу нәтижесін шығару.

c = hypotenuse (a, b)

print (f; Түптеңіз: {c})

2 - тапсырма

Автокөлік жүк сыйымдылығы мен километр қиын жүргізу. Құндылығы мен километр жолға
қуық қажет күндер санын есептеу үшін негізгі формуланы пайдалануға болады:

$$\text{күндер саны} = \left\lceil \frac{m}{n} \right\rceil$$

мұндағы [x] - x саны жоғарыға дөңселектемесі

Декларация етуі.

n = int (input ("Жүк сыйымдылығын қамтамасыз етуге (n) екізгізіңіз (1-ден 10000-
ға дейін)."))

m = int (input ("Меншік қамтамасыз етуге (m) екізгізіңіз (1-ден 10000-ға дейін)."))

Екізгізіңіздер 1-ден 10000-ға дейін және екісі тең емес.

$$1 < n <= 10000 \text{ and } 1 < m <= 10000$$

Қажетті күндер санын есептеу.

days = - (-m // n) # m-ді n-ге бөліп, жоғарыға дөңселектемесі.

print (f"Автокөлік {m} тәуір жолға {days} күнге жүргізу керек")

Қатысушының шешімдерін толтыруға арналған өріс / Поле для заполнения решений участника Парақ / Страница №

3- тапсырма.

Әдіс-тәсілдерді енгізу.

$N, M = \text{map}(\text{int}, \text{input}().\text{split}())$

Матрицаны енгізу және K санын санау.
matrix.

count = 0

for - in range(N):

row = list(map(int, input().split()))

matrix.append(row)

Бүтін санды енгізу

$K = \text{int}(\text{input}())$

- 1) Бірінші жағдай: матрицаның өлшемдерін (N және M) алу.
- 2) N және M жағдай: матрицаны енгізеді.
- 3) Санға енгізілген санды K - бүтін санды сан.
- 4) Бірінші жағдай бойынша K саны қанша уақыт шығатынын есептейді.
- 5) Қолжетпейтін жағдайда шығарады.

Қатысушының шешімдерін толтыруға арналған өріс / Поле для заполнения решений участника Парақ / Страница №

1-Тансырма
`import math`

`a=10`
`b=20`

`print(math.sqrt(2**3 + b**2))`

2-Тансырма

`n=100`

`m=1000`

`print(m/n, "күндер")`

3-Тансырма.

`N=5`

`M=5`

`K=5`

`strings = [`

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

`6, 7, 8, 5, 9,`

`10, 11, 5, 12, 13,`

`14, 5, 15, 16, 17,`

`5, 18, 19, 20, 5.`

`]`

`K_count = 0`

Қатысушының шешімдерін толтыруға арналған өріс / Поле для заполнения решений участника Парақ / Страница №

1) `print(int(input()) ** 2 + int`

`(input()) ** 2) ** 3`

2) `#include <iostream>`

`int main() {`

`int n, m;`

`cin >> n >> m;`

`cout << (k+n-1)/n << endl;`

`return 0; }`

2

`a = 700`

`b = 2 * 100`

`c = 2 * 100 / 700`

`print(f)`

3. 1- орыптағы программау Python.

2- тишеге на новаи сүктеме, matrix = [[1, 2, 3, 4, 5]]

2 [6, 12, 8, 9, 10]

3 [11, 12, 12, 19, 55]

4 [16, 17, 18, 12, 20]

дәл ұдобствал
сүрдімнен
се мөткен
сделат мак

3. Утобсе затолнитә таблицу зилел, нутоно по очереди затолнитә кождую строчу:

Қатысушының шешімдерін толтыруға арналған өріс / Поле для заполнения решений участника Парақ / Страница №

1. `n_rows = list(map(int, input().split()))[0]`

2

3. `matrix = []`

4

5. `for_in range(n_rows):`

6. `matrix.append(list(map(int, input().split())))`

1. `for row in matrix:`

2. `for item in row:`

3. `print(item)`

Басқосуға нәтижелер: `print(count)`

Задача 3.

```
Енгізілетін екі нүсқа  
N, M = map(int, input().split())  
Матрица екі нүсқа және кәсіптік санау  
матрица  
санау = 0  
for i in range(N):
```

Задача 1

Гипотенуза теоремасына: $c^2 = a^2 + b^2$ теориясы пайдаланылады.

$$c = \sqrt{a^2 + b^2} = c = \sqrt{3^2 + 4^2}$$

import math

Ввод данных:

```
a, b = map(int, input().split())
```

→ Көрсеткіш

Вычисление гипотенузы:

```
c = math.sqrt(a**2 + b**2)
```

3-тарсыма жауап

Вывод результата

```
for i in range(1, n+1):  
    print(i)
```

print(c)

судайын санау енгізуде

```
k = int(input())
```

Алғашқы жауап шығаруға өзіңіздің | N | M | аяқтау

Қатысушының шешімдерін толтыруға арналған өріс / Поле для заполнения решений участника Парақ / Страница №

Задача

Задача 1

$$c = \sqrt{a^2 + b^2}$$

ескелу бағалымағы

Гурат мат.

def function (a, b):

return math.sqrt(a**2 + b**2)

Талапталушыларға енгізу ағы

a = float(input("a келесі енгізіңіз: "))

b = float(input("b көрсетіп енгізіңіз"))

Талапталушыларға ескелу күше шығару

c = function(a, b)

print(f, "Талапталушы: {c}")

2 төлөвү

$$x \text{ саны} = \left[\frac{a}{n} \right]$$

мында $\left[x \right]$ - x саны жақсартуға дөңделгендігі

және енгізу

n = int(input("Күе санына керсетіңіз (n) енгізіңіз | L-ден 10000-ға дейі"))

m = int(input("және керсетіңіз (m) енгізіңіз | L-ден 10000-ға дейі"))

Парақтың артқы жағын толтырмаңыз / Обратную сторону листа не заполнять

2 төлөвү жақсарту

Есептеулерді жүзеге асыру және жақсарту

if 1 <= n <= 10000 and 1 <= m <= 10000
және енгізу күше шығару

days = (m // n) * 24 + m % n // 24
шағарға дөңделгендігі

Print(f "Жақсарту: {days} күне")

{days} күне керсетіңіз

Қатысушының шешімдерін толтыруға арналған өріс / Поле для заполнения решений участника Парақ / Страница №

1 задание

```
import math  
a = 10
```

```
b = 20
```

```
print math a**2 + b**2 (math.sqrt(a**2 + b**2))
```

2 задание

```
n = 100
```

```
m = 1000
```

```
print(m/n + "грней")
```

3 задание

```
k = 5
```

```
M = 5
```

```
k = 5
```

```
strings = [
```

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

```
6, 7, 8, 5, 9,
```

```
10, 11, 5, 12, 13,
```

```
14, 5, 15, 16, 17,
```

```
18, 19, 20, 5.
```

```
]
```

```
k_count = 0
```

Қатысушының шешімдерін толтыруға арналған өріс / Поле для заполнения решений участника Парақ / Страница №

```
for i in strings:  
    if i == 5:  
        k_count += 1  
  
print(k_count)
```

Қатысушының шешімдерін толтыруға арналған өріс / Поле для заполнения решений участника Парақ / Страница №

```
1 print((int(input()) ** 2 + int(input()) ** 2) ** 0.5)
```

Входные данные.

```
num1 = 300
```

```
num2 = 500
```

```
print(num1, num2)
```

2-Задача.

```
N = int(input("Пройдите за мной"))
```

```
K = int(input("Сколько раз мне пройти?"))
```

```
d = 0
```

```
K = 0
```

```
while True:
```

```
    K = K + N
```

```
    d = d + 1
```

```
    if K >= K:
```

```
        break
```

```
print("пройдем за.")
```

```
print(d)
```

Входные данные

```
import random
```

```
for i in range(10000):
```

```
    print(i)
```

3 Задача.

```
def count_occurrences(matrix, k):
```

```
    count = 0
```

```
    for row in matrix:
```

```
        count += row.count(k)
```

```
    return count
```

```
matrix = E
```

```
[1, 2, 3, 4]
```

```
[5, 6, 3, 8]
```

```
[9, 1, 2, 3]
```

К=2

```
result = count_occurrences(matrix, k)  
print(f"Элемент {k} встречается {result} раз(a) в матрице.")
```

Входные данные

```
n = int(input())  
m = int(input())  
s = 0  
for i in range(n * m):  
    m = int(input())  
    s = s + m  
print(s)
```